

Oracle® Communications

Cloud Native Core Automated Testing Suite

Guide



Release 24.1.1

F93454-09

September 2024

ORACLE®

Copyright © 2019, 2024, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1 Introduction

1.1	Overview	1
1.2	Deployment Model	2
1.3	References	3

2 ATS Framework Features

2.1	ATS API	3
2.1.1	Generating an API Token for a User	3
2.1.2	Use the RESTful Interfaces	5
2.1.2.1	Configuring Host	5
2.1.2.2	Starting Jobs	6
2.1.2.3	Monitoring Jobs	10
2.1.2.4	Stopping Jobs	11
2.1.2.5	Getting Test Suite Artifacts	13
2.2	ATS Feature Activation and Deactivation	16
2.3	ATS GUI Enhancements	18
2.4	ATS Health Check	20
2.5	ATS Jenkins Job Queue	30
2.6	Application Log Collection	31
2.6.1	Application Log Collection and Parallel Test Execution Integration	33
2.7	ATS Maintenance Scripts	34
2.7.1	ATS Scripts	35
2.7.2	Updating ats_install.sh	36
2.7.3	Restarting Jenkins without Restarting Pod	36
2.7.4	Updating Stub Scripts	37
2.7.5	Running ATS and Stub Deployment Scripts	38
2.8	ATS System Name and Version Display on the ATS GUI	38
2.9	ATS Tagging Support	39
2.9.1	Combination of Tags and their Results	41
2.10	Custom Folder Implementation	43
2.11	Individual Stage Group Selection	44
2.12	Lightweight Performance	45
2.12.1	Configure <NF>-Performance Pipeline	45

2.13	Managing Final Summary Report, Build Color, and Application Log	46
2.14	Modifying Login Password	53
2.15	Multiselection Capability for Features and Scenarios	54
2.15.1	Feature Level Selection	54
2.15.2	Scenario Selection	55
2.16	Parallel Test Execution	56
2.16.1	ATS GUI Page Changes	57
2.16.2	ATS Console Log Changes	58
2.16.3	Downloading or Viewing Individual Group Logs	59
2.17	Parameterization	62
2.17.1	Running Test Cases	63
2.18	PCAP Log Collection	65
2.18.1	Application Log Collection and Parallel Test Execution Integration	65
2.19	Persistent Volume for 5G ATS	66
2.19.1	Processing Flow	66
2.19.2	Deploying Persistent Volume	66
2.19.3	Backward Porting	71
2.20	Single Click Job Creation	71
2.20.1	Configuring Single Click Job	71
2.21	Support for ATS Deployment in OCI	74
2.21.1	Accessing ATS GUI in OCI	74
2.21.2	Adding Ingress and Egress Rules to Access the OCI Console	75
2.22	Support for Transport Layer Security	79
2.22.1	Deploy ATS with TLS Enabled	80
2.22.1.1	Generate JKS File for Jenkins Server	80
2.22.1.2	Enable ATS GUI with HTTPS	85
2.23	Test Results Analyzer	86
2.23.1	Accessing Test Results Analyzer Feature	87
2.24	Support for Test Case Mapping and Count	89
2.24.1	Access Test Case Mapping and Count Feature	90

3 Installing ATS for Different Network Functions

3.1	Installing ATS for BSF	1
3.1.1	Resource Requirements	1
3.1.2	Downloading the ATS Package	4
3.1.3	Pushing the Images to Customer Docker Registry	5
3.1.4	Configuring ATS	6
3.1.4.1	Enabling Static Port	6
3.1.4.2	Enable Static API Node Port	7
3.1.4.3	Service Account Requirements	7
3.1.4.4	Enabling Aspen Service Mesh	7

3.1.4.5	Enabling Health Check	10
3.1.4.6	Enabling Persistent Volume	11
3.1.5	Deploying ATS and Pods	14
3.1.5.1	Deploying ATS in Kubernetes Cluster	14
3.1.5.2	Deploying Stub Pod in Kubernetes Cluster	15
3.1.5.3	Deploying DNS Stub in Kubernetes Cluster	17
3.1.5.4	Deploying ocdiam Simulator in Kubernetes Cluster	18
3.1.6	Post-Installation Steps	20
3.2	Installing ATS for NEF	21
3.2.1	Resource Requirements	22
3.2.2	Downloading the ATS Package	25
3.2.3	Pushing the Images to Customer Docker Registry	27
3.2.4	Configuring ATS	28
3.2.4.1	Enabling Static Port	28
3.2.5	Deploying ATS, Stub and CNC Console in Kubernetes Cluster	29
3.2.5.1	Deploy and Configure CNC Console	30
3.2.5.2	Creating Console secrets	30
3.2.6	Verifying ATS Deployment	31
3.3	Installing ATS for NRF	31
3.3.1	Resource Requirements	31
3.3.2	Downloading the ATS Package	33
3.3.3	Pushing the Images to Customer Docker Registry	34
3.3.4	Configuring ATS	35
3.3.4.1	Enabling Static Port	35
3.3.4.2	Enabling Aspen Service Mesh	35
3.3.4.3	Enabling Persistent Volume	37
3.3.4.4	Enabling NF FQDN Authentication	37
3.3.5	Deploying ATS and Stub in Kubernetes Cluster	38
3.3.6	Verifying ATS Deployment	39
3.3.7	Post-Installation Steps (if Persistent Volume is Used)	40
3.4	Installing ATS for NSSF	40
3.4.1	Resource Requirements	41
3.4.2	Locating and Downloading ATS and Simulator Images	43
3.4.3	Deploying ATS in Kubernetes Cluster	45
3.5	Installing ATS for Policy	51
3.5.1	Resource Requirements	51
3.5.2	Downloading the ATS Package	55
3.5.3	Pushing the Images to Customer Docker Registry	57
3.5.4	Configuring ATS	59
3.5.4.1	Enabling Static Port	59
3.5.4.2	Enabling Static API Node Port	59
3.5.4.3	Service Account Requirements	59

3.5.4.4	Enabling Aspen Service Mesh	60
3.5.4.5	Enabling Persistent Volume	62
3.5.4.6	Enabling Health Check	65
3.5.5	Deploying ATS and Pods	66
3.5.5.1	Deploying ATS in Kubernetes Cluster	66
3.5.5.2	Deploying Stub Pod in Kubernetes Cluster	67
3.5.5.3	Deploying DNS Stub in Kubernetes Cluster	71
3.5.5.4	Deploying AMF Stub in Kubernetes Cluster	72
3.5.5.5	Deploying LDAP Stub in Kubernetes Cluster	73
3.5.5.6	Deploying ocdiam Simulator in Kubernetes Cluster	74
3.5.6	Post-Installation Steps	76
3.6	Installing ATS for SCP	78
3.6.1	Prerequisites	78
3.6.1.1	Software Requirements	78
3.6.1.2	Environment Setup Requirements	79
3.6.1.3	Resource Requirements	79
3.6.1.4	Downloading the ATS Package	81
3.6.1.5	Pushing the Images to Customer Docker Registry	89
3.6.1.6	Preinstall Preparation of SCP for SCP-ATS	91
3.6.1.7	Preinstallation Preparation for SCP-ATS	96
3.6.2	Configuring SCP-ATS and SCP	99
3.6.2.1	Updating ATS Configuration	99
3.6.3	Deploying ATS and Stub in the Kubernetes Cluster	116
3.6.4	Post Installation and Deployment Steps	116
3.6.4.1	Verifying ATS Deployment	117
3.6.4.2	Modifying the scpc-alternate-resolution Microservice	117
3.6.4.3	ATS Testcase Parametrization on User Input	118
3.6.5	Appendix	127
3.6.5.1	Creating Custom Service Account	127
3.6.5.2	Adding New Stubs to SCP-ATS	128
3.7	Installing ATS for SEPP	129
3.7.1	Resource Requirements	129
3.7.1.1	SEPP ATS Compatibility Matrix	131
3.7.2	Downloading the ATS Package	132
3.7.3	Pushing the Images to Customer Docker Registry	133
3.7.4	Creating Secrets and Support for TLSv1.2 and TLSv1.3	134
3.7.4.1	Configuring Root Certificate	134
3.7.4.2	Generating ATS Certificate	135
3.7.4.3	Creating ATS Secret and Configuring Helm chart	139
3.7.4.4	Create ATS Health Check secret	141
3.7.5	Configuring ATS	142
3.7.5.1	Enabling Aspen Service Mesh	142

3.7.5.2	Enabling Static Port	144
3.7.5.3	Enabling Roaming Hub Mode	144
3.7.5.4	Enabling Hosted SEPP Mode	145
3.7.5.5	Configuring Egress Rate Limiting Feature	145
3.7.5.6	Configuring Ingress Rate Limiting Feature	145
3.7.5.7	Configuring the Cache Refresh Timeout Value	146
3.7.5.8	Configuring the Topology Cache Refresh Timeout Value	146
3.7.5.9	Configuring the Security Counter Measure Cache Refresh Timeout	146
3.7.5.10	Configuring the n32cHandshakePlmnIdListValidationEnabled	147
3.7.5.11	Request Timeout	150
3.7.5.12	Idle Timeout	151
3.7.5.13	EvictSanHeaderCacheDelay	152
3.7.5.14	Configuring the Reroute Attempts for Egress Gateway	152
3.7.5.15	NRF Discovery Cache Refresh Timeout	152
3.7.5.16	Configuring Alternate Routing based on the DNS SRV Record for Home Network Functions	153
3.7.5.17	Configuring Load Sharing among Multiple Remote SEPP Nodes	154
3.7.5.18	Enabling Persistent Volume Storage	155
3.7.6	Deploying ATS in Kubernetes Cluster	155
3.7.7	Verifying ATS Deployment	156
3.7.8	Post Installation Steps (If persistent volume is used)	157
3.8	Installing ATS for UDR	157
3.8.1	Resource Requirements	157
3.8.2	Locating and Downloading ATS Image	173
3.8.3	Preparing to Deploy ATS and Stub Pods	175
3.8.4	Loading UDR ATS Image	175
3.8.5	Loading UDR Stub Images in the SLF-NewFeatures, EIR-NewFeatures, SLF-Regression, or EIR-Regression Pipeline	177
3.8.6	Loading UDR Stub Images in the UDR-NewFeatures or UDR-Regression Pipeline	179
3.8.7	Configuring ATS	181
3.8.7.1	Configuring Docker Registry	182
3.8.7.2	Enabling Static Port	182
3.8.7.3	Enabling Persistent Volume	182
3.8.7.4	Settings for OAuth2 Test Cases	183
3.8.7.5	Enabling IPv6 on ATS	184
3.8.7.6	Configuring ATS to Run Health-Check Pipeline	184
3.8.7.7	Creating Service Account	185
3.8.7.8	Enabling Service Mesh	186
3.8.7.9	Configuring Stubs	188
3.8.8	Deploying ATS in Kubernetes Cluster	191
3.8.8.1	Deploying ATS Pod and Stubs in Kubernetes Cluster	192
3.8.9	Verifying ATS Deployment	193

4 Running NF Test Cases Using ATS

4.1	Running BSF Test Cases using ATS	1
4.1.1	Prerequisites	1
4.1.2	Logging into ATS	4
4.1.3	Running BSF_NewFeatures Pipeline	6
4.1.4	Running BSF-HealthCheck Pipeline	11
4.1.5	BSF New Features Documentation	12
4.1.6	Running BSF Regression Pipeline	13
4.1.7	BSF Regression Documentation	14
4.2	Running NEF Test Cases using ATS	14
4.2.1	Prerequisites	14
4.2.2	Logging into ATS	22
4.2.3	NEF-NewFeatures Pipeline	24
4.2.4	NEF-NewFeatures Documentation	31
4.2.5	NEF-Regression Pipeline	31
4.2.6	NEF-Regression Documentation	36
4.3	Running NRF Test Cases using ATS	37
4.3.1	Prerequisites	37
4.3.2	Logging into ATS	46
4.3.3	NRF-NewFeatures Pipeline	48
4.3.4	NRF-NewFeatures Documentation	59
4.3.5	NRF-Regression Pipeline	60
4.3.6	NRF-Regression Documentation	62
4.4	Running NSSF Test Cases using ATS	64
4.4.1	Prerequisites	64
4.4.2	Logging into ATS	64
4.4.3	NSSF-NewFeatures Pipeline	66
4.4.4	NSSF-NewFeatures Documentation	76
4.4.5	NSSF-Regression Pipeline	77
4.4.6	NSSF-Regression Documentation	84
4.5	Running Policy Test Cases using ATS	85
4.5.1	Prerequisites	86
4.5.2	Logging into ATS	91
4.5.3	Running Policy-HealthCheck Pipeline	93
4.5.4	Running Policy-NewFeatures Pipeline	94
4.5.5	Policy New Features Documentation	101
4.5.6	Running Policy Regression Pipeline	102
4.5.7	Policy Regression Documentation	104
4.6	Running SCP Test Cases using ATS	105

4.6.1	Logging in to ATS	106
4.6.2	SCP-NewFeatures Pipeline	108
4.6.2.1	Configuring New Feature Pipelines	108
4.6.2.2	Running SCP New Feature Pipelines	112
4.6.2.3	SCP-NewFeatures Documentation	114
4.6.3	SCP-Regression Pipeline	115
4.6.3.1	Configuring Regression Pipelines	116
4.6.3.2	Running Regression Pipelines	119
4.6.3.3	SCP-Regression Documentation	120
4.7	Running SEPP Test Cases using ATS	129
4.7.1	Prerequisites	129
4.7.2	Logging into ATS	131
4.7.3	SEPP-NewFeatures Pipeline	133
4.7.4	NewFeatures - Documentation	137
4.7.5	SEPP Regression Pipeline	137
4.7.6	SEPP Regression Documentation	138
4.7.7	Running SEPP Health Check Pipeline	139
4.8	Running UDR Test Cases using ATS	141
4.8.1	Prerequisites	141
4.8.1.1	Prerequisites for SLF Pipelines	142
4.8.1.2	Prerequisites for UDR Pipelines	152
4.8.1.3	Prerequisites for EIR Pipelines	165
4.8.2	UDR Golden Configuration Support	170
4.8.3	UDR Application Log Collection	175
4.8.4	Parallel Test Execution	178
4.8.5	Logging into ATS	208
4.8.6	Running SLF Pipelines	209
4.8.6.1	Running SLF-NewFeatures Pipeline	210
4.8.6.2	Running SLF Regression Pipeline - Single Site Deployment	215
4.8.6.3	Running SLF Regression Pipeline - Multiple Site Deployment	219
4.8.6.4	Running SLF HealthCheck Pipeline	222
4.8.6.5	Running SLF-Performance Pipeline: Model 2: Single Site Deployment	224
4.8.6.6	SLF NewFeatures Documentation - Single Site Scenarios	226
4.8.6.7	SLF Regression Documentation - Single Site Scenarios	228
4.8.6.8	SLF Regression Documentation - Multiple Site Scenarios	228
4.8.7	Running UDR Pipelines	229
4.8.7.1	Running UDR NewFeatures Pipeline	229
4.8.7.2	Running UDR Regression Pipeline	233
4.8.7.3	Running UDR HealthCheck Pipeline	241
4.8.7.4	UDR NewFeatures Documentation	243
4.8.7.5	UDR Regression Documentation	244
4.8.8	Running EIR Pipelines	244

4.8.8.1	Running EIR NewFeatures Pipeline	245
4.8.8.2	Running EIR Regression Pipeline	248
4.8.8.3	Running EIR HealthCheck Pipeline	252
4.8.8.4	EIR NewFeatures Documentation	254
4.8.8.5	EIR Regression Documentation	255

My Oracle Support

My Oracle Support (<https://support.oracle.com>) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at <http://www.oracle.com/us/support/contact/index.html>. When calling, make the selections in the sequence shown below on the Support telephone menu:

- For Technical issues such as creating a new Service Request (SR), select **1**.
- For Non-technical issues such as registration or assistance with My Oracle Support, select **2**.
- For Hardware, Networking and Solaris Operating System Support, select **3**.

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Acronyms

The following table lists the acronyms and the terminologies used in the document:

Table Acronyms

Acronym	Meaning
ATS	Oracle Communications Cloud Native Core, Automated Test Suite
BDD	Behavior Driven Development
BSF	Oracle Communications Cloud Native Core, Binding Support Function
CI	Continuous Integration
EIR	Equipment Identity Register
Policy	Oracle Communications Cloud Native Core, Converged Policy
NEF	Oracle Communications Cloud Native Core, Network Exposure Function
NRF	Oracle Communications Cloud Native Core, Network Repository Function
NSSF	Oracle Communications Cloud Native Core, Network Slice Selection Function
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment
OL	Oracle Linux is a Linux distribution packaged and freely distributed by Oracle, available partially under the GNU (General Public License) since late 2006. It is compiled from Red Hat Enterprise Linux source code, replacing Red Hat branding with Oracle's.
PCF	Oracle Communications Cloud Native Core, Policy Control Function
cnPCRF	Oracle Communications Cloud Native Core, Policy and Charging Rules Function
SAN	Subject Alternative Name
SCP	Oracle Communications Cloud Native Core, Service Communication Proxy
SEPP	Oracle Communications Cloud Native Core, Security Edge Protection Proxy
SLF	Subscriber Location Function
UDR	Oracle Communications Cloud Native Core, Unified Data Repository

What's New in This Guide

This section introduces the documentation updates for release 24.1.x.

ATS Framework Release 24.1.1 - F93454-09, September 2024

NRF Release 24.1.3

- ATS release tag for NRF is changed to 24.1.3.
- Updated the following sections with the new images for 24.1.3:
 - [Prerequisites](#)
 - [Logging into ATS](#)
 - [NRF-NewFeatures Pipeline](#)
 - [NRF-Regression Pipeline](#)
- **Number of new Test Cases added to pipeline:**
 - **New Features:-** Provides a total of **15** feature files with **68** scenarios.
 - **Regression:-** Provides a total of **572** feature files with **1589** scenarios.
 - * "2-Site-Geo" cases: These are part of Regression pipeline. It has 6 feature files with 15 scenarios. These are already provided 2 Site Geo test cases.
 - * "3-Site-Geo" cases: These are part of Regression pipeline. There are 10 feature files with 14 scenarios.
 - * "NF-FQDN-Authentication-Feature" cases: These are also part of Regression pipeline. It has 5 feature files with 25 scenarios.
 - * "CCA-Header" cases: These are part of Regression pipeline. It has 5 feature files with 17 scenarios.
 - * "NRF-Growth": Provides a total of 20 features files with 27 new scenarios.
 - **Total regression count :-** Provides a total of **572** feature files with **1589** scenarios.

ATS Framework Release 24.1.1 - F93454-08, September 2024

NRF Release 24.1.2

- ATS release tag for NRF is changed to 24.1.2.
- Updated the following sections with the new images for 24.1.2:
 - [Prerequisites](#)
 - [Logging into ATS](#)
 - [NRF-NewFeatures Pipeline](#)
 - [NRF-Regression Pipeline](#)
- **Number of new Test Cases added to pipeline:**
 - **New Features:-** Provides a total of **14** feature files with **57** scenarios.
 - **Regression:-** Provides a total of **572** feature files with **1589** scenarios.
 - * "2-Site-Geo" cases: These are part of Regression pipeline. It has 6 feature files with 15 scenarios. These are already provided 2 Site Geo test cases.

- * "3-Site-Geo" cases: These are part of Regression pipeline. There are 10 feature files with 14 scenarios.
 - * "NF-FQDN-Authentication-Feature" cases: These are also part of Regression pipeline. It has 5 feature files with 25 scenarios.
 - * "CCA-Header" cases: These are part of Regression pipeline. It has 5 feature files with 17 scenarios.
 - * "NRF-Growth": Provides a total of 20 features files with 27 new scenarios.
- **Total regression count** :- Provides a total of **617** feature files with **1680** scenarios.

ATS Framework Release 24.1.1 - F93454-07, August 2024

UDR Release 24.1.1

- ATS release tag for UDR is changed to 24.1.1.
- **Number of Test Cases in UDR-NewFeatures pipeline:** Provides a total of **35** test cases.
- Added UDR_Suppressed_notification_to_the_initiator feature to the UDR New Features List. For more information, see [Table 4-19](#).

ATS Framework Release 24.1.1 - F93454-06, June 2024

SCP Release 24.1.0

- ATS release tag for SCP is changed to 24.1.1.
- Added the sutScpIPFamiliesforATSRUN parameter to the deployment file under global parameters in the [Downloading the ATS Package](#) section to specify the IP families ATS considers for running test cases.

ATS Framework Release 24.1.1 - F93454-04, June 2024

NSSF Release 24.1.1

- ATS release tag for NSSF is changed to 24.1.1.
- Updated installation process in [Installing ATS for NSSF](#) section.
- **New Test Cases added:** No.

NRF Release 24.1.1

- **Number of Test Cases added to pipeline:**
 - **Regression:** Provides a total of **571** feature files with **1586** scenarios.
 - * "2-Site-Geo" cases: These are part of Regression pipeline. It has 6 feature files with 15 scenarios. These are already provided 2 Site Geo test cases.
 - * "3-Site-Geo" cases: There are 10 feature files with 14 scenarios.
 - * "NF-FQDN-Authentication-Feature" cases: These are also part of Regression pipeline. It has 5 feature files with 25 scenarios.
 - * "CCA-Header" cases: These are part of Regression pipeline. It has 5 feature files with 17 scenarios.
 - * "NRF-Growth": Provides a total of 20 features files with 27 new scenarios.
 - * "All-Except-Above-Options" suite: Provides a total of 571 feature files with 1585 scenarios.
 - **Total regression count:** Provides a total of **617** feature files with **1680** scenarios.

- Updated the ocats_ocnrf_csar_24.1.1_0_0.zip.sha256 filename to ocats_ocnrf_csar_24_1_1_0_0.zip.sha256 in the [Downloading the ATS Package](#) section.
- Updated the ocats_ocnrf_csar_24.1.1_0_0.zip filename to ocats_ocnrf_csar_24_1_1_0_0.zip in the following sections:
 - [Pushing the Images to Customer Docker Registry](#)
 - [Downloading the ATS Package](#)

ATS Framework Release 24.1.1 - F93454-03, May 2024

NRF Release 24.1.1

- ATS release tag for NRF is changed to 24.1.1.
- Updated the following sections with the new images for 24.1.1:
 - [Prerequisites](#)
 - [Logging into ATS](#)
 - [NRF-NewFeatures Pipeline](#)
 - [NRF-Regression Pipeline](#)
- **Number of new Test Cases added to pipeline:**
 - **Regression:-** Provides a total of **571** feature files with **1586** scenarios.
 - **Total regression count :-** Provides a total of **617** feature files with **1680** scenarios.

ATS Framework Release 24.1.1 - F93454-02, May 2024

NRF Release 24.1.0

Updated the cnDBTier resource requirement details in the [Resource Requirements](#) section.

ATS Framework Release 24.1.1 - F93454-01, April 2024

Updated the following sections with the details of ATS 24.1.0:

Added the following features in the [ATS Framework Features](#) section:

- [Multiselection Capability for Features and Scenarios](#) feature to select and run single or multiple features and scenarios.
- [Support for ATS Deployment in OCI](#) feature to deploy and run ATS in an Oracle Cloud Infrastructure (OCI) environment.
- [ATS GUI Enhancements](#) feature to navigate effortlessly between different functionalities.

NEF Release 24.1.0

- ATS release tag for NEF is changed to 24.1.0.
- Updated the following sections with 24.1.0 release updates:
 - [Resource Requirements](#)
 - [Downloading the ATS Package](#)
 - [Pushing the Images to Customer Docker Registry](#)
 - [Deploying ATS, Stub and CNC Console in Kubernetes Cluster](#)
- Added [Deploy and Configure CNC Console](#) as part of NEF Integration with the CNC Console feature.

- The following sections are updated for the Support for MSISDN-Less MO-SMS and NEF Integration with the CNC Console features:
 - [Prerequisites](#)
 - [NEF-NewFeatures Pipeline](#)
 - [Resource Requirements](#)
- Number of test cases in NEF_NewFeatures pipeline: 17 new feature files and 85 new scenarios
- Number of test cases in NEF_Regression pipeline: 103 feature files and 610 scenarios

BSF Release 24.1.0

- ATS release tag for BSF is changed to 24.1.0.
- Updated [ATS Framework Features](#) section for BSF 24.1.0 features.
- Added ATS tagging support (Framework support only).
- ATS UI displays test cases under their stage and group names on BuildWithParameters page.
- Added support for individual scenario-based selection.
- Added support for stage-group selection.
- **Number of new Test Cases added to pipeline:** Provides a total of **1 feature file** and **6 scenario** of BSF_NewFeatures pipeline.
- **Previous Release Test Cases:** Provides a total of **27 feature files** and **80 scenarios** of BSF_Regression pipeline.
- Updated the following sections with 24.1.0 release updates under **Installing ATS for BSF:**
 - [Resource Requirements](#)
 - [Downloading the ATS Package](#)
 - [Enabling Static Port](#)
 - [Enable Static API Node Port](#)
 - [Enabling Aspen Service Mesh](#)
- Updated the following sections with 24.1.0 release updates under **Running BSF Test Cases using ATS:**
 - [Running BSF_NewFeatures Pipeline](#)
 - [BSF New Features Documentation](#)
 - [Running BSF Regression Pipeline](#)
 - [BSF Regression Documentation](#)

Policy Release 24.1.0

- ATS release tags for Policy is changed to 24.1.0.
- Number of new Test Cases added to pipeline (PCF): Provides a total of 11 feature files and 17 scenarios of Policy_NewFeatures pipeline.
- Number of new Test Cases added to pipeline (cnPCRF): Provides a total of 0 feature file and 0 scenarios of Policy_NewFeatures pipeline.
- Number of new Test Cases added to pipeline (Converged Policy): Provides a total of 15 feature files and 30 scenarios of Policy_NewFeatures pipeline.

- Previous Release Test Cases (PCF): Provides a total of 195 feature files and 585 scenarios of Policy_Regression pipeline.
- Previous Release Test Cases (cnPCRf): Provides a total of 24 feature files and 83 scenarios of Policy_Regression pipeline.
- Previous Release Test Cases (Converged Policy): Provides a total of 248 feature files and 749 scenarios of Policy_Regression pipeline.
- Added details of Policy ATS feature **MultiSelection Capability for Features and Scenarios** to [Multiselection Capability for Features and Scenarios](#).
- Added details of Policy ATS feature **Individual Stage Group Selection** to [Individual Stage Group Selection](#).
- Updated the following sections with 24.1.0 release updates:
 - [Downloading the ATS Package](#)
 - [Pushing the Images to Customer Docker Registry](#)
- Updated the following sections with 24.1.0 release updates under Running Policy Test Cases using ATS:
 - [Logging into ATS](#)
 - [Running Policy-HealthCheck Pipeline](#)
 - [Running Policy-NewFeatures Pipeline](#)
 - [Policy New Features Documentation](#)
 - [Running Policy Regression Pipeline](#)
 - [Policy Regression Documentation](#)

SCP Release 24.1.0

- ATS release tag for SCP is changed to 24.1.0.
- Number of test cases in SCP_NewFeatures pipeline: 23 new feature files with 129 new scenarios for Release 16 and no new feature file with no new scenarios for Release 15 SCP deployments.
- Number of test cases in the SCP_Regression pipeline: 298 feature files with 1964 scenarios for Release 16, and 39 feature files with 366 scenarios for Release 15 SCP deployments.
- Number of sanity features: 27 sanity features and 46 test cases for Release 16.
- Renamed the feature files as per the naming convention in the [Renaming of Regression Feature Files](#) section.
- Added the following sections:
 - [Dual Stack Support](#) to provide information on the type of endpoint the stack supports.
 - [Modifying IpFamilyPolicy or IpFamilies of Stubs](#) to understand about the deployment of stubs.
- Updated the following sections with 24.1.0 release updates:
 - [Preinstall Preparation of SCP for SCP-ATS](#)
 - [Configuring ATS for CCA Test Cases](#)
 - [Logging in to ATS](#)
 - [SCP-NewFeatures Pipeline](#)
 - [SCP-Regression Pipeline](#)

- [Downloading the ATS Package](#)
- [Modifying the scpc-alternate-resolution Microservice](#)

NRF Release 24.1.0

- ATS release tag for NRF is changed to 24.1.0.
- Updated the following sections with the new images for 24.1.0
 - [Prerequisites](#)
 - [Logging into ATS](#)
 - [NRF-NewFeatures Pipeline](#)
 - [NRF-Regression Pipeline](#)
- **New Test Cases added:** Yes.
- **Number of new Test Cases added to pipeline:**
 - **NewFeatures** :- Provides a total of 11 features files with 54 new scenarios.
 - **Regression**:- Provides a total of 571 feature files with 1585 scenarios.
 - **Total regression count** :- Provides a total of 597 feature files with 1656 scenarios.
- Added the annotation for excludeOutboundPorts in the following sections:
 - [Prerequisites](#)
 - [Enabling Aspen Service Mesh](#)
- NRF-ATS supports OSO 24.1.x.

NSSF Release 24.1.0

- ATS release tag for NSSF is changed to 24.1.0.
- Updated the following sections with the new images for 24.1.0
 - [Prerequisites](#)
 - [Logging into ATS](#)
 - [NSSF-NewFeatures Pipeline](#)
 - [NSSF-Regression Pipeline](#)
- Updated installation process in [Installing ATS for NSSF](#) section.
- **New Test Cases added:** Yes.
- **Number of new Test Cases added to pipeline:**
 - **NewFeatures:** Provides a total of 2 features files with 39 new scenarios.
 - **Regression:** Provides a total of 74 feature files with 692 scenarios.

UDR Release 24.1.0

- ATS release tag for UDR is changed to 24.1.0.
- Updated multiple images according to release 24.1.0.
- **Number of Test Cases in UDR-NewFeatures pipeline:** Provides a total of **24** test cases.
- **Number of Test Cases in EIR-NewFeatures pipeline:** Provides a total of **3** test cases.
- **Number of Test Cases in Single and Multiple Site SLF-NewFeatures pipeline:** **33** test cases.

- **Number of Test Cases in SLF-Regression pipeline:** 332 test cases with 21 MultiSite scenarios.
- **Number of Test Cases in UDR-Regression pipeline:** 1364 test cases with 22 MultiSite scenarios.
- **Number of Test Cases in EIR-Regression pipeline:** 276 test cases.
- Updated the values in ATS Resource Requirements for EIR mode and EIR - Total Number of Resources tables. For more information, see [Resource Requirements](#).
- Added EIR ATS Stages and Groups for Parallel Test Execution. For more information, see [Parallel Test Execution](#).
- Updated the images in the following sections:
 - [Running SLF-NewFeatures Pipeline](#)
 - [Running SLF Regression Pipeline - Single Site Deployment](#)
 - [Running SLF Regression Pipeline - Multiple Site Deployment](#)
 - [Running SLF HealthCheck Pipeline](#)
 - [Running SLF-Performance Pipeline: Model 2: Single Site Deployment](#)
 - [SLF NewFeatures Documentation - Single Site Scenarios](#)
 - [SLF Regression Documentation - Single Site Scenarios](#)
 - [SLF Regression Documentation - Multiple Site Scenarios](#)
 - [Running UDR NewFeatures Pipeline](#)
 - [Running UDR Regression Pipeline](#)
 - [Running UDR HealthCheck Pipeline](#)
 - [UDR NewFeatures Documentation](#)
 - [UDR Regression Documentation](#)
 - [Running EIR NewFeatures Pipeline](#)
 - [Running EIR Regression Pipeline](#)
 - [Running EIR HealthCheck Pipeline](#)
 - [EIR NewFeatures Documentation](#)
 - [EIR Regression Documentation](#)
 - Updated the New Features list in the following sections:
 - * [SLF NewFeatures Documentation - Single Site Scenarios](#)
 - * [UDR NewFeatures Documentation](#)
 - * [EIR NewFeatures Documentation](#)

SEPP Release 24.1.0

- ATS release tag for SEPP is changed to 24.1.0.
- Updated multiple images according to release 24.1.0.
- Updated the [Resource Requirements](#) section.
- Updated the [SEPP ATS Compatibility Matrix](#) section.
- Updated the following sections:
 - [Create ATS Health Check secret](#)

- [Enabling Aspen Service Mesh](#)
- [Configuring Load Sharing among Multiple Remote SEPP Nodes](#)
- Number of test cases in SEPP_NewFeatures pipeline (in SEPP mode): 2 new feature files and 10 new scenarios.
- Number of test cases in SEPP_Regression pipeline (in SEPP mode): 148 feature files and 852 scenarios.
- Number of test cases in SEPP_NewFeatures pipeline (in Roaming Hub mode): 2 new feature files and 10 new scenarios.
- Number of test cases in SEPP_Regression pipeline (in Roaming Hub mode): 133 feature files and 652 scenarios.

1

Introduction

This document provides information about Automated Testing Suite (ATS) deployment model for 5G Network Functions (NFs).

ATS allows you to run software test cases using an automated testing tool. It compares the actual results with the expected or predicted results.

1.1 Overview

Using ATS, you can deploy and test 5G NFs.

This document provides information to implement ATS for the following 5G NFs:

- Oracle Communications Cloud Native Core, Binding Support Function (BSF)
- Oracle Communications Cloud Native Core, Converged Policy (Policy)
- Oracle Communications Cloud Native Core, Network Exposure Function (NEF)
- Oracle Communications Cloud Native Core, Network Repository Function (NRF)
- Oracle Communications Cloud Native Core, Network Slice Selection Function (NSSF)
- Oracle Communications Cloud Native Core, Service Communication Proxy (SCP)
- Oracle Communications Cloud Native Core, Security Edge Protection Proxy (SEPP)
- Oracle Communications Cloud Native Core, Unified Data Repository (UDR)

ATS for 5G Network Functions

For 5G NFs, ATS is developed using Oracle Linux 8-slim as the base image. Jenkins is a part of the ATS image, and it provides a graphical user interface (GUI) to test either a single NF or multiple NFs independently in the same network environment.

ATS comprises NF docker images, ATS image, simulator images, and test cases of the specific NF. All these images and test cases constitute a fully automated suite to deploy and test NFs. You can combine it with any other Continuous Integration (CI) pipeline with minimal changes because 5G ATS uses Jenkins as GUI.

The ATS package contains the following elements:

- Test scripts and docker images of test container. The docker images have complete framework and libraries installed, which are common for all NFs working with the Behavior Driven Development (BDD) framework.
- Docker image of HTTP Server simulator.
- Helm chart to deploy the ATS (delivered as a tar file).
- Readme text file (.txt file).

ATS provides basic environment, framework, and a GUI (Jenkins) to run all the functional test cases.

ATS Framework Version with Supporting NF Version

Table 1-1 ATS Framework Version with Supporting NF

ATS Framework Version	BSF	NEF	NRF	NSSF	Policy	SCP	SEPP	UDR
24.1.1	24.1.0	24.1.0	24.1.3	24.1.1	24.1.0	24.1.1	24.1.0	24.1.1

1.2 Deployment Model

According to the In-Cluster deployment model, ATS can coexist in the same cluster where the NFs are deployed. This deployment model is useful for In-Cluster testing.

Figure 1-1 In-Cluster Deployment Model in OCCNE

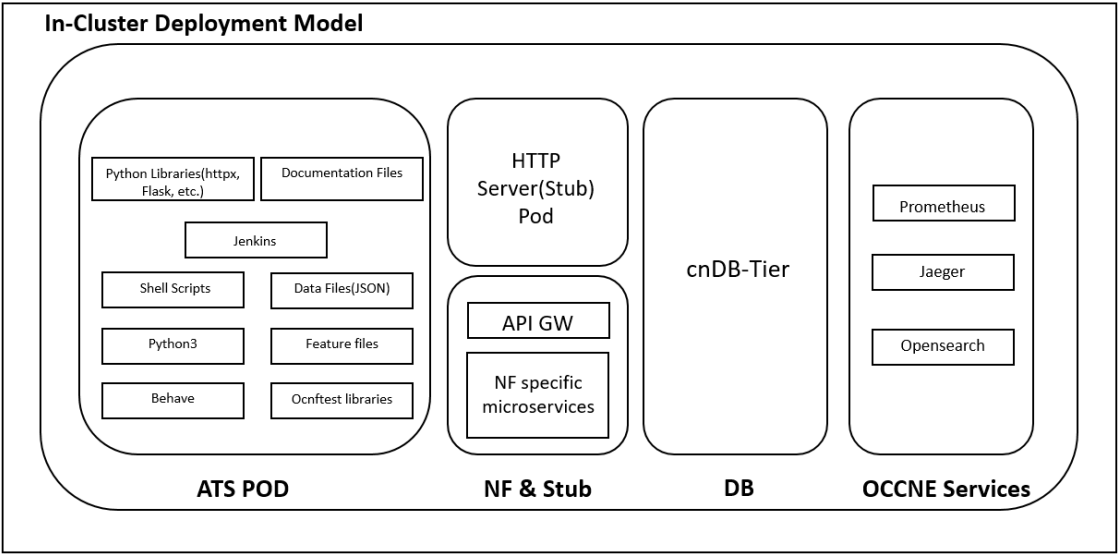
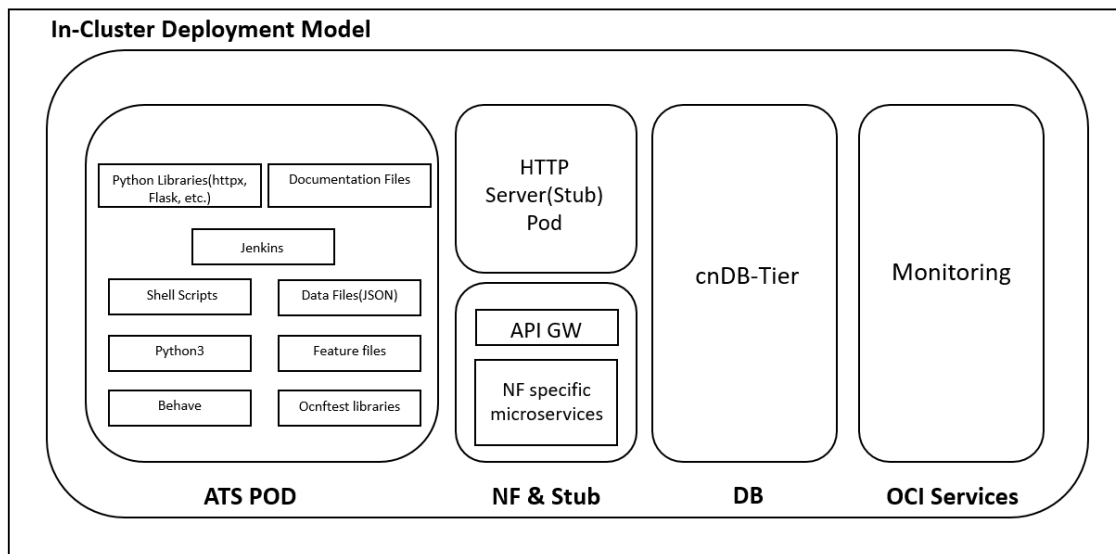


Figure 1-2 In-Cluster Deployment Model in OCI

1.3 References

For more information about NFs and their deployment processes, refer to the following documents:

- *Oracle Communications Cloud Native Core, Binding Support Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Exposure Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Slice Selection Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Networks Data Analytics Function Installation Guide*
- *Oracle Communications Cloud Native Core, Policy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Provisioning Gateway Guide*
- *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Security Edge Protection Proxy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*

2

ATS Framework Features

This chapter describes ATS Framework features.

The following table lists ATS framework features supported by different NFs:

Table 2-1 ATS Framework Features Compliance Matrix

Features	BSF	NEF	NRF	NSSF	Policy	SCP	SEPP	UDR
ATS API	Yes	No	Yes	No	Yes	No	No	Yes
ATS Feature Activation and Deactivation	Yes	Yes	No	No	Yes	No	Yes	Yes
ATS GUI Enhancements	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ATS Health Check	Yes	No	No	Yes	Yes	No	Yes	Yes
ATS Jenkins Job Queue	Yes	Yes	No	No	Yes	Yes	Yes	Yes
Application Log Collection	Yes	Yes	No	No	Yes	Yes	Yes	Yes
ATS Maintenance Scripts	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ATS System Name and Version Display on Jenkins GUI	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
ATS Tagging Support	No	Yes	No	No	Yes	No	No	No
Custom Folder Implementation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Individual Stage Group Selection	Yes	Partially compliant (functionality not supported)	No	No	Yes	Yes	Yes	No
Lightweight Performance	No	No	No	No	No	No	No	Yes
Managing Final Summary Report, Build Color, and Application Log	Yes	Yes	Partially compliant (Application Log is not supported.)	Yes	Yes	Yes	Yes	Yes
Modifying Login Password	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

Table 2-1 (Cont.) ATS Framework Features Compliance Matrix

Features	BSF	NEF	NRF	NSSF	Policy	SCP	SEPP	UDR
Multiselection Capability for Features and Scenarios	Yes	Yes	Partially compliant (Feature level selection is supported)	No	Yes	Partially compliant (Feature level selection is supported)	Partially compliant (Feature level selection is supported)	Partially compliant (Feature level selection is supported)
Parallel Test Execution	Partially compliant (Parallel Test Execution Framework integrated, but only supports sequential execution)	Partially compliant (Parallel Test Execution Framework integrated, but only supports sequential execution)	No	Partially compliant (Parallel Test Execution Framework integrated, but only supports sequential execution)	Yes	Yes	Partially compliant (Parallel Test Execution Framework integrated, but ATS test cases need to be organized to utilize the parallel execution)	Yes (UDR, SLF, and EIR)
Parameterization	Yes	Yes	Partially compliant (Supports only new features)	No	Yes	Yes	Yes	Yes
PCAP Log Collection	No	No	No	No	No	Yes	Yes	Yes
Persistent Volume	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Single Click Job Creation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Support for ATS Deployment in OCI	No	No	No	No	No	Yes	Yes	No
Support for Transport Layer Security	No	Yes	No	No	No	Yes	Yes	No
Test Result Analyzer	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Test Case mapping and Count	Yes	Yes	No	No	Yes	Yes	Yes	Yes

2.1 ATS API

The Application Programming Interface (API) feature provides APIs to perform routine ATS tasks as follows:

- Start: To initiate one of the three test suites, such as Regression, New Features, or Performance.
- Monitor: To obtain the progress of a test suite's execution.
- Stop: To cancel an active test suite.
- Get Artifacts: To retrieve the JUNIT format XML test result files for a completed test suite.

For more information about configuring the tasks, see [Use the RESTful Interfaces](#).

2.1.1 Generating an API Token for a User

An API token that has to be generated for the user to perform routine ATS tasks using the Restful Interfaces API. Any API call requires the use of an API token for authentication. You can generate the API token, and it works until it is revoked or deleted.

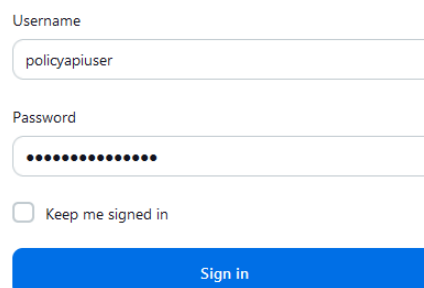
Perform the following procedure to generate an API token for a user:

1. Log in to Jenkins as an NF API user to generate an API token.

Figure 2-1 ATS Login Page

Oracle Communications Cloud Native Core - Automated Test Suite

Welcome to ATS!



Username

policyapiuser

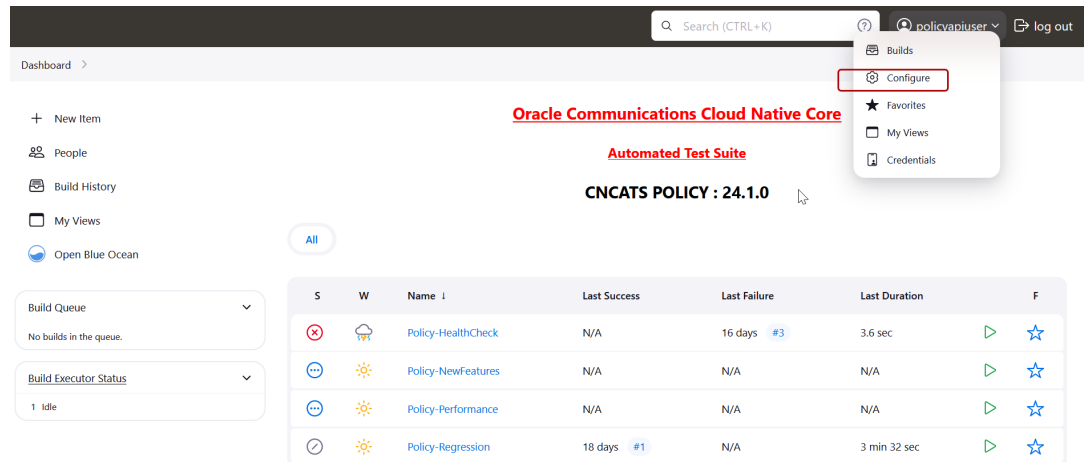
Password

.....

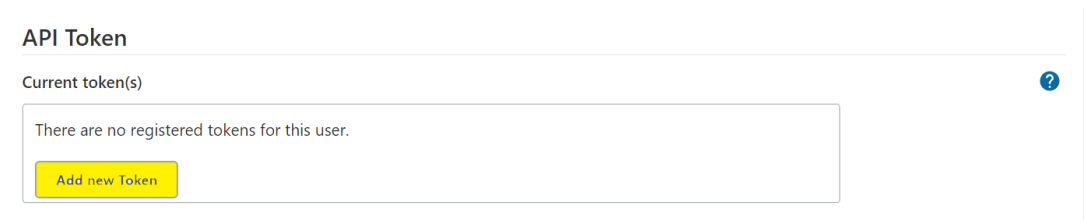
☐ Keep me signed in

Sign in

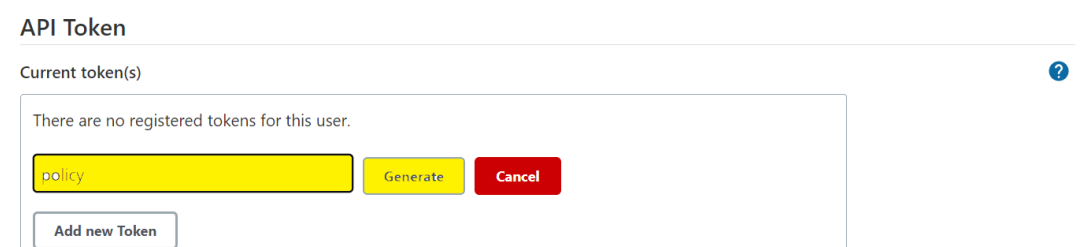
2. Click user name from the drop-down list at the top right of the Jenkins GUI, and then click **Configure**.

Figure 2-2 Configure to Add Token

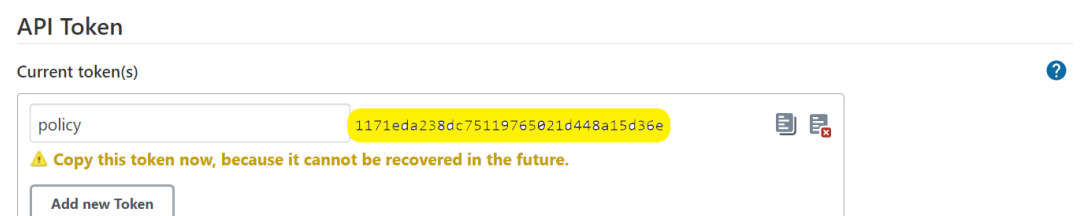
3. In the **API Token** section, click **Add New Token**.

Figure 2-3 Add New Token

4. Enter a suitable name for the token, such as **policy**, and then click **Generate**.

Figure 2-4 Generate Token

5. Copy and save the generated token.
You cannot retrieve the token after closing the prompt.

Figure 2-5 Save Generated Token

6. Click **Save**.
An API token is generated and can be used for starting, monitoring, and stopping a job using the REST API.

2.1.2 Use the RESTful Interfaces

This section provides an overview of each RESTful interface.

2.1.2.1 Configuring Host

The host to access the ATS GUI will remain the same in non-OCI setup.

For OCI Setup

The following two ways are supported to access the ATS API in OCI:

Using Loadbalancer IP

1. Add proper Ingress/Egress security rules for ATS API port (5001) and ATS service nodeport corresponding to ATS API port in loadbalancer (nf_lb_subnet) and node subnet (nf_node_subnet). To add ingress and egress security rules, see the [Adding Ingress and Egress Rules to Access the OCI Console](#) section.
2. If this step is not performed to access GUI, then insert the following annotations under the **Metadata** section to assign an external IP (Loadbalancer IP):

```
oci-network-load-balancer.oraclecloud.com/security-list-management-mode:
None
oci.oraclecloud.com/load-balancer-type: nlb
```

3. Edit and save the ATS service after ATS deployment:
For example,

```
kubectl edit svc ats-service-name -n
ats-namespace
```

4. Access the GUI using URL: <http/https>://<Loadbalancer IP>:5001

Note

The assignment of Loadbalancer IP to the ATS service is subject to availability. If the Loadbalancer IP is not assigned to the ATS service even after applying the required annotations, try to debug on the OCI side.

Using Tunneling

1. Add an ingress security rule for the node subnet (nf_node_subnet) to allow TCP traffic on all ports from the operator subnet. To add ingress and egress security rules, see the [Adding Ingress and Egress Rules to Access the OCI Console](#) section.
2. Run the following ssh tunneling command from a bash terminal on your local PC:

```
ssh -f -N -i <operator instance private key> -o StrictHostKeyChecking=no -
o ProxyCommand="ssh -i <bastion private key> -o StrictHostKeyChecking=no -
W %h:%p <bastion username>@<bastion IP>" <operator instance
```



```
username>@<operator instance IP> -L <desired system port>:<Worker Node
IP>:<ATS API NodePort> -o ServerAliveInterval=60 -o ServerAliveCountMax=300
```

```
For example,ssh -f -N -i id_rsa -o StrictHostKeyChecking=no -o
ProxyCommand="ssh -i id_rsa -o StrictHostKeyChecking=no -W %h:%p
opc@129.287.66.123" opc@10.1.76.7 -L 5009:10.9.60.118:32018 -o
ServerAliveInterval=60 -o ServerAliveCountMax=300
```

Here, ATS GUI URL is `http://localhost:5009`

TROUBLESHOOTING

If the ATS API returns an error stating "Network is unreachable," ensure that there is a proper ingress security rule in the loadbalancer subnet (nf_lb_subnet) allowing traffic from the system where the ATS API is being utilized.

2.1.2.2 Starting Jobs

To start a job, use the following RESTful interfaces:

- Default Jenkins API: The default Jenkins API to start a pipeline job
- Custom API: To start a job forcibly

Starting a Job using Default Jenkins API

If any other job is already running, the job started with this API goes to Jenkins' job queue.

- Run the following command to start a job (Default Jenkins method):

```
curl --request POST <Jenkins_host_port>/job/<Pipeline_name>/
buildWithParameters -user
<username>:<API_token> --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST <Jenkins_host_port>/job/<Pipeline_name>/
buildWithParameters --user <username>:<API_token> --verbose --cacert
<path_to_root_certificate>
```

The details of the parameters for the API are as follows:

Table 2-2 API Parameters

Parameters	Mandatory	Default Value	Description
userName	YES	NA	This parameter indicates the name of API user.
token	YES	NA	This parameter indicates the API token for API user.
Startjob_host_port	YES	NA	This parameter's format is <host>:<port> – <host> will be same as Jenkins host – <port> will be different(5001 or its nodeport)

Table 2-2 (Cont.) API Parameters

Parameters	Mandatory	Default Value	Description
pipelineName	YES	NA	This parameter indicates the name of the pipeline for which build is to be triggered.
pageAndQuery	YES	NA	This parameter can have two values: – buildWithParameters: for parametrized pipelines – build: for non-parametrized pipelines
jenkins_wait_time	NO	5	This parameter indicates the wait time for Jenkins in seconds. – If Jenkins is very slow in responding and the API response is not as expected, this wait time can be increased. – It is required when multiple running builds must be aborted before starting a new API build.
jenkins_host_port	YES	NA	This parameter's format is <Jenkins host>:<Jenkins port>.

For example,

```
curl --request POST http://10.123.154.163:30427/job/Policy-NewFeatures/
buildWithParameters
--user policyapiuser:111ad02d7471cec9ca689696e9c7a55c62 --verbose
```

When the atsGuiTLSEnabled parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/
buildWithParameters --user
scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
caroot.cer
```

Starting a Job Forcibly using Custom API

If another job is already running and has not been started by an API user, the running job is aborted, along with all other jobs in the queue that have not been started by an API user, and a new job is started.

If the running job is started by the API user, the new job does not start, and the start job request fails, returning a message in response: Build <job_id> of pipeline <pipeline_name> is already running, triggered by an API user.

Builds are aborted gracefully by a forceful API, such as when a running scenario completes its execution and cleanup before the corresponding build is aborted.

The forceful API now returns an aborted-builds parameter in response, which contains job IDs for all the aborted builds. It also returns a parameter called cancelled_builds_in_queue, which contains queue IDs for all the builds aborted in queue.

If a job ID is assigned to a build in queue, it contains a list of two values: [queueid, jobid] rather than just the queue ID.

Run the following command to start a job forcibly:

```
curl -s --request POST <Startjob_host_port>/build -H "Content-Type: application/json" -d '{"jenkins_host_port": "<Jenkins_host_port>", "pipelineName": "<Pipeline_name>", "pageAndQuery": "<pageAndQuery>"}' --user <NF_apiuser>:<token> --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl -s --request POST <Startjob_host_port>/build -H "Content-Type: application/json" -d '{"jenkins_host_port": "<Jenkins_host_port>", "pipelineName": "<Pipeline_name>", "pageAndQuery": "<pageAndQuery>"}' --user <NF_apiuser>:<token> --verbose --cacert <path_to_root_certificate>
```

For example,

```
curl --request POST http://10.75.217.25:31170/build -H "Content-Type: application/json" -d '{"jenkins_host_port": "10.75.217.25:30331", "pipelineName": "Policy-Regression", "pageAndQuery": "buildWithParameters"}' --user policyapiuser:11c1a628f808972c846c510151afa13ba2 --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30170/build -H "Content-Type: application/json" -d '{"jenkins_host_port": "10.75.217.25:30331", "pipelineName": "Policy-Regression", "pageAndQuery": "buildWithParameters"}' --user policyapiuser:11c1a628f808972c846c510151afa13ba2 --verbose
```

Note

Make sure the HTTPs protocol is given for the curl request.

Note

`Startjob_host_port` has the same IP as `jenkins_host_port`, but its port is different (the StartAPI port). ATS has two pairs of ports:

- Jenkins port (8080) with its nodeport
- StartAPI port (5001) with its nodeport

Customizing Job Parameters

Both of the Start APIs start the pipeline job with default parameter values. You can provide a different value for a parameter in an API call, such as `paramx=valuex`.

1. Append `paramx=value` to `buildWithParameters?`.

Example 1,

```
curl --request POST 10.75.217.40:31378/job/Policy-NewFeatures/
buildWithParameters?paramx=valuex --user
policyapiuser:110ed65222b9e63445689314998ff8c3bk -- verbose
```

Example 2,

```
curl --request POST 10.75.217.4:32476/build -H "Content-Type: application/
json" -d '{"jenkins_host_port": "10.75.217.40:31378",
"pipelineName": "Policy-NewFeatures", "pageAndQuery":
"buildWithParameters?paramx=valuex" }' --user <NF_apiuser>:<token>
--verbose
```

2. To add more than 1 parameter, such as `paramx=valuex` and `paramy=valuey`, append the other parameters to the API call using `&`.

Example 1,

```
curl --request POST 10.75.217.40:31378/job/Policy-NewFeatures/
buildWithParameters?paramx=valuex&paramy=valuey
--user policyapiuser:110ed65222b9e63445689314998ff8c3bk -- verbose
```

Example 2,

```
curl --request POST 10.75.217.4:32476/build -H "Content-Type: application/
json" -d '{"jenkins_host_port": "10.75.217.40:31378", "pipelineName":
"Policy-NewFeatures", "pageAndQuery": "buildWithParameters?
paramx=valuex&paramy=valuey" }' --user <NF_apiuser>:<token> --verbose
```

3. Replace `buildWithParameters?` with `build` for non-parametrized pipeline jobs.
4. Start the pipeline by using the default Jenkins API or by changing the `pageAndQuery` parameter's value to build in the following way:

```
curl --request POST <Jenkins_host_port>/job/<Pipeline_name>/build --user
<username>:<API_token> --verbose
```

Example 1,

```
curl --request POST 10.75.217.40:31378/job/Policy-NewFeatures/build --user
policyapiuser:110ed65222b9e63445689314998ff8c3bk --
verbose
```

Example 2,

```
curl --request POST 10.75.217.4:32476/build -H "Content-Type: application/
json" -d '{"jenkins_host_port": "10.75.217.40:31378",
```

```
"pipelineName": "Policy-NewFeatures", "pageAndQuery": "build" }'--
user <NF_apiuser>:<token> -verbose
```

2.1.2.3 Monitoring Jobs

This Default Jenkins API is used to monitor the progress of the job that was started.

For monitoring, the following APIs are used:

- A qid is obtained from the Location header in the response for starting a job. The first API uses this qid to get queue status about the corresponding job, including its job_id.
- The second API uses the job_id to obtain further information about the job status.

Monitoring a Job

To monitor jobs, run the following commands in a sequence:

1. `curl --request POST <Jenkins_host_port>/queue/item/<qid>/api/json --user <username>:<API_token> --verbose`
When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST
      <Jenkins_host_port>/queue/item/<qid>/api/json --user
<username>:<API_token> --verbose --cacert
      <path_to_root_certificate>
```

For example,

```
curl --request POST http://10.123.154.163:30427/queue/item/5/api/json--
user policyapiuser:111ad02d7471cec9ca689696e9c7a55c62
      --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/queue/item/27/api/json --user
      scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
      caroot.cer
```

2. `curl --request POST <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/api/json --user <username>:<API_token> --verbose`
When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST
      <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/api/json --user
<username>:<API_token> --verbose
      --cacert<path_to_root_certificate>
```

For example,

```
curl --request POST http://10.123.154.163:30427/job/Policy-
NewFeatures/3/api/json--user
policyapiuser:111ad02d7471cec9ca689696e9c7a55c62
      --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/2/api/
json --user
      scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
      caroot.cer
```

The following screenshot shows an example of monitoring the progress of the job:

Figure 2-6 Monitoring a Job

```

root@blurr5-bastion-1 ~# curl --request POST http://10.123.154.163:30427/job/Policy-NewFeatures/3/api/json --user policyapiuser:11lad0d27471c9c9ca589696
e9c7a55c62 --verbose
* Uses proxy env variable no_proxy == 'blurr5-bastion-1,blurr5-bastion-1,localhost,'
* Trying 10.123.154.163...
* TCP_NODELAY set
* Connected to 10.123.154.163 (10.123.154.163) port 30427 (#0)
* Server aut using Basic with user 'policyapiuser'
* POST /job/Policy-NewFeatures/3/api/json HTTP/1.1
* Host: 10.123.154.163:30427
* Authorization: Basic cG9saWNSYXpjdXNlcjoxMTFhZDZhc0ZlZWMSY2E2ODk2OTZlOWM3MTU1YzYy
* User-Agent: curl/7.61.1
* Accept: */*
<
< HTTP/1.1 200 OK
< Date: Tue, 08 Nov 2022 10:08:29 GMT
< X-Content-Type-Options: nosniff
< X-Jenkins: 2.363
< X-Jenkins-Session: f969f471
< X-Frame-Options: deny
< Content-Type: application/json;charset=utf-8
< Content-Length: 1875
< Server: Jetty(10.0.11)
{
  "class": "org.jenkinsci.plugins.workflow.job.WorkflowRun", "actions": [{"class": "hudson.model.ParametersAction", "parameters": [{"class": "hudson.model.StringParameterValue", "name": "...", "value": "..."}, {"class": "hudson.model.StringParameterValue", "name": "TestSuite", "value": "..."}, {"class": "hudson.model.StringParameterValue", "name": "SUT", "value": "PCF"}, {"class": "hudson.model.StringParameterValue", "name": "Configuration Type", "value": "Product Config"}, {"class": "hudson.model.StringParameterValue", "name": "Select Option", "value": "All"}, {"class": "hudson.model.StringParameterValue", "name": "Features", "value": "..."}, {"class": "hudson.model.StringParameterValue", "name": "TestCases", "value": "..."}, {"class": "hudson.model.StringParameterValue", "name": "Fetch Log Upon Failure", "value": "YES"}]}, {"class": "hudson.model.CauseAction", "causes": [{"class": "hudson.model.CauseUserIdCause", "shortDescription": "Started by user policyapiuser", "userId": "policyapiuser", "org.jenkinsci.plugins.workflow.libs.LibraryReference": "policyapiuser:11lad0d27471c9c9ca589696:1"}, {"class": "org.jenkinsci.plugins.workflow.cps.EnvActionImpl", "env": "Environment", "org.jenkinsci.plugins.displayurlapi.actions.RunDisplayAction": "Run"}, {"class": "org.jenkinsci.plugins.pipeline.modeldefinition.actions.RestartDeclarativePipelineAction", "pipeline": "Policy-NewFeatures/3", "org.jenkinsci.plugins.workflow.views.FlowGraphAction": "FlowGraph", "artifacts": [{"building": true, "description": null, "displayName": "#3", "duration": "0", "estimatedDuration": "1", "executor": "Policy-NewFeatures #3", "fullDisplayName": "Policy-NewFeatures #3", "id": "3", "keepLog": false, "number": "3", "queueId": "5", "result": null, "timestamp": "1667981546337", "url": "http://10.123.154.163:30427/job/Policy-NewFeatures/3/"}], "culprits": [{"inProgress": true, "nextBuild": null, "previousBuild": null}]}]}

```

2.1.2.4 Stopping Jobs

Stop API is used to stop the currently running job using its `job_id`. It is also a default Jenkins API.

Stopping a Job

For ATS without Parallel Test Execution framework integrated:

```
curl --request POST
      <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/stop --user
<username>:<API_token>
      --verbose
```

When the `atsGuiTLSEnabled` parameter is set to `true`:

```
curl --request POST
      <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/stop --user
<username>:<API_token> --verbose -cacert
      <path to root certificate>
```

For example,

```
curl --request POST http://10.75.217.4:31881/job/UDR-Regression/21/stop --user udrapiuser:1139a72213e0a686972cbff4a2f9333a9f --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/2/stop --
user
    scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
caroot.cer
```

Note

- If the rerun count is greater than zero, the job must be stopped twice.
- This Stop API call does not abort the build gracefully.

For ATS with Parallel Test Execution framework integrated:

```
curl --request POST
    <Stopjob_host_port>/job/<Pipeline_name>/<job_id>/stop --user
<username>:<API_token>
    --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST
    <Stopjob_host_port>/job/<Pipeline_name>/<job_id>/stop --user
<username>:<API_token> --verbose --cacert
    <path_to_root_certificate>
```

For example.

```
curl --request POST http://10.75.217.4:32476/job/UDR-Regression/21/stop --user
    udrapiuser:1139a72213e0a686972cbff4a2f9333a9f --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30170/job/SCP-Regression/2/stop --
user
    scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
caroot.cer
```

The following table lists the parameter details for Stop API:

Table 2-3 Stop API Details

Parameter	Mandatory	Default Value	Description
userName	Yes	NA	Name of API user
token	Yes	NA	The API token for the API user

Table 2-3 (Cont.) Stop API Details

Parameter	Mandatory	Default Value	Description
Stopjob_host_port	Yes	NA	Format is <host>:<port> <host> is same as Jenkins host <port> is different such as 5001 or its nodeport
pipelineName	Yes	NA	Name of the pipeline for which build is to be stopped
immediate	No	False	To stop the build immediately, send a query parameter ("immediate=true") with API call. For example, <pre>curl --request POST</pre> <pre><Stopjob_host_port>/j</pre> <pre>ob/<Pipeline_name>/</pre> <pre><job_id>/stop?</pre> <pre>immediate=true --user</pre> <pre><username>:<API_token</pre> <pre>> --verbose</pre> <code>immediate</code> can also have values such as <code>yes</code> or <code>1</code>. These values work similar to the <code>true</code> value.

2.1.2.5 Getting Test Suite Artifacts

Default Jenkins API is used to get the JUNIT-formatted XML test result files for a completed test suite.

For getting artifacts for any completed test suite, the following APIs are used:

- For getting an overall build summary
- For getting a JUNIT XML test result file for every feature file that ran

For getting an overall build summary:

```
curl --request POST <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/
testReport/api/xml?exclude=testResult/suite --user <username>:<API_token> --
verbose
```

When the `atsGuiTLSEnabled` parameter is set to `true`:

```
curl --request POST
    <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/testReport/api/xml?
exclude=testResult/suite --user
    <username>:<API_token> --verbose --cacert <path_to_root_certificate>
```


For example,

```
curl --request POST http://10.123.154.163:30427/job/Policy-NewFeatures/4/
testReport/api/xml?exclude=testResult/suite --user
policyapiuser:111ad02d7471cec9ca689696e9c7a55c62 --verbose
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/1/
testReport/api/xml?exclude=testResult/suite
--user scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
caroot.cer
```

For getting Feature-wise XML, `Select_Option = All`:

```
curl --request POST
    <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/artifact/test-results/
reports/*zip*/test-results.zip
--user <username>:<API_token> --verbose --output test-results.zip
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST
    <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/artifact/test-results/
reports/*zip*/test-results.zip
--user <username>:<API_token> --verbose --output test-results.zip --
cacert
    <path_to_root_certificate>
```

For example,

```
curl --request POST http://
    10.75.217.4:31881/job/Policy-NewFeatures/21/artifact/test-results/
reports/*zip*/test-results.zip --user
policyapiuser:11c3344996c4fda01ded2124bec4f9aa17
--verbose --output test-results.zip
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/1/artifact/
test-results/reports/*zip*/test-results.zip
--user scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
caroot.cer
```

For getting Feature-wise XML, `Select_Option = Single/MultipleFeatures`:

```
curl --request POST
    <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/artifact/test-results/
reports/
    *.<Feature1_name>.xml,*.<Feature2_name>.xml/*zip*/test-results.zip --
```

```
user <username>:<API_token>
--verbose --output test-results.zip
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST
    <Jenkins_host_port>/job/<Pipeline_name>/<job_id>/artifact/test-results/
reports/
    *.<Feature1_name>.xml, *.<Feature2_name>.xml/*zip*/test-results.zip --
user <username>:<API_token>
--verbose --output test-results.zip --cacert
    <path_to_root_certificate>
```

For example,

```
curl --request POST http://
    10.75.217.4:31881/job/Policy-NewFeatures/21/artifact/test-results/
reports/*.goldenfeature.xml, *.AMPolicy.xml/*zip*/test-results.zip
    -user policyapiuser:11c3344996c4fda01ded2124bec4f9aa17 --verbose --
output test-results.zip
```

When the `atsGuiTLSEnabled` parameter is set to true:

```
curl --request POST https://10.75.217.25:30301/job/SCP-Regression/1/artifact/
test-results/reports/*.SCP_Registration_With_PLMNList.xml/*zip*/test-
results.zip
    --user scpapiuser:11c2fde49cea6eb8f332ad23a7877ea2de --verbose --cacert
    caroot.cer
```

API calls for `Select_Option = All` and `Select_Option = Single/MultipleFeatures` return a zip file with JUNIT XMLs, one XML for each feature.

Figure 2-7 Sample XML Output for AMPolicy.feature

```
<testsuite name="features.regression.templ.AMPolicy.AMPolicy.This feature is to refactor and/or redesign the Access and Mobility Management Service of PCF an
d to test Policy Establishment/Termination for AMP" tests="3" errors="0" failures="3" skipped="0" time="51.84246" timestamp="2022-03-24T07:11:12.693986" host
name="ocasts-ocasts-policy-85865d8d4c-d1c42"><properties><property name="vendor.name" value="Oracle" /><property name="vendor.product" value="Automated Test Su
ite" /></properties><testcase classname="features.regression.templ.AMPolicy.AMPolicy.This feature is to refactor and/or redesign the Access and Mobility Mana
gement Service of PCF and to test Policy Establishment/Termination for AMP" name="AM_Policy_Create_Query_UDM_Subscribe_True" status="failed" time="17.251319"
><failure type="AssertionError" message="Maximum number of retries exhausted!!!">
<![CDATA[
Failing step: Then Cleanup before test run ... failed in 12.130s
location: features/regression/templ/AMPolicy/AMPolicy.Feature:23
Assertion Failed: Maximum number of retries exhausted!!!
]]>
</failure>
</testsuite>
```

In the API call, specify other selected features in comma-separated form as /
*<Feature1_name>.xml, *<Feature2_name>.xml, *<Feature3_name>.xml, *<Feature4_name>.x
ml/ for `Select_Option = Single/MultipleFeatures`.

The API call for getting the overall build summary returns an XML with values for duration, failCount, passCount, and skipCount for the current build.

Figure 2-8 Sample Output

```
▼<testResult _class="hudson.tasks.junit.TestResult">
  <duration>61.424786</duration>
  <empty>false</empty>
  <failCount>12</failCount>
  <passCount>0</passCount>
  <skipCount>294</skipCount>
</testResult>
```

It is recommended to maintain a gap of at least a few seconds between two API calls. This gap depends on the time Jenkins takes to complete the API request.

2.2 ATS Feature Activation and Deactivation

ATS Feature Activation and Deactivation feature allows users to activate or deactivate specific features within the ATS using Helm charts.

Note

Once these features are removed, they cannot be reinstated in the deployed ATS. However, users have the option to reinstall the ATS to restore the disabled features.

The following table lists the features to enable or disable using Helm charts.

Note

These parameters can be edited in the ATS deployment file (values.yaml).

Table 2-4 Enable or Disable ATS Feature

Features	Parameter	Description
Support for Test Case Mapping and Count	testCaseMapping	Set this parameter to true to activate the feature in the ATS GUI.
Application Log Collection and PCAP Log Collection	logging	Set this parameter to true to collect the ATS logs. If the parameter is set to false, the logs will not be collected.
Lightweight Performance	lightWeightPerformance	Set this parameter to true to activate the feature in the ATS GUI. If the parameter is set to false, the performance pipeline will not be accessible.

Table 2-4 (Cont.) Enable or Disable ATS Feature

Features	Parameter	Description
ATS Health Check	healthcheck	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If the parameter is set to <code>false</code> , the health check pipeline will not be accessible.
ATS API	atsApi	Set this parameter to <code>true</code> to activate the feature. If the parameter is set to <code>false</code> , the ATS API feature on the 5001 port will be disabled.
Parameterization	parameterization	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If the parameter is set to <code>false</code> , the <code>Configuration_Type</code> parameter on the GUI will not be available.
Parallel Test Execution	parallelFrameworkChangesIntegrated	Set this parameter to <code>true</code> if all parallel test case execution features are picked by NF and changes are made to files. Note: Do not change the values provided in the <code>values.yaml</code> file for this parameter.
Parallel Test Execution	parallelTestCaseExecution	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , all the features will be copied into a single stage or group, resulting in sequential execution. Note: It is not advisable to edit the default value given in the <code>values.yaml</code> file for this parameter.
Parallel Test Execution	mergedExecution	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , the option to include other pipelines for <code>mergedExecution</code> will not be available. Note: It is not advisable to edit the default value given in the <code>values.yaml</code> file for this parameter.
ATS Support to Execute Scenarios	scenarioSelection	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , the ability to select single or multiple scenarios will be removed. It is dependent on the <code>testCaseMapping</code> parameter. If this parameter is set to <code>false</code> , the <code>scenarioSelection</code> parameter will be set to <code>false</code> too. Note: It is not advisable to edit the default value given in the <code>values.yaml</code> file for this parameter.

Table 2-4 (Cont.) Enable or Disable ATS Feature

Features	Parameter	Description
ATS Tagging Support	executionWithTagging	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , the ability to execute test cases based on tags will be removed. It is dependent on the <code>testCaseMapping</code> parameter. If this parameter is set to <code>false</code> , the <code>executionWithTagging</code> parameter will be set to <code>false</code> too.
Stage or Group Level Execution	individualStageGroupSelection	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , the ability to select all test cases from individual stages or groups using a single checkbox will be removed. Note: It is not advisable to edit the default value given in the <code>values.yaml</code> file for this parameter.
Support for Transport Layer Security	atsGUI TLSEnabled	Set this parameter to <code>true</code> to activate the feature in the ATS GUI. If set to <code>false</code> , ATS will work in HTTP mode.

Note

- You can edit the parameters relating to the features that NF supports. Keep the default value for the remaining parameters.
- For the current release, `mergedExecution`, `individualStageGroupSelection`, and `parallelTestCaseExecution` parameters value should not be modified.

2.3 ATS GUI Enhancements

With this enhancement, the ATS Graphical User Interface (GUI) layout is redesigned to streamline user interaction by segregating execution options and features or test cases into distinct compartments, allowing the users to navigate effortlessly between different functionalities.

Figure 2-9 Layout Enhancement

Pipeline UDR-Regression

This build requires parameters:

Oracle Communications Cloud Native Core
Automated Test Suite - UDR

EXECUTION OPTIONS

TestSuite Regression	SUT UDR	SUTSuite All	Configuration_Type Product_Config
Include_NewFeatures NO	Fetch_Log_Upon_Failure NO	Log_Type For PcapLog collection debug container must be running ☒ Not Applicable	Log_Level Not_Applicable

FEATURES AND TESTCASES

Features	All
TestCases	All

Test cases now feature a display of fixed length, ensuring that even lengthier test cases remain visually manageable. Additionally, the inclusion of hover-scroll functionality allows for easy access to content beyond the visible area, enhancing readability and user friendliness.

Figure 2-10 Test Cases Visibility

TestCases All

Total Features are: 7 & TestCases are: 78

Stage: Stage_1 Group: bulk_import_metrics_autocreate_true

UDR_Bulk_Import

- pre_condition
- bulk_import_create_modify_delete_msidn
- bulk_import_create_modify_delete_extid
- bulk_import_error_subscriber_already_exist
- disable_ondemand
- bulk_import_create_modify_delete_nai
- bulk_import_error_delete_imsi_subscriber_not_found

UDR_Bulk_Import_PolicyAndVsa

- pre_condition
- bulk_import_CreateVSADData_ModifyPCF

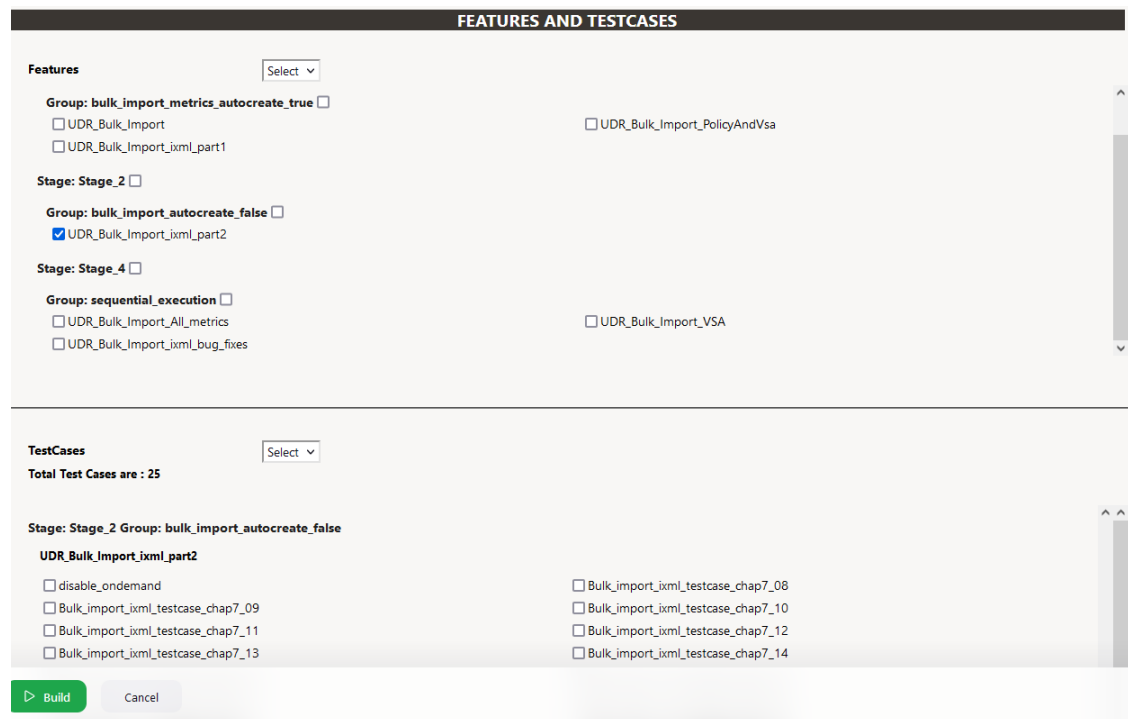
UDR_Bulk_Import_ixml_part1

- disable_ondemand
- Bulk_import_ixml_testcase_05
- Bulk_import_ixml_testcase_chap5_err_01
- Bulk_import_ixml_testcase_chap5_err_03
- Bulk_import_ixml_testcase_chap5_err_05
- Bulk_import_ixml_testcase_04
- Bulk_import_ixml_testcase_06
- Bulk_import_ixml_testcase_chap5_err_02
- Bulk_import_ixml_testcase_chap5_err_04
- Bulk_import_ixml_testcase_chap5_err_06

Tooltips have been introduced to allow user to have quick insights into the grouping and staging of individual test cases, thereby improving overall understanding and navigation within the ATS environment.

Figure 2-11 Tooltips

To facilitate the use of multiselect features and scenarios, distinct drop down menus have been introduced. This enables users to efficiently choose and manage multiple options simultaneously, thereby improving flexibility and productivity in selecting test cases.

Figure 2-12 Multiselect Features and Scenarios

2.4 ATS Health Check

ATS Health Check functionality is to check the health of the System Under Test (SUT)

Earlier, ATS used Helm test functionality to check the health of the System Under Test (SUT). With the implementation of the ATS Health Check pipeline, the SUT health check process has

been automated. ATS health checks can be performed on webscale and non-webscale environments.

Convert a Value in Base64

The following command can be utilized to convert any value into base64 encoding:

```
echo -n "value" | base64
```

For example,

```
echo -n "126.98.76.43" | base64
```

Deploying ATS Health Check in a Webscale Environment

- Set the Webscale to 'true' and the following parameters by encoding them with [base64](#) in the ATS values.yaml file:
- Set the following parameter to encrypted data:

```
webscalejumpserverip: encrypted-data
webscalejumpserverusername: encrypted-data
webscalejumpserverpassword: encrypted-data
webscaleprojectname: encrypted-data
webscalelabserverFQDN: encrypted-data
webscalelabserverport: encrypted-data
webscalelabserverusername: encrypted-data
webscalelabserverpassword: encrypted-data
```

Encrypted data is the value of parameters encrypted in base64. Fundamentally, Base64 is used to encode the parameters.

For example:

```
webscalejumpserverip=$(echo -n '10.75.217.42' | base64), Where Webscale Jump
server ip needs to be provided
webscalejumpserverusername=$(echo -n 'cloud-user' | base64), Where Webscale
Jump server Username needs to be provided
webscalejumpserverpassword=$(echo -n '*****' | base64), Where Webscale Jump
server Password needs to be provided
webscaleprojectname=$(echo -n '*****' | base64), Where Webscale Project Name
needs to be provided
webscalelabserverFQDN=$(echo -n '*****' | base64), Where Webscale Lab Server
FQDN needs to be provided
webscalelabserverport=$(echo -n '*****' | base64), Where Webscale Lab Server
Portneeds to be provided
webscalelabserverusername=$(echo -n '*****' | base64), Where Webscale Lab
Server Username needs to be provided
webscalelabserverpassword=$(echo -n '*****' | base64), Where Webscale Lab
Server Password needs to be provided
```

Running ATS Health Check Pipeline in an Webscale Environment

To run ATS Health Check pipeline:

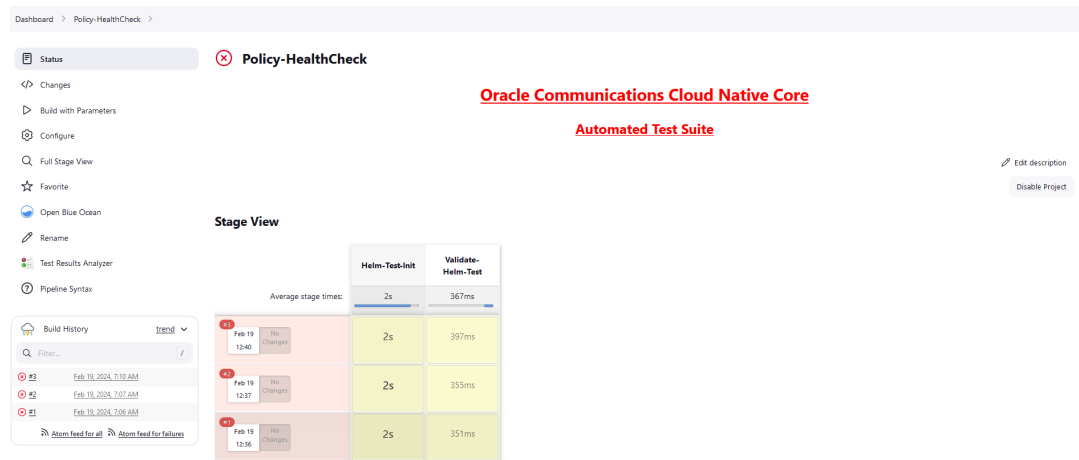
1. Log in to ATS using respective <NF> login credentials.

- Click <NF>HealthCheck pipeline and then click **Configure**.

Note

<NF> denotes the network function. For example, in Policy, it is called as Policy-HealthCheck pipeline.

Figure 2-13 Configure Healthcheck



- Provide parameter a with Helm release name deployed. If there are multiple releases, use comma to provide all Helm release names.

//a = helm releases [Provide Release Name with Comma Separated if more than 1]

Provide parameter c with the appropriate Helm command, such as helm, helm3, or helm2.

//c = helm command name [helm or helm2 or helm3]

Figure 2-14 Save the Changes

Pipeline script

Script ?

```
1 node{
2   def myVar = 'initial_value'
3   def buildVar = 'initial_value'
4   properties(
5     [
6       parameters(
7         [string(defaultValue: '', name: 'Helm_releases',description: "Provide Rel
8           string(defaultValue: '', name: 'Namespace', description: "Provide the nan
9         )
10      ]
11    )
12  )
13  stage('Helm-Test-Init') {
14    catchError(buildResult: 'SUCCESS', stageResult: 'UNSTABLE') {
15
16      //a = helm releases [Provide Release Name with Comma Seperated if more than 1 ]
17      //b= Namespace, If not applicable then remove the argument
18      //c= helm command name [helm or helm2 or helm3]
```

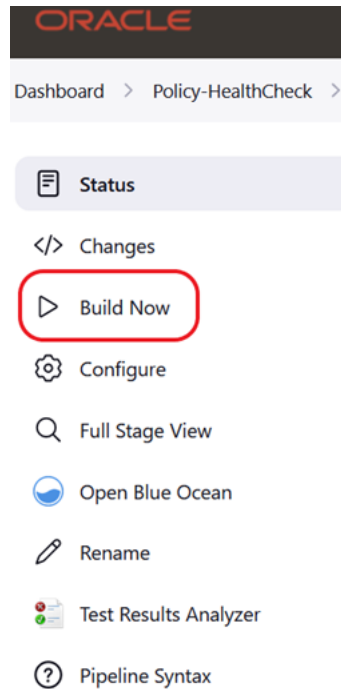
☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

4. Save the changes and click **Build Now**. ATS runs the health check on respective network function.

Figure 2-15 Build Now

Deploying ATS Health Check Pipeline in an OCI Environment

To use a ssh private key, create healthcheck-oci-secret and set the value of the key "passwordAuthenticationEnabled" to false.

Creating healthcheck-oci-secret

Create healthcheck-oci-secret to use ssh private keys instead of passwords using the following command:

```
kubectl create secret generic healthcheck-oci-secret --from-  
file=bastion_key_file='<path of bastion ssh private key file>' --from-  
file=operator_instance_key_file='<path of operator instance ssh private key  
file>' -n <ATS namespace>
```

For example,

```
kubectl create secret generic healthcheck-oci-secret --from-  
file=bastion_key_file='/tmp/bastion_private_key' --from-  
file=operator_instance_key_file='/tmp/operator_instance_private_key' -n  
seppsvc
```

Note

- Maintain the name of the secret as "healthcheck-oci-secret".
- Ensure that the '--from-file' keys retain the same names: "bastion_key_file" and "operator_instance_key_file".
- If the SSH private key is identical for both the bastion and operator instance, you can use the same path for both in the secret creation command.

Perform the following procedure to deploy ATS Health Check in a OCI environment:

Set the webscale parameter set to 'false' and following parameters by encoding it with [base64](#) in the ATS values.yaml file.

- To use password, provide [base64](#) encoded values for key "password" for both bastion and operator instances, and set the value of key **passwordAuthenticationEnabled** to "true".
- Set the following parameter to encrypted data:

```
envtype: encrypted-data
ociHealthCheck:
  passwordAuthenticationEnabled: true or false
  bastion:
    ip: encrypted-data
    username: encrypted-data
    password: encrypted-data
  operatorInstance:
    ip: encrypted-data
    username: encrypted-data
    password: encrypted-data
```

Note

All fields are mandatory except for passwords. When the "passwordAuthenticationEnabled" field is set to true, only the "password" field needs to be updated; otherwise, it can remain with its default value.

Running ATS Health Check Pipeline in an OCI Environment

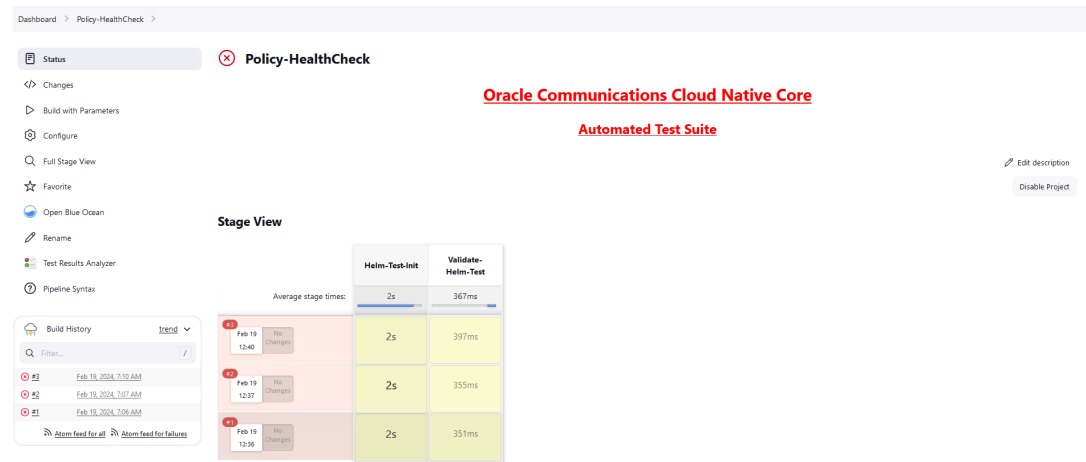
To run ATS Health Check pipeline:

1. Log in to ATS using respective <NF> login credentials.
2. Click <NF>HealthCheck pipeline and then click **Configure**.

Note

<NF> denotes the network function. For example, in Policy, it is called as Policy-HealthCheck pipeline.

Figure 2-16 Configure Healthcheck



3. Provide parameter a with Helm release name deployed. If there are multiple releases, use comma to provide all Helm release names.

//a = helm releases [Provide Release Name with Comma Separated if more than 1]

Provide parameter c with the appropriate Helm command, such as helm, helm3, or helm2.

//c = helm command name [helm or helm2 or helm3]

Figure 2-17 Save the Changes

Pipeline script

Script ?

```
1 node{
2   def myVar = 'initial_value'
3   def buildVar = 'initial_value'
4   properties(
5     [
6       parameters(
7         [string(defaultValue: '', name: 'Helm_releases',description: "Provide Rel
8           string(defaultValue: '', name: 'Namespace', description: "Provide the nan
9         )
10      ]
11    )
12  )
13  stage('Helm-Test-Init') {
14    catchError(buildResult: 'SUCCESS', stageResult: 'UNSTABLE') {
15
16      //a = helm releases [Provide Release Name with Comma Seperated if more than 1 ]
17      //b= Namespace, If not applicable then remove the argument
18      //c= helm command name [helm or helm2 or helm3]
```

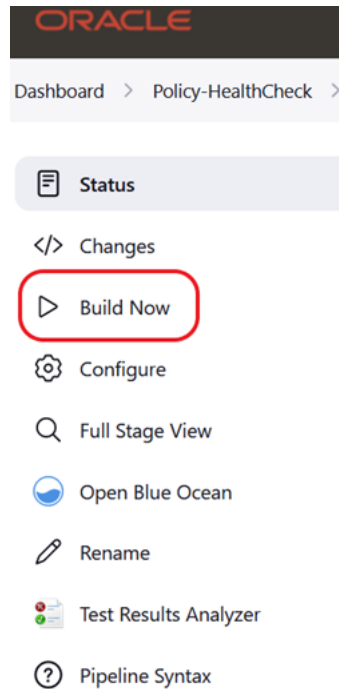
☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

4. Save the changes and click **Build Now**. ATS runs the health check on respective network function.

Figure 2-18 Build Now

Deploying ATS Health Check in a Non-Webscale or Non-OCI Environment

Perform the following procedure to deploy ATS Health Check in a non-webscale or non-OCI environment such as OCCNE:

Set the Webscale parameter set to 'false' and following parameters by encoding it with [base64](#) in the ATS values.yaml file:

```
occnehostip: encrypted-data
occnehostusername: encrypted-data
occnehostpassword: encrypted-data
```

Example:

```
occnehostip=$(echo -n '10.75.217.42' | base64) , Where occne host ip needs to
be provided
occnehostusername=$(echo -n 'cloud-user' | base64), Where occne host username
needs to be provided
occnehostpassword=$(echo -n '*****' | base64), Where password of host needs to
be provided
```

Running ATS Health Check Pipeline in a Non-Webscale or Non-OCI Environment

Perform the following procedure to run the ATS Health Check pipeline in a non-webscale or non-OCI environment such as OCCNE:

1. Log in to ATS using respective <NF> login credentials.
2. Click <NF>HealthCheck pipeline and then click **Configure**.
3. Provide parameter a with Helm release name deployed. If there are multiple releases, use comma to provide all Helm release names.

Provide parameter b with SUT deployed namespace name.

Provide parameter c with the appropriate Helm command, such as helm, helm3, or helm2.

```
//a = helm releases [Provide Release Name with Comma Separated if more
than 1 ]
```

```
//b = Namespace, If not applicable to WEBSCALE environment then remove the
argument
```

```
//c = helm command name [helm or helm2 or helm3]
```

Figure 2-19 Save the Changes

Pipeline script

Script ?

```

1 node{
2   def myVar = 'initial_value'
3   def buildVar = 'initial_value'
4   properties(
5     [
6       parameters(
7         [string(defaultValue: '', name: 'Helm_releases',description: "Provide Re]
8         string(defaultValue: '', name: 'Namespace', description: "Provide the nan
9       )
10    ]
11  )
12  stage('Helm-Test-Init') {
13    catchError(buildResult: 'SUCCESS', stageResult: 'UNSTABLE') {
14      //a = helm releases [Provide Release Name with Comma Separated if more than 1 ]
15      //b= Namespace, If not applicable then remove the argument
16      //c= helm command name [helm or helm2 or helm3]

```

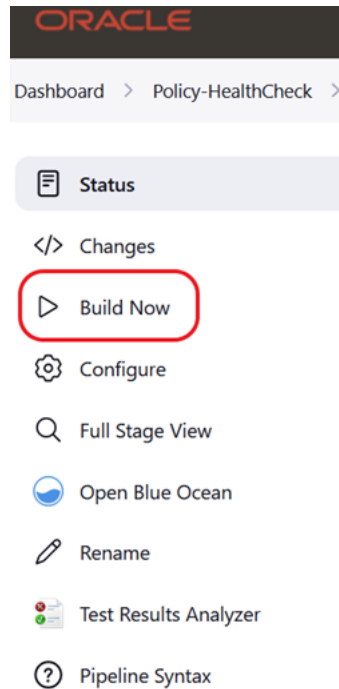
☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

4. Save the changes and click **Build Now**. ATS runs the health check on respective network function.

Figure 2-20 Build Now

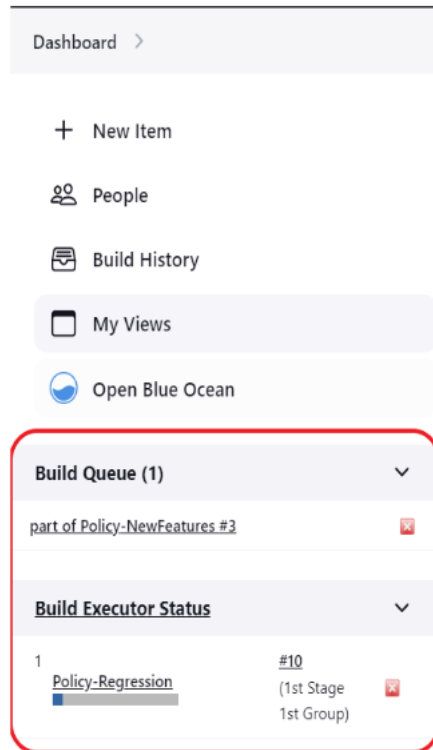
By clicking **Build Now**, you can run the health check on ATS and store the result in the console logs.

2.5 ATS Jenkins Job Queue

The ATS Jenkins Job Queue feature places the second job in a queue if the current job is already running from the same or different pipelines to prevent jobs from running in parallel to one another.

Job/build queue status can be viewed in the left navigation pane on the ATS home page. The following image shows the build queue status when a user has tried to run the NewFeatures pipeline when the Regression pipeline is already running.

Figure 2-21 Build Executor Status



2.6 Application Log Collection

Using Application Log Collection, you can debug a failed test case by collecting the application logs for NF System Under Test (SUT). Application logs are collected for the duration that the failed test case was run.

Application Log Collection can be implemented by using ElasticSearch or Kubernetes Logs. In both these implementations, logs are collected per scenario for the failed scenarios.

Application Log Collection Using ElasticSearch

To access the option to collect logs using ElasticSearch:

1. Log in to ATS using respective <NF> login credentials.
2. On the NF home page, click any new feature or regression pipeline, from where you want to collect the logs.
3. In the left navigation pane, click **Build with Parameters**.
4. Select YES or NO from the drop-down menu of **Fetch_Log_Upon_Failure** to select whether the log collection is required for a particular run.

Figure 2-22 Fetch_Log_Upon_Failure

The screenshot shows the Jenkins configuration page for a pipeline named 'SCP-Regression'. On the left sidebar, there are links for 'Favorite', 'Open Blue Ocean', 'Rename', 'Test Results Analyzer', and 'Pipeline Syntax'. Below these is a 'Build History' section showing 'No builds'. The main configuration area is divided into two sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'. In the 'EXECUTION OPTIONS' section, the 'TestSuite' is 'Regression', 'Execute_Suite' is 'R15', 'Configuration_Type' is 'Product_Config', and 'FilterWithTags' is 'NO'. The 'Include_NewFeatures' is 'NO'. The 'Fetch_Log_Upon_Failure' option is highlighted with a red box and set to 'YES'. The 'Log_Type' is 'AppLog' (checked) and 'PcapLog' (unchecked). The 'Log_Level' is 'DEFAULT'. The 'HTTPX_LOG_LEVEL' is 'Disabled'. In the 'FEATURES AND TESTCASES' section, 'Features' and 'TestCases' are both set to 'All'.

5. If option **Log_Type** is also available, select value **AppLog** for it.
6. Select the Log Level from the drop-down menu of Log_Level to set the log level for all the microservices. The possible values for Log_Level are as follows:
 - **WARN:** Designates potentially harmful situations.
 - **INFO:** Designates informational messages that highlight the progress of the application at coarse-grained level.
 - **DEBUG:** Designates fine-grained informational events that are most useful to debug an application.
 - **ERROR:** Designates error events that might still allow the application to continue running.
 - **TRACE:** The TRACE log level captures all the details about the behavior of the application. It is mostly diagnostic and is more granular and finer than DEBUG log level.

Note

Log_Level values are NF dependent.

7. After the build execution is complete, go into the ATS pod, then navigate to following path to find the applogs: `.jenkins/jobs/<Pipeline Name>/builds/<build number>/`
For example, `.jenkins/jobs/SCP-Regression/builds/5/`
Applogs is present in zip form. Unzip it to get the log files.

The following tasks are carried out in the background to collect logs:

- ElasticSearch API is used to access and fetch logs.
- Logs are fetched from ElasticSearch for the failed scenarios
- Hooks (after scenario) within the cleanup file initiate an API call to Elasticsearch to fetch Application logs.
- Duration of the failed scenario is calculated based on the time stamp and passed as a parameter to fetch the logs from ElasticSearch.
- Filtered query is used to fetch the records based on Pod name, Service name, and timestamp (Failed Scenario Duration).

- For ElasticSearch, there is no rollover or rotation of logs over time.
- The maximum records that the ElasticSearch API can fetch per microservice in a failed scenario is limited to 10K.
- The following configuration parameters are used for collecting logs using Elastic Search:
 - ELK_WAIT_TIME: Wait time to connect to Elastic Search
 - ELK_HOST: Elastic Search HostName
 - ELK_PORT: Elastic Search Port

Application Log Collection Using Kubernetes Logs

To access the option to collect logs using Kubernetes Logs:

1. On the NF home page, click any new feature or regression pipeline, from where you want to collect the logs.
2. In the left navigation pane, click **Build with Parameters**.
3. Select YES or NO from the drop-down menu of **Fetch_Log_Upon_Failure** to select whether the log collection is required for a particular run.
4. Select the Log Level from the drop-down menu of Log_Level to set the log level for all the microservices. The possible values for Log_Level are as follows:
 - WARN: Designates potentially harmful situations.
 - INFO: Designates informational messages that highlight the progress of the application at coarse-grained level.
 - DEBUG: Designates fine-grained informational events that are most useful to debug an application.
 - ERROR: Designates error events that might still allow the application to continue running.

Note

Log_Level values are NF dependent.

The following tasks are carried out in the background to collect logs:

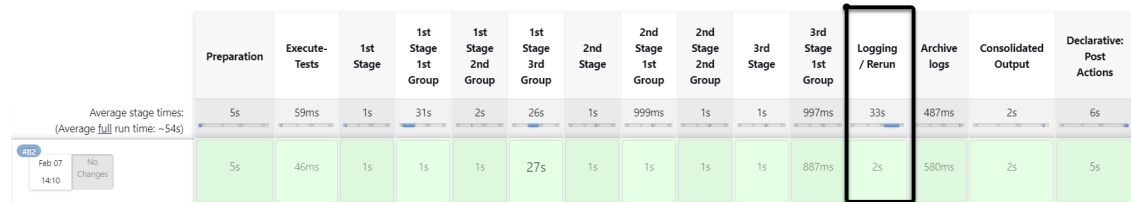
- Kube API is used to access and fetch logs.
- For failed scenarios, logs are directly fetched from microservices.
- Hooks (after scenario) within the cleanup file initiate an API call to Elasticsearch to fetch Application logs.
- The duration of the failed scenario is calculated based on the time stamp and passed as a parameter to fetch the logs from microservices.
- Logs roll can occur while fetching the logs for a failed scenario. The maximum loss of logs is confined to a single scenario.

2.6.1 Application Log Collection and Parallel Test Execution Integration

A new stage, "Logging/Rerun", has been added at the end of the Execute-Tests stage to collect rerun logs, such as applog and PCAP logs, by running the failed test cases in a sequence.

Figure 2-23 Logging/Rerun new stage

Stage View



If the `Fetch_Log_Upon_Failure` parameter is set to YES and if any test case fails in the initial run, then:

- The failed test case reruns and log collection start in the Logging/Rerun stage after the initial run is completed for all the test cases.
- The logs from the initial execution are collected, but they might be incorrect.
- Even if the rerun parameter is set to 0, the failed test case reruns in the Logging/Rerun stage and the log is collected.

Note

Not applicable for all the NFs.

- If the `Fetch_Log_Upon_Failure` parameter is set to NO and if any test case fails in the initial run, then the failed test case rerun starts in the same stage after the initial execution is over for all the test cases in its group.

2.7 ATS Maintenance Scripts

ATS maintenance scripts are used to perform the following operations:

- Taking a backup of the ATS custom folders and Jenkins pipeline.
- Viewing the configuration and restoring the Jenkins pipeline.
- Viewing the configuration and installing or uninstalling ATS and stubs.

ATS maintenance scripts are present in the ATS image at the following path: `/var/lib/jenkins/ocats_maint_scripts`

Run the following command to copy the scripts to a local system (bastion):

```
kubectl cp <NAMESPACE>/<POD_NAME>:/var/lib/jenkins/ocats_maint_scripts
<DESTINATION_PATH_ON_BASTION> pod
```

For example,

```
kubectl cp ocpcf/ocats-ocats-policy-694c589664-js267:/var/lib/Jenkins/
ocats_maint_scripts /home/meta-user/ocats_maint_scripts pod
```

2.7.1 ATS Scripts

ATS maintenance scripts are used to perform various task related to ATS and Jenkin pipeline.

The following are the types of scripts:

- **ats_backup.sh**: This script requires the user's input and takes a backup of the ATS custom folders, Jenkins jobs, and user's folders on the user's system. The backup can be of the Jenkins jobs and user's folder, the custom folders, or both. The custom folders include `cust_regression`, `cust_newfeatures`, `cust_performance`, `cust_data`, and `custom_config`. For a Jenkins job or a user's folder, the script only takes a backup of the `config.xml` file. Also, the script requires the user to store a backup on the user's system (the default path is the location from where the script is being run) and to create a backup folder on the system and take the backup of the chosen folder from the corresponding ATS into the backup folder. The backup folder name can be of the following notation: `ats_<version>_backup_<date>_<time>`.
- **ats_uninstall.sh**: This script requires the user's input and uninstalls the corresponding ATS.
- **ats_install.sh**: This script requires the user's input and installs a new ATS. If `PVEnabled` is set to `true`, the script also reads the PVC name from `values.yaml` and creates `values.yaml` before installation. Also, if needed, the script performs the postinstallation steps, such as copying tests and Jenkins jobs' folders from the `ats_data` tar file to the ATS pod when PV is deployed, and then restarts the pod.
- **ats_restore.sh**: This script requires the user's inputs, restores the new release ATS pipeline, and views the configuration by referring to the last release ATS Jenkins jobs and the user's configuration. It depends on the user whether to use the backup folders from the user's system to restore the ATS configuration. If the user instructs the script to use the backup from the system, the script requires the path of the backup and uses the backup to restore. Otherwise, the script requires the last ATS Helm release name to refer to its Jenkins jobs and the user's configuration to restore.
The script refers to the last release of ATS Jenkins pipelines and sets the `Discard old builds` property if this property is set in the last release of ATS for a pipeline but not in the current release. If this property is set in both releases, the script just updates the values according to the last release. Also, the script restores the `pipeline environment variables` values as per the last release of ATS. If any custom pipeline (created by the user) was present in the last release of ATS, the script restores that as well. It also restores the extra views created by NF users, for example, policy users, SCP users, and NRF users. Moreover, the script displays messages about the pending configuration that the user needs to perform manually. For example, a new pipeline or a new environment variable (for a pipeline) is introduced in the new release.

While deploying ATS without PV, Jenkins needs to be restarted for the restore process to complete. If the last release ATS contains the `Configuration_Type` parameter, the `Configuration_Type` script needs to be approved with the **In Process Script Approval** setting under **Manage ATS** in Jenkins for the restore process to complete.

2.7.2 Updating ats_install.sh

Currently, the `ats_install.sh` script copies the tests folder and Jenkins jobs folder into the ATS pod and then restarts the pod when deployed with PV.

How to Update ats_install.sh

Other NFs can also use the `ats_install.sh` scripts. However, additional post installation steps may have to be performed manually for a few NFs.

For the additional post installation commands, perform the following steps:

1. In the `ats_install.sh` script, there is a post install section between `#####POST_INSTALL_START#####` and `##### POST_INSTALL_END #####`.
 - a. Add the required post install commands.

Note

These commands are NF-specific.

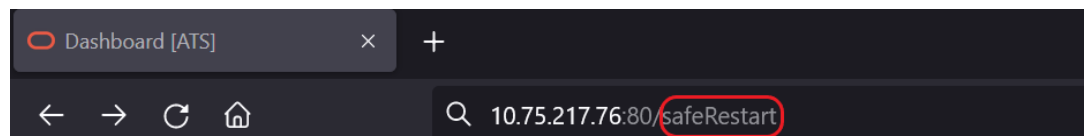
- b. Use the following commands:
 - `$namespace` for the namespace value
 - `$pod_name` for the pod name
 - `$ats_data_path` for the `ats_data` folder path (it has tests folder and Jenkins jobs folder provided as tar file in ATS package)
 - c. In the if-else block related to whether PV is enabled or not, add the following:
 - Add a command specific to `PVEnabled=true` in the `if` block.
 - Add a command specific to `PVEnabled=false` in the `else` block.
2. For additional inputs, enter the required code between `##### INPUT_START #####` and `##### INPUT_END #####`.

2.7.3 Restarting Jenkins without Restarting Pod

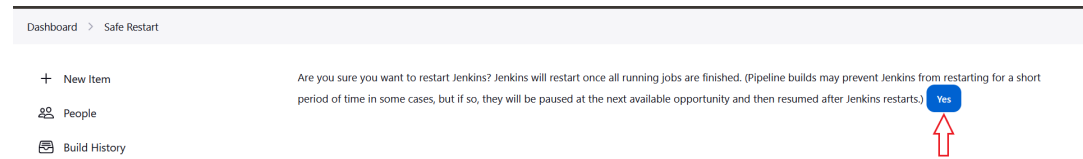
Perform the following procedure to restart Jenkins without restarting pods:

1. Log in as the Jenkins admin.
2. Go to the `<Jenkins_IP>:<port>/safeRestart`, for example, `10.87.73.32:32156/safeRestart`.

Figure 2-24 Safe Restart



3. Click **Yes**.

Figure 2-25 Restart Jenkins

2.7.4 Updating Stub Scripts

The following stubs can be updated:

- **stub_uninstall.sh**: This script requires the user's inputs and uninstalls all the stubs.
- **stub_install.sh**: This script requires the user's inputs and installs all the stubs.

Note

Currently, `stub_uninstall.sh` and `stub_install.sh` work.

Perform the following procedure to update the stub scripts for other NFs (NRF in this case):

1. Go to the stub folder.
2. From each script:
 - a. Remove the CNC Policy-specific stubs inputs (dns, amf, and ldap), and add the input code blocks for NRF-specific stubs.
 - b. For the stubs to uninstall, change the value of the `stubUninstallList` variable, and delete the variables for the CNC Policy-specific stubs below it.

Note

`stubUninstallList` contains the Helm release names of the common stubs that are deployed generally.

- c. Declare the variables for the NRF-specific stubs below the `stubUninstallList` line.
- d. Remove the Helm uninstallation commands of the policy-specific stubs, and add the Helm uninstallation commands of the NRF-specific stubs.
- e. For the stubs to install, change the value of the `stubInstallList` variable, and delete the variables for the CNC Policy-specific stubs below it.

Note

`stubInstallList` contains the Helm release names of the common stubs that are deployed generally.

- f. Declare the variables for the NRF-specific stubs below the `stubInstallList` line.
- g. Remove the Helm installation commands of the CNC Policy-specific stubs, and add the Helm installation commands of the NRF-specific stubs.

2.7.5 Running ATS and Stub Deployment Scripts

Perform the following procedure to run ATS and stub deployment scripts:

① Note

If you want to take a backup of the custom folders or Jenkins jobs and user's configuration or both, run the `ats_backup.sh` script.

1. Run the `ats_install.sh` script to install the new release ATS (values.yaml of the ATS Helm chart must be updated before this step).
2. Run the `ats_restore.sh` script to restore the new ATS pipeline and view configuration.

① Note

- You might perform the manual steps required for the restore script.
- You must copy all the necessary changes to the new release ATS from the last release ATS. To get the changes in the last release, you must refer to the custom folders in the last release ATS backup on the system with an existing backup using `ats_backup.sh` before this step.
- You can remove the last release ATS pod using the `ats_uninstall.sh` script while continuing to retain the last release PVC. You can use the last release PVC to port backward. Delete the last release PVC when you do not require the backward porting.

3. Run the `stub_install.sh` script to install all the new release stubs values.yaml of the stub Helm charts must be updated before this step.
4. Run the `stub_uninstall.sh` script to uninstall all the last release stubs.

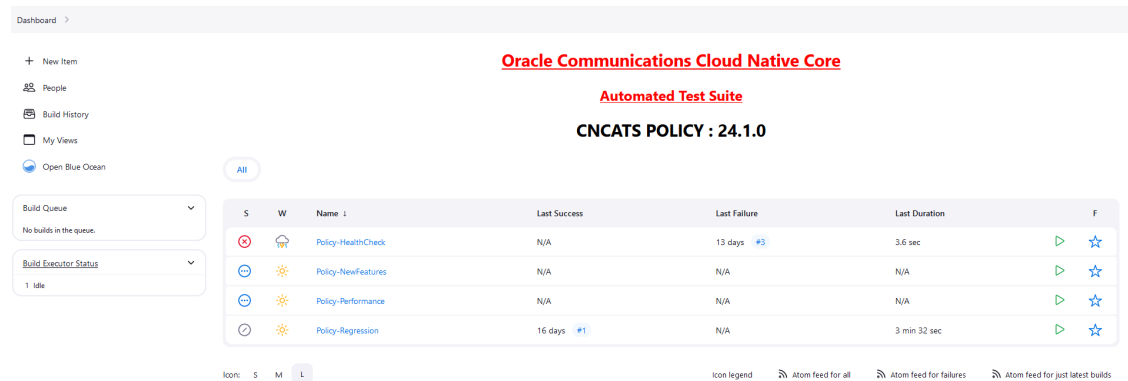
2.8 ATS System Name and Version Display on the ATS GUI

This feature displays the ATS system name and version on the ATS GUI.

You can log in to the ATS application using the login credentials to view the following:

- ATS system name: Abbreviated product name followed by NF name.
- ATS Version: Release version of ATS.

Figure 2-26 ATS System Name and Version



2.9 ATS Tagging Support

The ATS Tagging Support feature assists in running the feature files after filtering features and scenarios based on tags. Instead of manually navigating through several feature files, the user can save time by using this feature.

The GUI offers the following four options for selecting tag types:

- **Feature_Include_Tags:** The features that contain either of the tags available in the **Feature_Include_Tags** field are considered for tagging.
 - For example, "cne-common", "config-server". All the features that have either "cne-common" or "config-server" tags are taken into consideration.
- **Feature_Exclude_Tags:** The features that contain neither of the tags available in the **Feature_Exclude_Tags** field are considered for tagging.
 - For example, "cne-common", "config-server". All the features that have neither "cne-common" nor "config-server" as tags are taken into consideration.
- **Scenario_Include_Tags:** The scenarios that contain either of the tags available in the **Scenario_Include_Tags** field are considered.
 - For example, "sanity", "cleanup". The scenarios that have either "sanity" or "cleanup" tags are taken into consideration.
- **Scenario_Exclude_Tags:** The features that contain neither of the tags available in the **Scenario_Exclude_Tags** field are considered.
 - For example, "sanity", "cleanup". The scenarios that have neither "sanity" nor "cleanup" as tags are taken into consideration.

Filter with Tags

The procedure to filter feature files and scenarios based on tags are as follows:

1. On the NF home page, click any new feature or regression pipeline, where you want to use this feature.
2. In the left navigation pane, click **Build with Parameters**. The following image appears.

Figure 2-27 Filter with Tags

Dashboard > SCP-Regression >

Favorite
Open Blue Ocean
Rename
Test Results Analyzer
Pipeline Syntax

Build History trend
No builds
Atom feed for all Atom feed for failures

EXECUTION OPTIONS

TestSuite: Regression
Execute_Suite: R15
Configuration_Type: Product_Config
FilterWithTags: NO
Include_NewFeatures: NO
Fetch_Log_Upon_Failure: YES
Log_Type: AppLog
Log_Level: DEFAULT
HTTPX_LOG_LEVEL: Disabled

FEATURES AND TESTCASES

Features: All
TestCases: All

3. Select **Yes** from the **FilterWithTags** drop-down menu. The result shows four input fields. The default value of **FilterWithTags** field is "No".
4. The input fields serve as a search or filter, displaying all tags that match the prefix entered. You can select one or multiple tags.

Figure 2-28 Tags Matching with Entered Prefix

Dashboard > Policy-Regression >

Favorite
Open Blue Ocean
Stages
Rename
Test Results Analyzer
Pipeline Syntax

Build History trend
Filter...
Feb 20, 2024, 7:02 AM
Atom feed for all Atom feed for failures

EXECUTION OPTIONS

TestSuite: Regression
SUT: PCF
Configuration_Type: Product_Config
FilterWithTags: Yes
Feature_Include_Tags: c
Feature_Exclude_Tags:
Scenario_Include_Tags:
Scenario_Exclude_Tags:

☒ cmservice
☐ config-service
☒ cne-common
☐ config-server

Submit

5. Select the required tags from the different tags list and click **Submit**.

The specified feature-level tags are used to filter out features that contain any one of the include tags and none of the exclude tags. Here, any or both the fields may be left empty. All features are automatically taken into consideration when both fields are empty.

The scenario level tags are used to filter out the scenarios from the features filtered above. Only scenarios with any of the include tags and none of the exclude tags are considered. Any or both fields can be empty. When both fields are empty, all the scenarios from the above filtered feature files are considered.

Note

- If you select the Select_Option as 'All', all the displayed features and scenarios will run.
- If you select the Select_Option as 'Single/MultipleFeatures', it enables you to select some features, and only those features and respective scenarios are going to run.

2.9.1 Combination of Tags and their Results

The combination of tags and expected results are as follows.

Table 2-5 Result of Filtered Tags

Feature_Include	Feature_Exclude	Scenario_Include	Scenario_Exclude	Results
-	-	-	-	All the features and scenarios are taken into consideration.
"abc","def"	-	-	-	Features with either "abc" or "def" tags and all scenarios from the filtered features are taken into consideration.
-	"abc","def"	-	-	All the features with neither "abc" nor "def" tags and all scenarios from the filtered features are taken into consideration.
-	-	"sanity","cne"	-	Scenarios with either "sanity" or "cne" tags and features having these scenarios are taken into consideration.
-	-	-	"sanity","cne"	Scenarios with neither "sanity" nor "cne" tags and features having these filtered scenarios are taken into consideration.
"abc","def"	"ghi"	-	-	Features with either "abc" or "def" tags but without the "ghi" tag and all scenarios from filtered features are taken into consideration.

Table 2-5 (Cont.) Result of Filtered Tags

Feature_Include	Feature_Exclude	Scenario_Include	Scenario_Exclude	Results
"abc","def"	-	"sanity","cne"	-	Scenarios only with either "sanity" or "cne" tags and only features that contain these scenarios and have either "abc" or "def" as feature tags are taken into consideration.
"abc","def"	-	-	"sanity","cne"	Scenarios with neither "sanity" nor "cne" tags and only features that contain the filtered scenarios and have either "abc" or "def" feature tags are taken into consideration.
-	"ghi"	"sanity","cne"	-	Features without the "ghi" tag and scenarios with either "sanity" or "cne" tags from the filtered features are taken into consideration.
-	"ghi"	-	"sanity","cne"	Features without the "ghi" tag and scenarios without the "sanity" and "cne" tags from filtered features are taken into consideration.
-	-	"sanity","cne"	"cleanup"	Scenarios with either the "sanity" or "cne" tags and without the "cleanup" tag and features with filtered scenarios are taken into consideration.
"abc","def"	"ghi"	"sanity","cne"	-	Scenarios with either the "sanity" or "cne" tags and features that have these scenarios and have either the "abc" or "def" tags but not the "ghi" tag are taken into consideration.
"abc","def"	-	"sanity","cne"	"cleanup"	Scenarios with either the "sanity" or "cne" tags and without the "cleanup" tag, and features having the filtered scenarios and having the feature tags either "abc" or "def" are taken into consideration.

Table 2-5 (Cont.) Result of Filtered Tags

Feature_Include	Feature_Exclude	Scenario_Include	Scenario_Exclude	Results
"abc","def"	"ghi"	-	"cleanup"	Scenarios without the tag "cleanup", and features with filtered scenarios and having either "abc" or "def" as feature tags but not the "ghi" tag are taken into consideration.
-	"ghi"	"sanity","cne"	"cleanup"	Scenarios with either the "sanity" or "cne" tags and without the "cleanup" tag, and features with filtered scenarios and not the tag "ghi," are taken into consideration.
"abc","def"	"ghi"	"sanity","cne"	"cleanup"	Scenarios with either "sanity" or "cne" tags and without the "cleanup" tag, and features with filtered scenarios and feature tags either "abc" or "def" but without the tag "ghi" are taken into consideration.

 Note

- The tags mentioned in the table are just examples; they may or may not be actually used.
- The **Replay** option in the Jenkins GUI is not supported for tag-related test case execution. Always trigger builds related to tagging from the **Build with Parameter** step, and do not replay any previous builds.
-

2.10 Custom Folder Implementation

The Custom Folder Implementation feature allows the user to update, add, or delete test cases without affecting the original product test cases in the new features, regression, and performance folders. The implemented custom folders are cust_newfeatures, cust_regression, and cust_performance. The custom folders contain the newly created, customised test cases.

Initially, the product test case folders and custom test case folders will have the same set of test cases. The user can perform customization in the custom test case folders, and ATS always runs the test cases from the custom test case folders. If the option "Configuration_Type" is present on the GUI, the user needs to set its value to "Custom_Config" to populate test cases from the custom test case folders.

Figure 2-29 Custom Config Folder

Automated Test Suite - POLICY

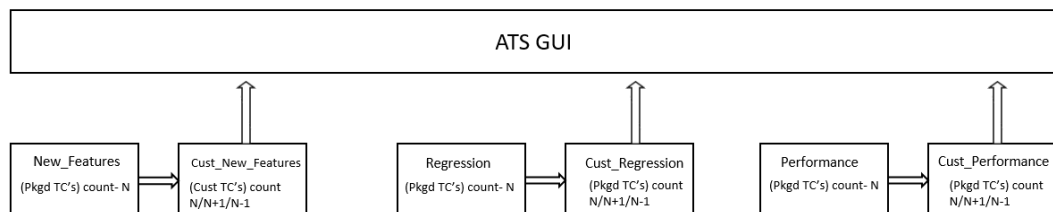
EXECUTION OPTIONS

TestSuite Regression	SUT PCF	Configuration_Type Product_Config Product_Config Custom_Config	FilterWithTags No
Include_NewFeatures NO	Fetch_Log_Upon_Failure YES		

Summary of Custom Folder Implementation

- Separate folders such as cust_newfeatures, cust_regression, and cust_performance are created to hold the custom cases.
- The prepackaged test cases are available in the newfeature and regression Folder.
- The user copies the required test cases to the cust_newfeatures and cust_regression folders, respectively.
- Jenkins always points to the cust_newfeatures and cust_regression folders to populate them in the menu.
If someone initially launches ATS, they will not see any test cases in the menu if the cust folders are not populated. To avoid this, it is recommended to prepopulate both the folders, cust and original, and ask the user to modify only the cust folder if needed.

Figure 2-30 Summary of Custom Folder Implementation



2.11 Individual Stage Group Selection

The Individual Stage Group Selection feature allows you to select and execute a single or multiple stages or groups by selecting a check box for the corresponding stage or group.

Follow the steps to select an individual stage or group:

1. Click NF-Regression or NF-NewFeatures, and then click **Build with Parameters**.
2. On the FEATURE and TESTCASES section, click **Select** from the **Features** drop-down menu.
3. Select the corresponding check box to select any number of stages or groups you want to run from the list available for execution.

Figure 2-31 Stages or Groups Selection

Dashboard > Policy-NewFeatures >

Favorite
Open Blue Ocean
Rename
Test Results Analyzer
Pipeline Syntax

Build History trend
No builds
Atom feed for all Atom feed for failures

EXECUTION OPTIONS

TestSuite: NewFeatures SUT: PCF Configuration_Type: Product_Config FilterWithTags: No

Include_Regression: NO Fetch_Log_Upon_Failure: YES

FEATURES AND TESTCASES

Features: Select
Stage: stage1
Group: group1
Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_verify
internal_SM_Multi_ServerHeader_with_Retry_using_SCP
internal_Binding_Header_AMPolicy_Svname_nf_instance
internal_SM_Rt_AAR-L_without_MCD

Stage: stage2
Group: group3
Pending_Transaction_N7_Fast_Lock_Bulwark

Stage: stage3
Group: group3
internal_max_n_session

TestCases: All
Total Test Cases are : 14

Build Cancel

4. Scroll down to click **Build**.

2.12 Lightweight Performance

The Lightweight Performance feature allows you to run performance test cases. In ATS, a new pipeline known as "<NF>-Performance", where NF stands for Network Function, is introduced, for example, Policy-Performance.

Figure 2-32 Sample Screen: Home Page

Dashboard >

+ New Item
People
Build History
My Views
Open Blue Ocean

Build Queue
No builds in the queue.

Build Executor Status
1 title

Oracle Communications Cloud Native Core
Automated Test Suite
CNCATS POLICY : 24.1.0

S	W	Name	Last Success	Last Failure	Last Duration	F
⊙	☀	Policy-HealthCheck	N/A	13 days #3	3.6 sec	▶ ☆
⊙	☀	Policy-NewFeatures	N/A	N/A	N/A	▶ ☆
⊙	☀	Policy-Performance	N/A	N/A	N/A	▶ ☆
⊙	☀	Policy-Regression	16 days #1	N/A	3 min 32 sec	▶ ☆

Icon: S M L
Icon legend Atom feed for all Atom feed for failures Atom feed for just latest builds

The <NF>-Performance pipeline verifies from 500 to 1k TPS (Transactions per Second) of traffic using the http-go tool, a tool used to run the traffic on the backend. It also helps to monitor the CPU and memory of microservices while running lightweight traffic.

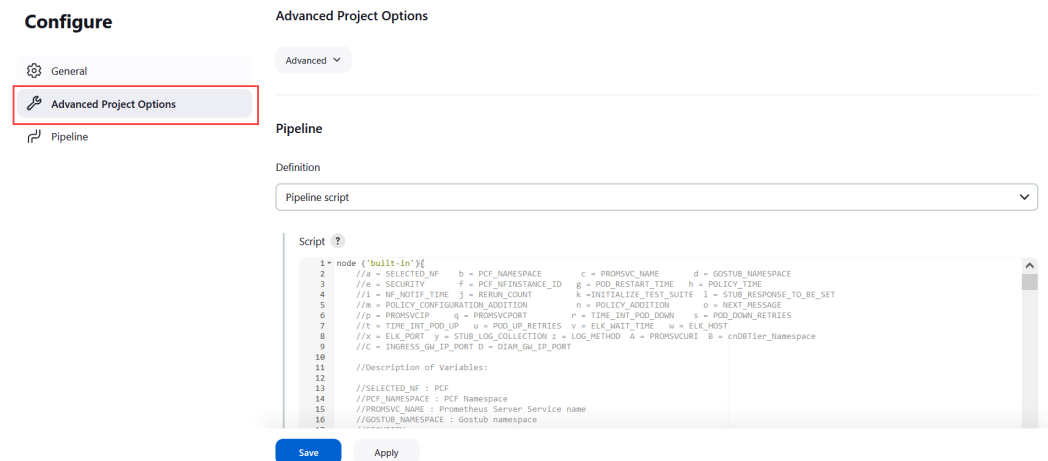
The duration of the traffic run can be configured on the pipeline.

2.12.1 Configure <NF>-Performance Pipeline

Perform the following to configure the performance pipeline:

1. On the NF home page, click **<NF>-Performance** pipeline, and then click **Configure**. The **General** tab appears. The user must wait for the page to load completely.
2. Click the **Advanced Project Options** tab. Scroll down to reach the **Pipeline** configuration section.

Figure 2-33 Advanced Project Options



3. Update the configurations as per your NF requirements and click **Save**. The **Pipeline <NF>-Performance** page appears.
4. Click **Build Now**. This triggers lightweight traffic for the respective network function. For more information, see [Running SLF-Performance Pipeline: Model 2: Single Site Deployment](#).

2.13 Managing Final Summary Report, Build Color, and Application Log

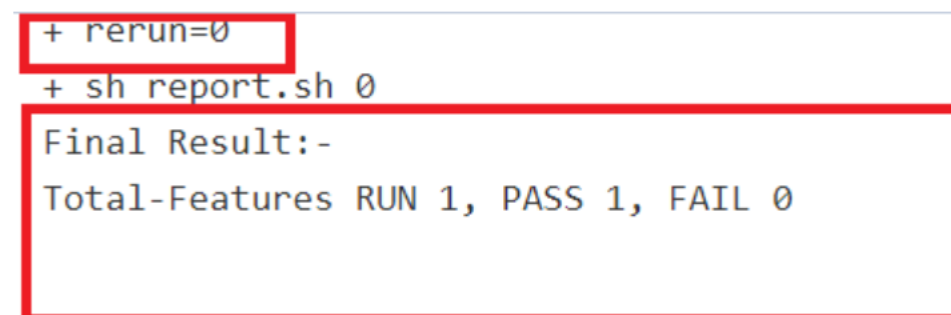
This feature displays an overall execution summary, such as the total run count, pass count, and fail count.

Supports Implementation of Total-Features

ATS supports implementation of **Total-Features** in the final summary report. Based on the rerun value set, the **Final Result** section in the final summary report displays the Total-Features output.

- If rerun is set to 0, the test result report shows the following result:

Figure 2-34 Total-Features = 1, and Rerun = 0



- If rerun is set to non-zero, the test result report shows the following result:

Figure 2-35 Total-Features = 1, and Rerun = 2

```
+ rerun=2
+ sh report.sh 2
Initial Run :-
Features RUN 1, PASS 0, FAIL 1

1st Rerun:-
Features RUN 1, PASS 0, FAIL 1

2nd Rerun:-
Features RUN 1, PASS 0, FAIL 1

Final Result:-
Total-Features RUN 1, PASS 0, FAIL 1
```

Changes After Parallel Test Execution Framework Feature Integration

After incorporating the Parallel Test Execution feature, the following results were obtained:

Final Summary Report Implementations

Figure 2-36 Group Wise Results

```
*****Stage: stage1 Group: group3 Result*****
Initial Run :-
Features RUN 1, PASS 0, FAIL 1

1st Rerun:-
Features RUN 1, PASS 0, FAIL 1

2nd Rerun:-
Features RUN 1, PASS 0, FAIL 1

stage1 group3
Final Result:-
Total-Features RUN 1, PASS 0, FAIL 1
```

Figure 2-37 Overall Result When Selected Feature Tests Pass

```
*****Overall Result*****
Took 0h0m50s

Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0
```

Figure 2-38 Overall Result When Any of the Selected Feature Tests Fail

```

*****Overall Result*****
Failing scenarios:
Stage:stage1 Group:group3
  SCP_Registration_With_PLMNList.feature:45  Static configuration to NRF for all nfType

Took 0h1m59s

Initial Run:-
Features RUN 1, PASS 0, FAIL 1

1st Rerun:-
Features RUN 1, PASS 0, FAIL 1

2nd Rerun:-
Features RUN 1, PASS 0, FAIL 1

Final Result:-
Total-Features RUN 1, PASS 0, FAIL 1

```

Implementing Build Colors

ATS supports implementation of build color. The details are as follows:

Table 2-6 Build Color Details

Rerun Values	Rerun set to zero		Rerun set to non-zero		
Status of Run	All Passed in Initial Run	Some Failed in Initial Run	All Passed in Initial Run	Some Passed in Initial Run, Rest Passed in Rerun	Some Passed in Initial Run, Some Failed Even After Rerun
Build Status	SUCCESS	FAILURE	SUCCESS	SUCCESS	FAILURE
Pipeline Color	GREEN	Execution Stage where test cases failed shows YELLOW color, rest of the successful stages are GREEN.	GREEN	GREEN	Execution Stage where test cases failed shows YELLOW color, rest of the successful stages are GREEN
Status Color	BLUE	RED	BLUE	BLUE	RED

Changes After Integrating Parallel Test Execution Framework Feature

In sequential execution, the build color or overall pipeline status of any run was mainly dependent on the following parameters:

- the rerun count and the pass or fail status of test cases in the initial run
- the rerun count and the pass or fail status of test cases in the final run

For the parallel test case execution, the pipeline status also depends on another parameter, "Fetch_Log_Upon_Failure," which is given in the **build with parameters** page. If the parameter Fetch_Log_Upon_Failure is not there, its default value is considered "NO".

Table 2-7 Pipeline Status When Fetch_Log_Upon_Failure = NO

Rerun Values	Rerun set to zero		Rerun set to non-zero		
Passed/Failed	All Passed in Initial Run	Some Failed in Initial Run	All Passed in Initial Run	Some Passed in Initial Run, Rest Passed in Rerun	Some Passed in Initial Run, Some Failed Even After Rerun
Status	SUCCESS	FAILURE	SUCCESS	SUCCESS	FAILURE

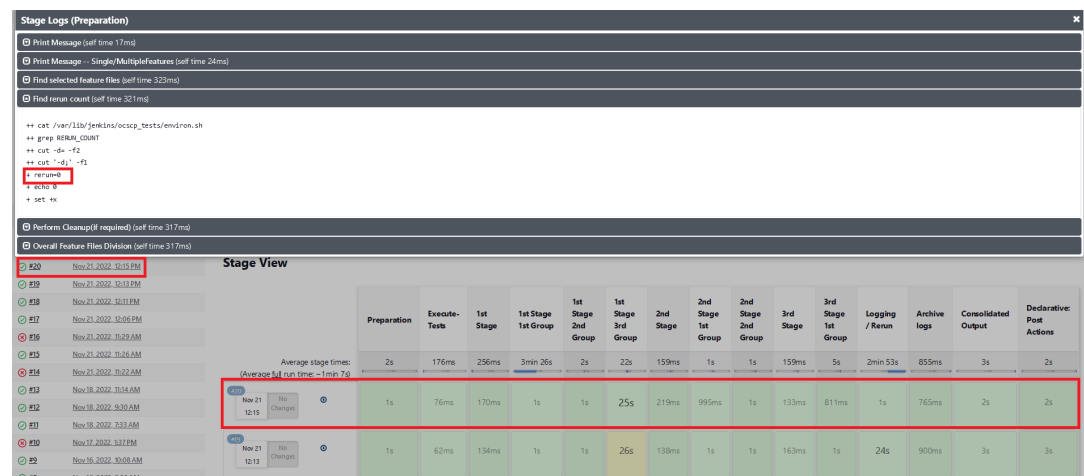
Table 2-8 Pipeline Status When Fetch_Log_Upon_Failure = YES

Rerun Values	Rerun set to zero			Rerun set to non-zero		
Passed/Failed	All Passed in Initial Run	Some Failed in Initial Run and Failed in Rerun	Some Failed in Initial Run and Passed in Rerun	All Passed in Initial Run	Some Passed in Initial Run, Rest Passed in Rerun	Some Passed in Initial Run, Some Failed Even After Rerun
Status	SUCCESS	FAILURE	SUCCESS	SUCCESS	SUCCESS	FAILURE

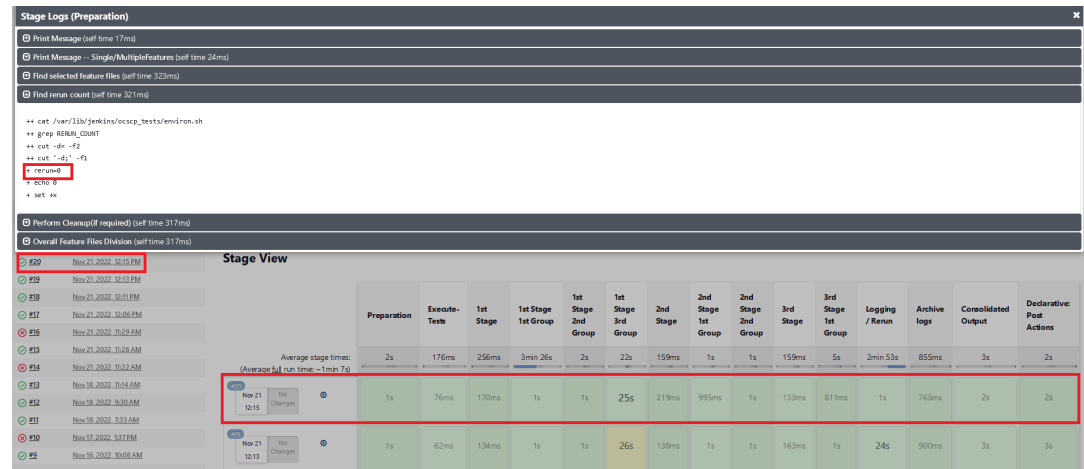
Some common combinations of these parameters, such as rerun_count, Fetch_Log_Upon_Failure, and pass/fail status of test cases in initial and final run and the corresponding build colors are as follows:

- When Fetch_Log_Upon_Failure is set to YES and rerun_count is set to 0, test cases pass in the initial run. The pipeline will be green, and its status will show as blue.

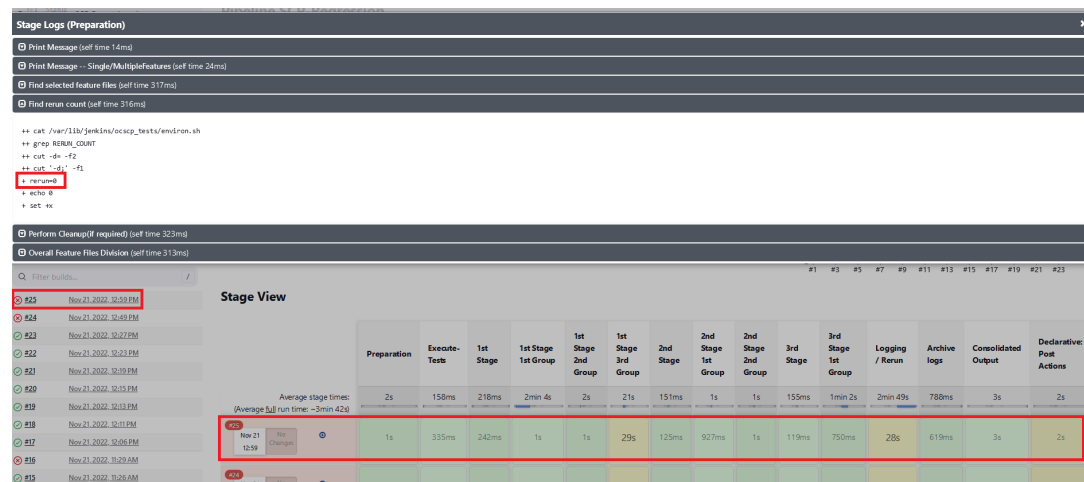
Figure 2-39 Fetch_Log_Upon_Failure is set to YES and rerun_count is set to 0, test cases pass



- When Fetch_Log_Upon_Failure is set to YES and rerun_count is set to 0, test cases fail on the initial run but pass during the rerun. The initial execution stage is yellow and all subsequent successful stages will be green, and the status will be blue.

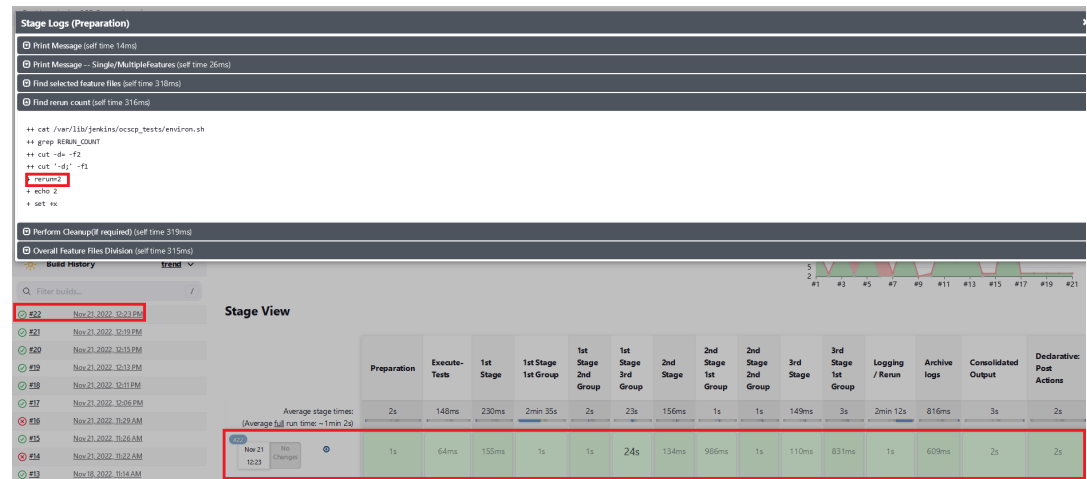
Figure 2-40 Test Cases Fail on the Initial Run but Pass in the Rerun

- When `Fetch_Log_Upon_Failure` is set to YES and `rerun_count` is set to 0, test cases fail in both the initial and the rerun. Execution stages will show as yellow, all other successful stages will be shown as green, and the overall pipeline status will be red.

Figure 2-41 Test Cases Fail in Both the initial and the Rerun

- When `Fetch_Log_Upon_Failure` is set to YES and the `rerun_count` is set to non-zero. If all of the test cases pass in the first run, no rerun will be initiated because the cases have already been passed. The pipeline will be green, and the status will be indicated in blue.

Figure 2-42 All of the Test cases Pass in the Initial Run



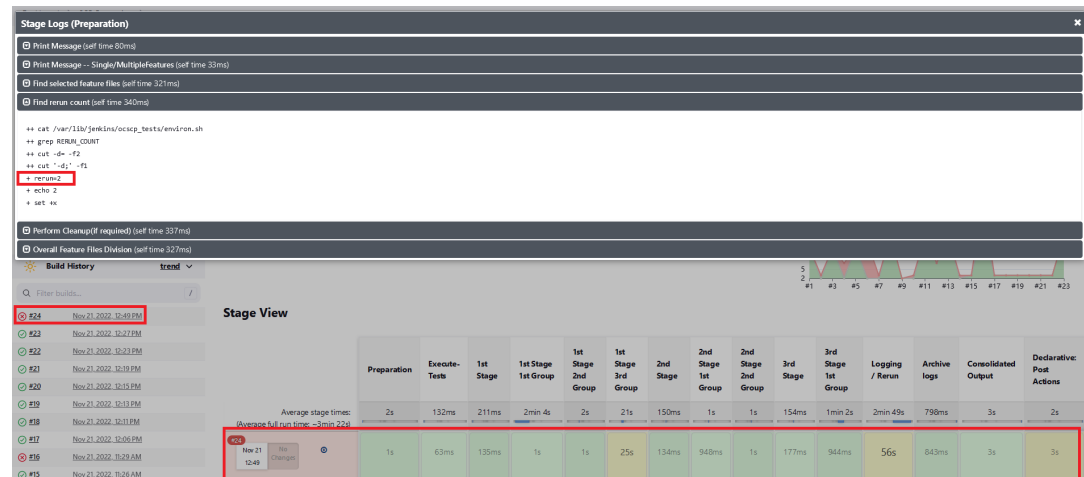
- When Fetch_Log_Upon_Failure is set to YES and the rerun count is set to non-zero. If some of the test cases fail in the initial run and the remaining ones pass in one of the remaining reruns, then the initial test case execution stages will show as yellow, the remaining stages as green, and the overall pipeline status as blue.

Figure 2-43 Test Cases Fail in the Initial Run and the Remaining Ones Pass



- When Fetch_Log_Upon_Failure is set to YES and the rerun count is set to non-zero. If some of the test cases fail in the initial run and the remaining ones fail in all the remaining reruns, the stages of test case execution will be shown in yellow, the remaining stages in green, and the overall pipeline status in red.

Figure 2-44 Test Cases Fail in the Initial and Remaining Reruns



- Whenever any of the multiple Behave processes that are running in the ATS are exited without completion, the stage in which the process exited and the consolidated output stage are shown as yellow, and the overall pipeline status will be yellow. Also in the consolidated output stage, near the respective stage result, the exact run in which the Behave processes exited without completion will be printed.

Figure 2-45 Stage View When Behave Process is Incomplete

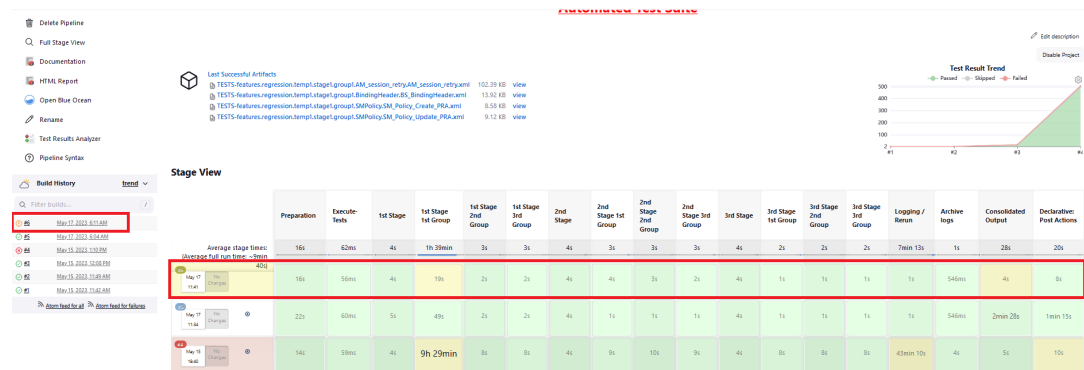
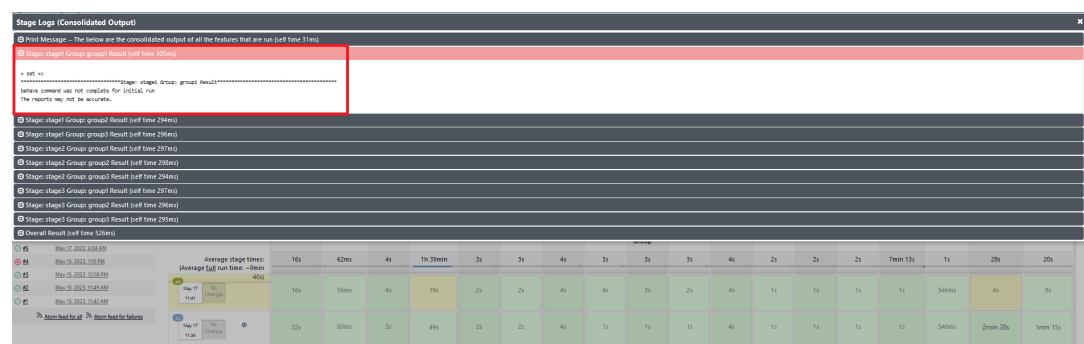


Figure 2-46 Consolidated Report for a Group When a Behave Process was Incomplete



Implementing Application Log

ATS automatically fetches the SUT Debug logs during the rerun cycle if it encounters any failures and saves them in the same location as the build console logs. The logs are fetched for the rerun time duration only using the timestamps. If, for some microservices, there are no log entries in that time duration, it does not capture them. Therefore, the logs are fetched only for the microservices that have an impact or are associated with the failed test cases.

Location of SUT Logs: `/var/lib/jenkins/.jenkins/jobs/PARTICULAR-JOB-NAME/builds/BUILD-NUMBER/date-timestamp-BUILD-N.txt`

① Note

The file name of the SUT log is added as a suffix with the date, timestamp, and build number (for which the logs are fetched). These logs share the same retention period as build console logs, set in the ATS configuration. It is recommended to set the retention period to optimal owing to the Persistent Volume Claim (PVC) storage space availability.

2.14 Modifying Login Password

You can log in to the ATS application using the default login credentials. The default login credentials are shared for each NF in the respective chapter of this guide.

Perform the following procedure to modify the default password:

1. Log in to the ATS application using the default login credentials. The home page of the respective NF appears.
2. Click the down arrow next to the user name.
3. Click **Configure**.
4. In the Password section, enter the new password in the **Password** and **Confirm Password** fields..

Figure 2-47 Logged-in User Details

The screenshot displays the 'Logged-in User Details' page in the Oracle Cloud Native Core Automated Testing Suite (ATS). The page is organized into several sections:

- Full Name:** A text field containing the name 'infuser'.
- Description:** A large text area for user description.
- API Token:** A section for managing API tokens, including a 'Current token' and an 'Add new token' button.
- Credentials:** A section for user credentials, with a note that they are only available to the user they belong to.
- E-mail:** A section for email address, showing 'infuser@oracle.com'.
- My Views:** A section for default view settings.
- Notification URL:** A section for notification URL settings.
- Password:** A section for password management, highlighted with a red box. It includes fields for 'Password', 'Confirm Password', and 'New Password'.
- SSH Public Keys:** A section for SSH public keys.
- Session Termination:** A section for session termination settings.
- Settings for search:** A section for search settings.

5. Click **Save**.
A new password is set for you.

2.15 Multiselection Capability for Features and Scenarios

ATS allows you to select and run single or multiple features and scenarios by selecting a check box for the corresponding features or scenarios.

2.15.1 Feature Level Selection

Perform the following procedure to run the single or multiple features:

1. Log in to ATS using the respective <NF> login credentials.
2. On the NF home page, click any new feature or regression pipeline from where you want to run the feature.
3. In the left navigation pane, click **Build with Parameters**.
4. Scroll down to the FEATURES AND TEST CASES section.
5. Click **Select** from the **Features** drop-down.

Figure 2-48 Feature Selection

Dashboard > Policy-NewFeatures >

Oracle Communications Cloud Native Core
Automated Test Suite - POLICY

EXECUTION OPTIONS

TestSuite NewFeatures	SUT PCF	Configuration_Type Product_Config	FilterWithTags No
Include_Regression NO	Fetch_Log_Upon_Failure YES		

FEATURES AND TESTCASES

Features: Select

Stage: stage1 ☐

Group: group1 ☐

- ☒ Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_verify
- ☒ Internal_SM_Multi_ServerHeader_with_Retry_using_SCP

Stage: stage2 ☐

Group: group3 ☐

- ☐ Internal_Binding_Header_AMPolicy_Svname_nf_instance
- ☐ Internal_SM_Rx_AAR-I_without_MCD

Stage: stage3 ☐

Group: group3 ☐

☐ Pending_Transaction_N7_Fast_Lock_Bulwark

6. Select any number of features by selecting the check box for the corresponding feature you want to run from the list available for execution.
7. Click **Build**.

2.15.2 Scenario Selection

Perform the following procedure to run the single or multiple test cases or scenarios related to features:

1. Log in to ATS using the respective <NF> login credentials.
2. On the NF home page, click any new feature or regression pipeline from where you want to run the feature.
3. In the left navigation pane, click **Build with Parameters**.
4. Scroll down to the FEATURES AND TEST CASES section.
5. Click **Select** from the **Features** drop-down.
6. Select any number of features by selecting the check box for the corresponding feature to view the TestCases details mapped to each feature.
7. Click **Select** from the **TestCases** drop-down.

Figure 2-49 Scenario or Testcase Selection

TestCases Select ▼

Total Test Cases are : 2

Stage: stage1 Group: group1

Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_verify

☒ Bulwark_CHF_UDR_notification_RedCap_verify

▶ Build Cancel

8. Select the check box for the corresponding test case or scenario to run the test cases mapped to the feature.
9. Click **Build**.

2.16 Parallel Test Execution

Parallel test execution allows you to perform multiple logically grouped tests simultaneously on the same System Under Test (SUT) to reduce the overall execution time of ATS.

ATS currently runs all its tests in a sequential manner, which is time-consuming. With parallel test execution, tests can be run concurrently rather than sequentially or one at a time. Test cases or feature files are now separated into different folders, such as stages and groups, for concurrent test execution. Different stages, such as stage 1, stage 2, and stage 3, run the test cases in a sequential order, and each stage has its own set of groups. Test cases or feature files available in different groups operate in parallel. When all the groups within one stage have completed their execution, only then the next stage will start the execution.

Pipeline Stage View

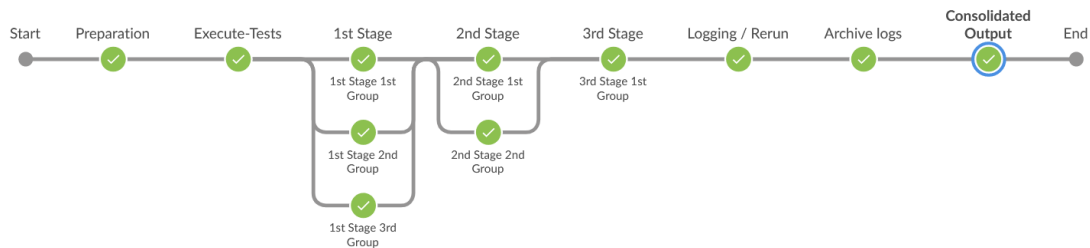
The pipeline stage view appears as follows:

Figure 2-50 Pipeline Stage View**Stage View**

	Preparation	Execute-Tests	1st Stage	1st Stage 1st Group	1st Stage 2nd Group	1st Stage 3rd Group	2nd Stage	2nd Stage 1st Group	2nd Stage 2nd Group	3rd Stage	3rd Stage 1st Group	Logging / Rerun	Archive logs	Consolidated Output	Declarative: Post Actions
Average stage times: (Average full run time: ~54s)	5s	59ms	1s	31s	2s	26s	1s	999ms	1s	1s	997ms	33s	487ms	2s	6s
#92 Feb 07 14:10 No Changes	5s	46ms	1s	1s	1s	27s	1s	1s	1s	1s	887ms	2s	580ms	2s	5s

Pipeline Blue Ocean View

Blue Ocean is a Jenkins plugin that provides a better representation of concurrent execution with stages and groups. The pipeline blue ocean view appears as follows:

Figure 2-51 Pipeline Blue Ocean View**Impact on Other Framework Features**

The integration of the parallel test framework feature has an impact on the following framework features. See the following sections for more details:

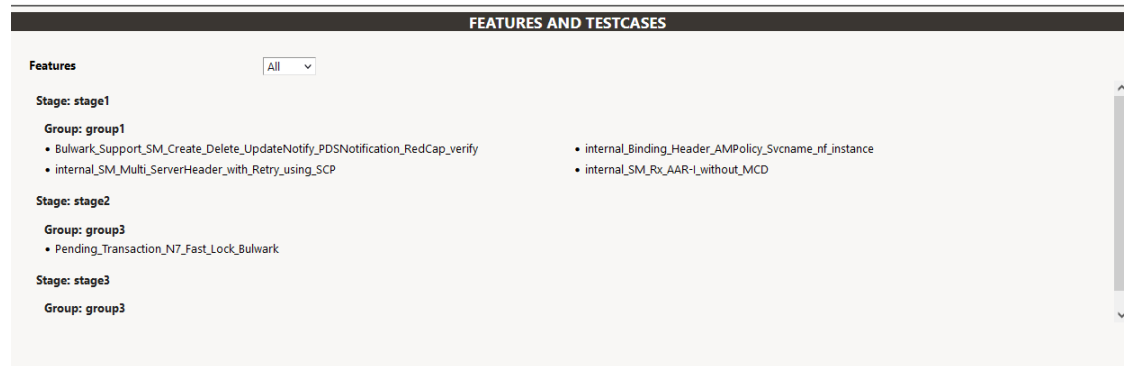
- [Application Log Collection](#)
- [ATS API](#)
- [Managing Final Summary Report, Build Color, and Application Log](#)
- [PCAP Log Collection](#)

2.16.1 ATS GUI Page Changes

This section describes the changes to the ATS GUI page to trigger a build.

Changes in ATS GUI Page to Trigger a Build

The feature name, file name, and test case name are displayed under their stage and group names.

Figure 2-52 Displays Stages and Groups

2.16.2 ATS Console Log Changes

This section describes the changes to the ATS console log. The ATS console log contains logs of all the stages and groups. For more details, see [Downloading or Viewing Individual Group Logs](#).

- A test case's stage and group names are listed in the logger statements for that test case.

Figure 2-53 Logger Statement

```
2022-12-06 09:34:53,762[32m INFO LOG: stage2.group1 STEP:5453 [0m] [32m10.233.109.13[0m
2022-12-06 09:34:53,762[32m INFO LOG: stage2.group1 STEP:5454 [0m] [32mudm1svc.scpsvc.svc.galaxy[0m
2022-12-06 09:34:53,893[32m INFO LOG: stage2.group1 STEP:5459 [0m] [32m{"url":"/USEast/nudm-uecm/v1/imsi
```

- When a test case fails, a list of test cases running in parallel gets printed to make the debugging easier. The name of the test case and the absolute path to the feature file it belongs to are listed in this list.

Figure 2-54 Absolute Path of Feature File

```
2022-12-06 11:27:19,253[32m INFO LOG: stage1.group3.CleanUP:2317 [0m] [32mBelow feature(s)/scenario(s) were found
to be running in parallel at the time of failure:
{
  "feature_path": "/var/lib/jenkins/ocscp_tests/features/regression/release16features/allfeatures/stage1/group3
/SCP_Registration_With_PLMNList.feature",
  "scenario_name": "Static configuration to NRF for all nfType"
}
{
  "feature_path": "/var/lib/jenkins/ocscp_tests/features/regression/release16features/allfeatures/stage1/group1
/Version_support.feature",
  "scenario_name": "Scenario-1_To test Forward route for notification request UECM AMF Registration messages with
3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding header bl=nfset, when the apiVersionInUri is set to v2 and the
message is also sent to v2"
}
```

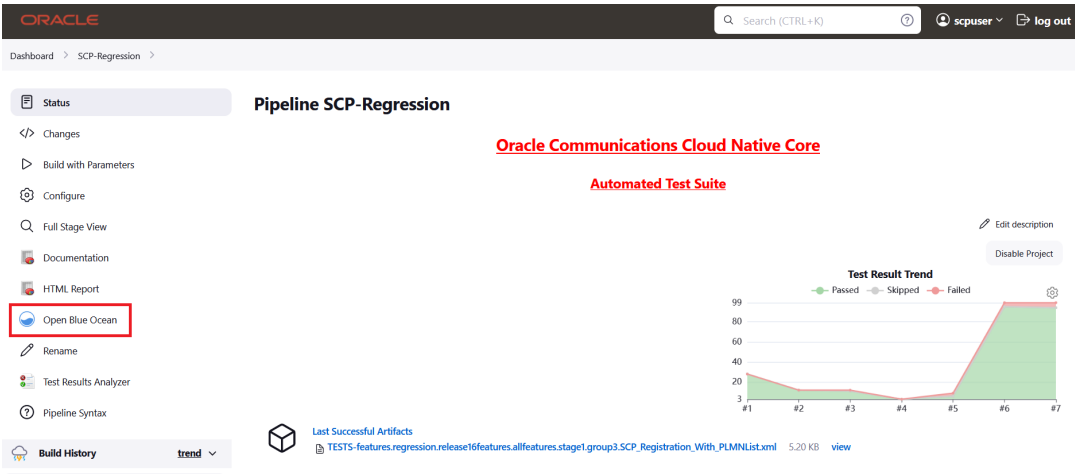
- The test result summary contains a summary for each group and an overall summary, along with the details of failing scenarios (stage-groupwise) and the total time taken by any pipeline execution. For further information, see the [Final Summary Report, Build Colour Changes](#).

2.16.3 Downloading or Viewing Individual Group Logs

To download individual group logs:

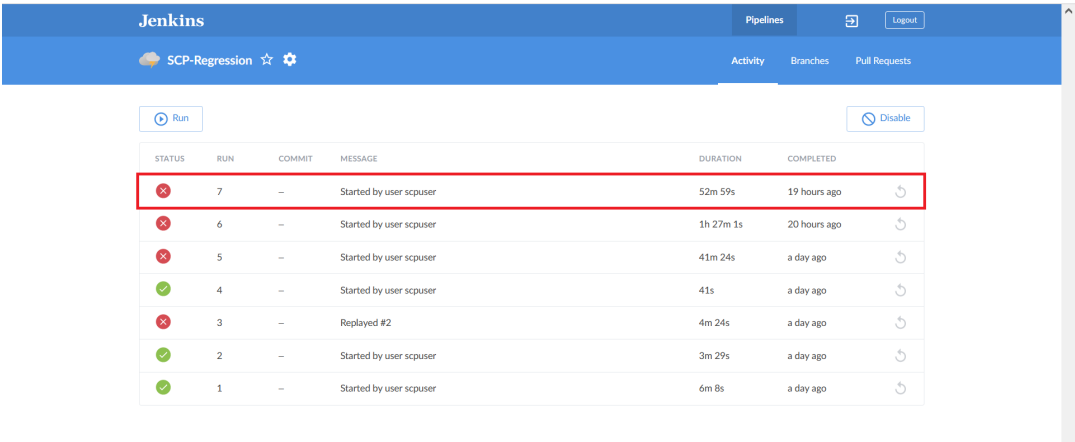
1. On the Jenkins pipeline page, click **Open Blue Ocean** in the left navigation pane.

Figure 2-55 Jenkins Pipeline Page



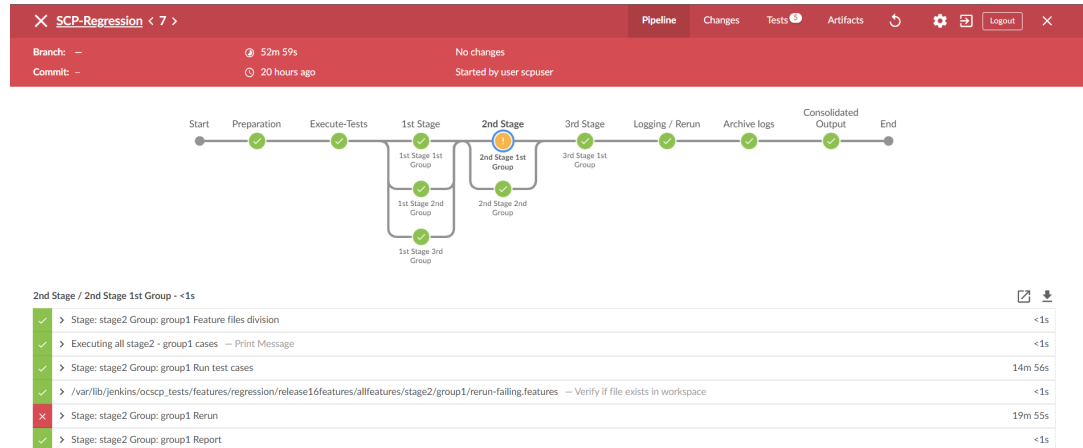
2. Click the desired build row on the Blue Ocean page.

Figure 2-56 Run the Build



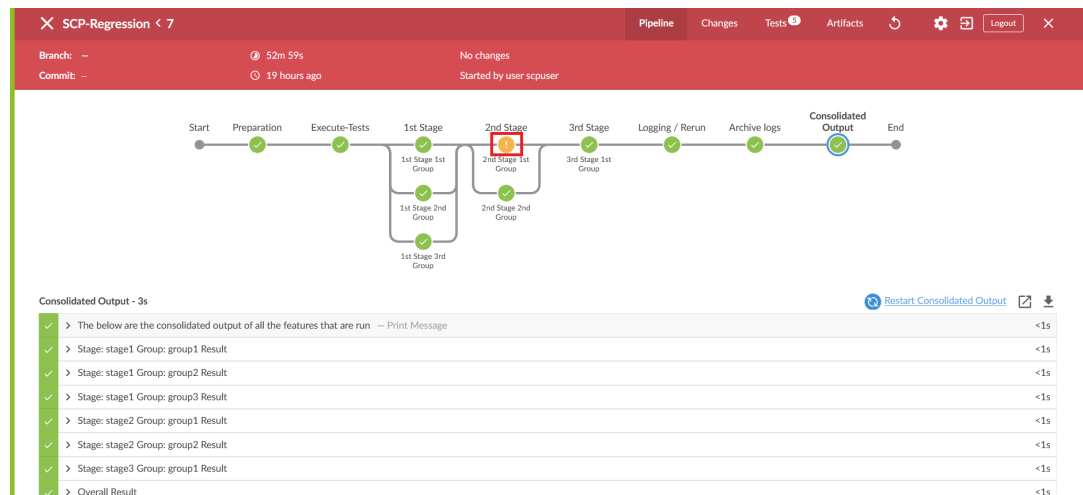
3. The selected build appears. The diagram displays the order in which the different stages, or groups, are executed.

Figure 2-57 Stage Execution



- Click the desired group to download the logs.

Figure 2-58 Executed Groups



- Click the **Download** icon on the bottom right of the pipeline. The log for the selected group is downloaded to the local system.

Figure 2-59 Download Logs

SCP-Regression < 7

Branch: — 52m 59s No changes
Commit: — 19 hours ago Started by user scpuser

Start Preparation Execute-Tests 1st Stage 2nd Stage 3rd Stage Logging / Rerun Archive logs Consolidated Output End

2nd Stage / 2nd Stage 1st Group - <1s

✓	> Stage: stage2 Group: group1 Feature files division	<1s	Download the log
✓	> Executing all stage2 - group1 cases — Print Message	<1s	
✓	> Stage: stage2 Group: group1 Run test cases	14m 56s	
✓	> /var/lib/jenkins/ocscp_tests/features/regression/release16/features/stage2/group1/rerun-failing.features — Verify if file exists in workspace	<1s	
✗	> Stage: stage2 Group: group1 Rerun	19m 55s	

Show complete log

- To view the log, click the **Display Log** icon. The logs are displayed in a new window.

Figure 2-60 Display Logs

SCP-Regression < 7 >

Branch: — 52m 59s No changes
Commit: — 20 hours ago Started by user scpuser

Start Preparation Execute-Tests 1st Stage 2nd Stage 3rd Stage Logging / Rerun Archive logs Consolidated Output End

2nd Stage / 2nd Stage 1st Group - <1s

✓	> Stage: stage2 Group: group1 Feature files division	<1s	Display Log
✓	> Executing all stage2 - group1 cases — Print Message	<1s	
✓	> Stage: stage2 Group: group1 Run test cases	14m 56s	
✓	> /var/lib/jenkins/ocscp_tests/features/regression/release16/features/stage2/group1/rerun-failing.features — Verify if file exists in workspace	<1s	
✗	> Stage: stage2 Group: group1 Rerun	19m 55s	
✓	> Stage: stage2 Group: group1 Report	<1s	

Viewing Individual Group Logs without using Blue Ocean

There are two alternate ways to view individual group logs:

- Using Stage View
 - On the Jenkins pipeline page, hover the cursor over the group in stage view to view the logs.
 - Click the label **Logs** which appears in a pop-up. There will be a new pop-up window. It contains many rows, where each row corresponds to the execution of one Jenkins step.
 - Click the row labeled **Stage: stage_name>."Group: <group_name> Run test cases** to view the log for this group's execution.
 - Click the row labeled **Stage: stage_name>." "group_name> Rerun** to display the rerun logs.
- Using Pipeline Steps Page

1. On the Jenkins pipeline page, click the desired build number under the **Build History** drop down.
2. Click the **Pipeline Steps** button on the left pane.
A table with columns for step, arguments, and status appears.
3. Under the Arguments column, find the label for the desired stage and group.
4. Click the step with the label **Stage: <stage_name>Group: <group_name>**.
5. Run test cases under it, or click the **Console output** icon near the status to view the log for this group execution.
6. To see rerun logs, find the step with the label **Stage: <stage_name> Group: <group_name>Rerun** under it.

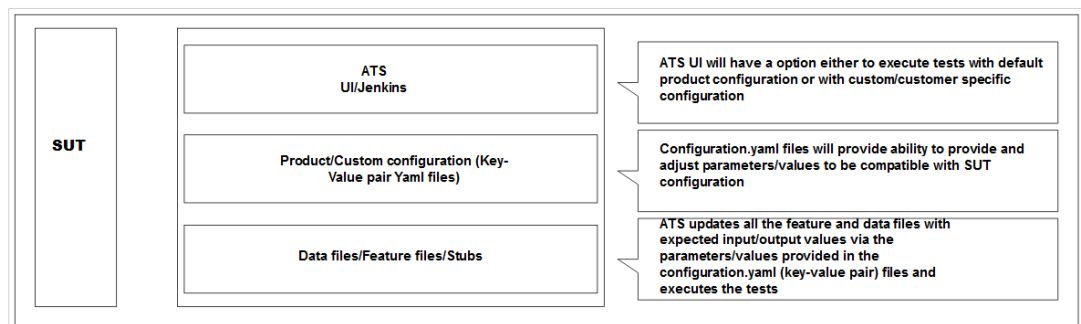
2.17 Parameterization

This feature allows you to provide or adjust values for the input and output parameters needed for the test cases to be compatible with the SUT configuration. You can update or adjust the key-value pair values in the `global.yaml` and `feature.yaml` files for each of the feature files so that they are compatible with SUT configuration. In addition to the existing custom test case folders (Cust New Features, Cust Regression, and Cust Performance), this feature enables folders to accommodate custom data, default product configuration, and custom configuration. You can maintain multiple versions or copies of the custom data folder to suit varied or custom SUT configurations. With this feature, the ATS GUI has the option to either execute test cases with the default product configuration or with a custom configuration.

This feature enables you to perform the following tasks:

- Define parameters and assign or adjust values to make them compatible with SUT configuration.
- Execute test cases either with default product configurations or custom configurations and multiple custom configurations to match varied SUT configurations.
- Assign or adjust values for input or output parameters through custom or default configuration yaml files (key-value pair files).
- Define or adjust the input or output parameters for each feature file with its corresponding configuration.
- Create and maintain multiple configuration files to match multiple SUT configurations.

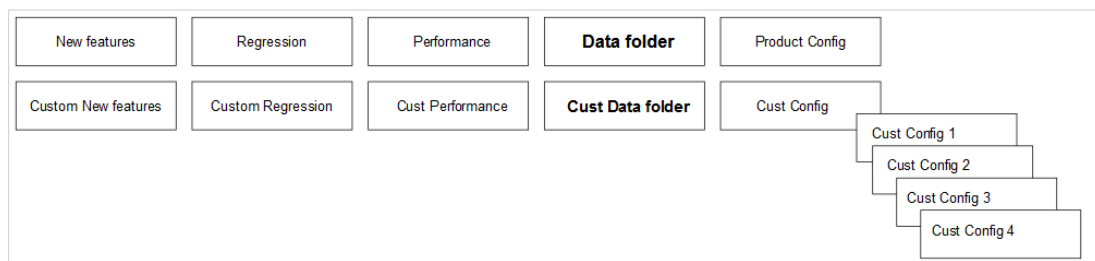
Figure 2-61 SUT Design Summary



In the folder structure:

- The Product Config folder contains default product configuration files (feature-wise yaml per key-value pair), which are compatible with default product configuration.
- New features, Regression and Performance, Data folder, and Product Config folders are replicated or copied into custom folders and delivered as part of the ATS package in every release.
- You can customize custom folders by:
 - Removing test cases not needed or as appropriate for your use.
 - Adding new test cases as needed or as appropriate for your use.
 - Removing or adding data files in the cust_data folder or as appropriate for your use.
 - Adjusting the parameters or values in the key-value pair per yaml file in the custom config folder for test cases to run or pass with a custom configured SUT.
- The product folders are always intact (unchanged) and you can update the Custom folders
- You can maintain multiple copies of Custom Configurations and bring them to use as needed or as appropriate for the SUT configuration.

Figure 2-62 Folder Structure



2.17.1 Running Test Cases

Enable

Rename or copy the Cust Config [1/2/3/N] folder to the Cust Config folder in order for ATS to run the test cases with a specific custom configuration. When the option to run test cases with custom configuration is chosen, it always points to the Cust Config folder.

To Run ATS Test Cases

ATS has the option to run test cases with default or custom configuration.

- If custom configuration is selected, then test cases from custom folders are populated on the ATS UI, and custom configuration is applied to them through the key-value pair per yaml file defined or present in the "Cust Config" folder.
- If product configuration is selected, then the test cases from product folders are populated on the ATS UI, and product configuration is applied to them through key-value pairs per yaml file defined or present in the Product Config folder.

Figure 2-63 ATS Execution Flow

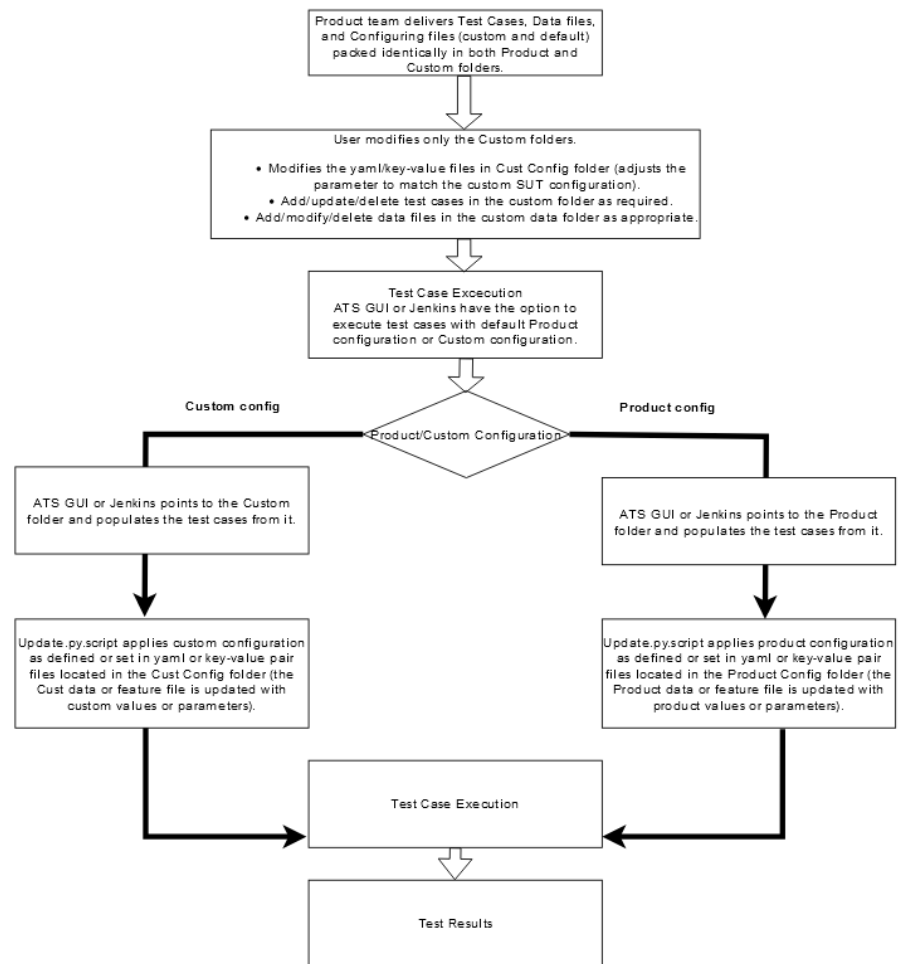


Figure 2-64 Sample: Configuration_Type

Oracle Communications Cloud Native Core

Automated Test Suite - POLICY

EXECUTION OPTIONS

TestSuite Regression	SUT PCF	Configuration_Type Product_Config Product_Config Custom_Config	FilterWithTags No
Include_NewFeatures NO	Fetch_Log_Upon_Failure YES		

2.18 PCAP Log Collection

PCAP Log Collection allows collecting the NF, SUT, or PCAP logs from the debug tool sidecar container. This feature can be integrated and delivered as a standalone or along with the Application Log Collection feature. For information, see [Application Log Collection](#).

PCAP Log Integration

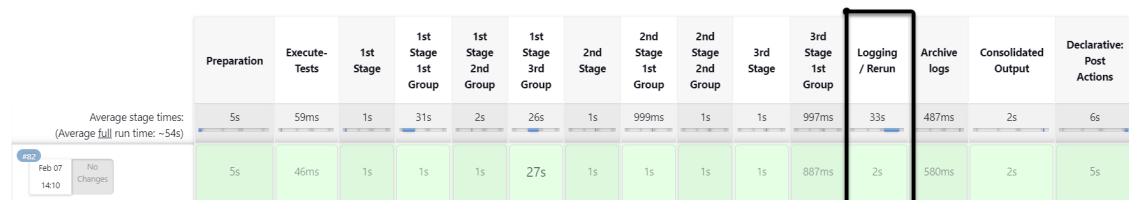
1. The Debug tool should be enabled on SUT Pods while deploying the NF. The name of the Debug container must be "tools".
For example, in SCP, the debug tool should be enabled for all the SCP microservice pods.
2. Update the following parameters in the values.yaml file, under the resource section, with ATS minimum resource requirements:
 - a. CPU: 3
 - b. memory: 3Gi
3. On the home page, click any new feature or regression pipeline.
4. In the left navigation pane, click **Build with Parameters**.
5. Select **YES** from the **Fetch_Log_Upon_Failure** drop-down menu.
6. If option **Log_Type** is available, select value **PcapLog [Debug Container Should be Running]** for it.
7. Select **PcapLog [Debug Container Should be Running]** to activate PCAP Log Collection in ATS-NF.
8. After the build execution is complete, go into the ATS pod, then navigate to below path to find the pcaplogs: `.jenkins/jobs/<Pipeline Name>/builds/<build number>/`
For example, `.jenkins/jobs/SCP-Regression/builds/5/`
Pcaplogs is present in zip form. Unzip it to get the log files.

2.18.1 Application Log Collection and Parallel Test Execution Integration

A new stage, "Logging/Rerun", has been added at the end of the Execute-Tests stage to collect rerun logs, such as applog and PCAP logs, by running the failed test cases in a sequence.

Figure 2-65 Logging/Rerun new stage

Stage View



If the `Fetch_Log_Upon_Failure` parameter is set to YES and if any test case fails in the initial run, then:

- The failed test case reruns and log collection start in the Logging/Rerun stage after the initial run is completed for all the test cases.

- The logs from the initial execution are collected, but they might be incorrect.
- Even if the rerun parameter is set to 0, the failed test case reruns in the Logging/Rerun stage and the log is collected.

Note

Not applicable for all the NFs.

- If the Fetch_Log_Upon_Failure parameter is set to NO and if any test case fails in the initial run, then the failed test case rerun starts in the same stage after the initial execution is over for all the test cases in its group.

2.19 Persistent Volume for 5G ATS

The Persistent Volume (PV) feature allows ATS to retain historical build execution data, test cases, and ATS environment configurations.

ATS Packaging When Using Persistent Volume

- **Without the Persistent Volume option:** ATS package includes an ATS image with test cases.
- **With Persistent Volume option:** ATS package includes the ATS image and test cases separately. The new test cases are provided between the releases.
To support both with and without Persistent Volume options, test cases and execution job data are packaged in the ATS image as well as a tar file.

2.19.1 Processing Flow

First Time Deployment

Initially, when you deploy ATS, for example, the PI-A ATS pod, you use PVC-A, which is provisioned and mounted to the PI-A ATS pod. By default, PVC-A is empty. So, you have to copy the data (ocslf_tests and jobs folders) from the PI-A tar file to the pod after the pod is up and running. Then restart the PI-A pod. At this point, you can change the number of build logs to maintain in the ATS GUI.

Subsequent Deployments

When you deploy ATS for the subsequent time, for example, in a PI-B ATS pod, you use PVC-B, which is provisioned and mounted to the PI-B ATS pod. By default, the PVC-B is empty, and you have to copy the data (ocslf_tests and jobs folders) from the PI-B tar file to the pod after the pod is up and running. At this point, copy all the necessary changes to the PI-B pod from the PI-A pod and restart the PI-B pod. You can change the number of build logs to maintain in the ATS GUI. After updating the number of builds, you can delete the PI-A pod and continue to retain the PVC-A. If you do not want backward porting, you can delete PVC-A.

2.19.2 Deploying Persistent Volume

Preinstallation Steps for Non-OCI setup

1. Before deploying Persistent Volume, create a PVC in the same namespace where you have deployed ATS. You have to provide values for the following parameters to create a PVC:
 - PVC Name

- Namespace
 - Storage Class Name
 - Size of the PV
2. Run the following command to create a PVC:

```
kubectl apply -f - <<EOF

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: <Please Provide the PVC Name>
  namespace: <Please Provide the namespace>
  annotations:
spec:
  storageClassName: <Please Provide the Storage Class Name>
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: <Please Provide the size of the PV>
EOF
```

 Note

It is recommended to suffix the PVC name with the release version to avoid confusion during subsequent releases. For example: ocats-slf-pvc-1.9.0

The output of the above command with parameters is as follows:

```
[cloud-user@atscne-bastion-1 templates]$ kubectl apply -f - <<EOF
>
> apiVersion: v1
> kind: PersistentVolumeClaim
> metadata:
>   name: ocats-slf-1.9.0-pvc
>   namespace: ocslf
>   annotations:
> spec:
>   storageClassName: standard
>   accessModes:
>     - ReadWriteOnce
>   resources:
>     requests:
>       storage: 1Gi
> EOF
```

The persistentvolumeclaim/ocats-slf-1.9.0-pvc is created.

3. To verify whether PVC is bound to PV and is available for use, run the following command:

```
kubectl get pvc -n <namespace used for pvc creation>
```

The output of the above command is as follows:

Figure 2-66 Verifying PVC

```
[cloud-user@atscne-bastion-1 templates]$ kubectl get pvc -n ocslf
NAME                STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
ocats-slf-1.9.0-pvc Bound     pvc-lee50daf-2eb7-4289-b541-51c111270513 1Gi        RWO            standard       8s
policy-pvc-test     Bound     pvc-d346d5c4-129e-4f13-8bd0-a7b7555e0e2f 1Gi        RWO            standard       31d
slf-pvc             Bound     pvc-0b0bce8c-b856-4e0e-b4d8-0f981250adbd 1Gi        RWO            standard       12d
```

Check that the STATUS is **Bound** and that the rest of the parameters, such as NAME, CAPACITY, ACCESS MODES, STORAGECLASS, and so on, are the same as specified in the PVC creation command.

Note

Do not proceed further if there is any issue with PVC creation. Contact your administrator to create a PV.

4. After creating persistent volume, change the following parameters in the values.yaml file to deploy persistent volume.
 - Set the **PVEnabled** parameter to "true".
 - Provide the value for the **PVClaimName** parameter. The PVClaimName value should be the same as the value used to create a PVC.

Preinstallation Steps for OCI setup

1. Before deploying persistent volumes, create a PVC in the same namespace where you have deployed ATS. You have to provide values for the following parameters to create a PVC:
 - PVC Name
 - Namespace
 - Storage Class Name
 - Size of the PV
2. Run the following command to create a PVC:

```
kubectl apply -f - <<EOF
```

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: <Please Provide the PVC Name>
  namespace: <Please Provide the namespace>
  annotations:
spec:
  storageClassName: <Please Provide the Storage Class Name>
  accessModes:
    - ReadWriteOnce
```

```
resources:
  requests:
    storage: <Please Provide the size of the PV>
EOF
```

- When creating Persistent Volume Claim in OCI, add the storageClassName "oci-bv" and maintain the 50 gigabyte storage value.

Note

The minimum allowed value for parameter "resources.requests.storage" is 50Gi. If the storage is set to any value less than 50Gi, it will be ignored.

For example, the output of the above command with parameters is as follows:

```
[cloud-user@atscne-bastion-1 templates]$ kubectl apply -f - <<EOF
>
> apiVersion: v1
> kind: PersistentVolumeClaim
> metadata:
>   name: ocats-slf-1.9.0-pvc
>   namespace: ocslf
>   annotations:
> spec:
>   storageClassName: oci-bv
>   accessModes:
>     - ReadWriteOnce
>   resources:
>     requests:
>       storage: 50Gi
> EOF
```

The persistentvolumeclaim/ocats-slf-1.9.0-pvc is created.

- To verify whether PVC is bound to PV and is available for use, run the following command:

```
kubectl get pvc -n <namespace used for pvc creation>
```

The output of the above command is as follows:

Figure 2-67 Verifying PVC

```
[opc@operatorinstance Scripts]$ kubectl get pvc ocsepp3
NAME                                STATUS    VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
ocats-sepp-24.1.0-pvc              Bound    csl-48a73509-c08a-4ff1-ba2e-c2a6e1245cd9   50Gi       RWO            oci-bv         5m16s
```

- After creating persistent volume, the STATUS will be Pending and when the ATS pod is deployed with that PVC, its status will become "BOUND".
- Change the following parameters in the values.yaml file to deploy persistent volume.
 - Set the **PVEnabled** parameter to "true".
 - Provide the value for the **PVClaimName** parameter. The PVClaimName value should be the same as the value used to create a PVC.

Postinstallation Steps

1. After deploying ATS, copy the <nf_main_folder> and <jenkins jobs> folders from the tar file to their ATS pod, and then restart the pod as a one-time activity.

- a. Run the following command to extract the tar file:

```
ocats-<nf_name>-data-<release-number>.tgz
```

Note

The ats_data.tar file is the name of the tar file containing <nf_main_folder> and jobs folders. It can be different for different NFs.

- b. Run the following set of commands to copy the required folders:

```
kubectl cp ats_data/jobs <namespace>/<pod-name>:/var/lib/  
jenkins/.jenkins/
```

```
kubectl cp ats_data/<nf_main_folder> <namespace>/<pod-name>:/var/lib/  
jenkins/
```

- c. Run the following command to restart the pods as one-time activity:

```
kubectl delete po <pod-name> -n <namespace>
```

Note

Before running the following command, copy the changes done on the new release pod from the old release pod using the `kubectl cp` command.
[Applicable in the case of subsequent deployment only]

2. When the pod is up and running, log in to the ATS GUI and go to your NF specific pipeline. Click **Configure** in the left navigation pane. The **General** tab appears. Configure the **Discard old Builds** option. This option allows you to configure the number of builds you want to retain in the persistent volume.

Figure 2-68 Discard Old Builds

☒ Discard old builds ?

Strategy: Log Rotation

Days to keep builds:

if not empty, build records are only kept up to this number of days

Max # of builds to keep: 4

if not empty, only up to this number of build records are kept

Advanced...

Note

It is recommended to configure this option. If you do not input a value for this option, the application will take into account all builds, which could be a large number, and will completely consume the Persistent Volume.

2.19.3 Backward Porting

The following deployment steps apply to the old release of PVC-supported ATS Pod.

Prerequisite: You should have the old PVC that contains the old release of POD data.

Note

This procedure is for backward porting purposes only and should not be considered a subsequent release of the POD deployment procedure.

The deployment procedure for the old release PVC-supported ATS pod is the same; however, while deploying the ATS pod, you have to update the values.yaml file with the following:

- Change the **PVEnabled** parameter to "true".
- Provide the name of the old PVC as the value for parameter **PVClaimName**.

2.20 Single Click Job Creation

With the help of Single Click Job Creation feature, ATS users can easily create a job to run TestSuite with a single click.

2.20.1 Configuring Single Click Job

Prerequisite: The network function specific user should have 'Create Job' access.

Perform the following procedure to configure the single-click feature:

1. Log in to ATS using network function specific log-in credentials..
2. Click **New Item** in the left navigation pane of the ATS application. The following page appears:

Figure 2-69 New Item Window

3. In the **Enter an item name** text box, enter the job name. Example: <NF-Specific-name>-NewFeatures.
4. In the **Copy from** text box, enter the actual job name for which you need single-click execution functionality. Example: <NF-Specific-name>-NewFeatures.
5. Click **OK**. You are automatically redirected to edit the newly created job's configuration.
6. Under the **General** group, deselect the **This Project is Parameterised** option.
7. Under the **Pipeline** group, make the corresponding changes to remove the 'Active Choice Parameters' dependency.
8. Provide the default values for the **TestSuite**, **SUT**, **Select_Option**, **Configuration_Type**, and other parameters, as required, on the **BuildWithParameters** page.
Example: Pipeline without Active Choice Parameter Dependency

```
node ('built-in'){
  //a = SELECTED_NF      b = PCF_NAMESPACE      c = PROMSVC_NAME      d
= GOSTUB_NAMESPACE
  //e = SECURITY        f = PCF_NFINSTANCE_ID    g = POD_RESTART_TIME  h
= POLICY_TIME
  //i = NF_NOTIF_TIME   j = RERUN_COUNT        k =INITIALIZE_TEST_SUITE
l = STUB_RESPONSE_TO_BE_SET
  //m = POLICY_CONFIGURATION_ADDITION          n = POLICY_ADDITION
o = NEXT_MESSAGE
  //p = PROMSVCIP      q = PROMSVCPORT          r = TIME_INT_POD_DOWN    s
= POD_DOWN_RETRIES
  //t = TIME_INT_POD_UP   u = POD_UP_RETRIES   v = ELK_WAIT_TIME    w =
ELK_HOST
  //x = ELK_PORT   y = STUB_LOG_COLLECTION z = LOG_METHOD A =
enable_snapshot B = svc_cfg_to_be_read C = PCF_API_ROOT
```

```

//Description of Variables:

//SELECTED_NF : PCF
//PCF_NAMESPACE : PCF Namespace
//PROMSVC_NAME : Prometheus Server Service name
//GOSTUB_NAMESPACE : Gostub namespace
//SECURITY : secure or unsecure
//PCF_NFINSTANCE_ID : nfInstanceId in PCF application-config config map
//POD_RESTART_TIME : Greater or equal to 60
//POLICY_TIME : Greater or equal to 120
//NF_NOTIF_TIME : Greater or equal to 140
//RERUN_COUNT : Rerun failing scenario count
//TIME_INT_POD_DOWN : The interval after which we check the POD status
if its down
//TIME_INT_POD_UP : The interval after which we check the POD status
if its UP
//POD_DOWN_RETRIES : Number of retry attempt in which will check the
pod down status
//POD_UP_RETRIES : Number of retry attempt in which will check the
pod up status
//ELK_WAIT_TIME : Wait time to connect to Elastic Search
//ELK_HOST : Elastic Search HostName
//ELK_PORT : Elastic Search Port
//STUB_LOG_COLLECTION : To Enable/Disable Stub logs collection
//LOG_METHOD : To select Log collection method either elasticsearch or
kubernetes
//enable_snapshot: Enable or disable snapshots that are created at the
start and restored at the end of each test run
//svc_cfg_to_be_read: Timer to wait for importing service
configurations
//PCF_API_ROOT: PCF_API_ROOT information to set Ingress gateway
service name and port
withEnv([
'TestSuite=NewFeatures',
'SUT=PCF',
'Select_Option=All',
'Configuration_Type=Custom_Config'
]){
sh '''
    sh /var/lib/jenkins/ocpcf_tests/preTestConfig-NewFeatures-PCF.sh \
    -a PCF \
    -b ocpcf \
    -c occne-prometheus-server \
    -d ocpcf \
    -e unsecure \
    -f fe7d992b-0541-4c7d-ab84-c6d70b1b0123 \
    -g 60 \
    -h 120 \
    -i 140 \
    -j 2 \
    -k 0 \
    -l 1 \
    -m 1 \
    -n 15 \
    -o 1 \
    -p occne-prometheus-server.occne-infra\

```

```

-q 80\
-r 30\
-s 5\
-t 30\
-u 5\
-v 0\
-w occne-elastic-elasticsearch-master.occne-infra\
-x 9200\
-y yes\
-z kubernetes\
-A no\
-B 15\
-C ocpcf-occpn-ingress-gateway:80\

...
    load "/var/lib/jenkins/ocpcf_tests/jenkinsData/Jenkinsfile-Policy-
NewFeatures"
  }
}

```

9. Click **Save**. The ATS application is ready to run TestSuite with 'SingleClick' using the newly created job.

2.21 Support for ATS Deployment in OCI

Oracle Cloud Infrastructure (OCI) is a set of complementary cloud services that enable you to build and run a range of applications and services in a highly-available, consistently high-performance environment. OCI offers powerful computing capabilities and storage capacity in a flexible virtual network that can be accessed from your on-premises network.

OCI infrastructure consists of Compartments, Network Load balancer, Bastion Host, Dynamic Routing Gateway (DRG), Remote Peering Connection (RPC), Service Gateway, Internet Gateway, and Oracle Kubernetes Engine (OKE) cluster, all in an OKE VCN.

ATS deployment is also supported in OCI.

2.21.1 Accessing ATS GUI in OCI

The following two ways are supported to access the ATS GUI in OCI:

Using Loadbalancer IP

1. Add proper Ingress or Egress security rules for ports (8080/8443) and ATS service Node port in loadbalancer (nf_lb_subnet) and Node subnet (nf_node_subnet). To add ingress and egress security rules, see the [Adding Ingress and Egress Rules to Access the OCI Console](#).
2. Insert the following annotations under the **Metadata** section in ATS service object to assign an external IP (Loadbalancer IP):

```

oci-network-load-balancer.oraclecloud.com/security-list-management-mode:
None
oci.oraclecloud.com/load-balancer-type: nlb

```

Note

This can be done by editing and saving ATS service object after ATS deployment:
For example,

```
kubectl edit svc ats-service-name -n ats-namespace
```

3. Access the GUI using URL: <http/https>://<Loadbalancer IP>:<8080/8443>

Note

- To open ATS GUI with https (TLS enabled), Load balancer IP must be known before deployment and listed down in the alt_names section of the ssl.conf file while generating the application or client certificate. Otherwise, the ATS GUI will open, but it will show "Not Secure" in the browser address bar. For more information, see the [Support for Transport Layer Security](#) section.
- The assignment of Loadbalancer IP to the ATS service is subject to availability. If the Loadbalancer IP is not assigned to the ATS service even after applying the required annotations, try to debug on the OCI side.

Using Tunneling

1. Add an ingress security rule for the node subnet (nf_node_subnet) to allow TCP traffic on all ports from the operator subnet. To add ingress and egress security rules, see the [Adding Ingress and Egress Rules to Access the OCI Console](#) section.
2. Run the following command from a bash terminal on your local PC:

```
ssh -f -N -i <operator instance private key> -o StrictHostKeyChecking=no -o ProxyCommand="ssh -i <bastion private key> -o StrictHostKeyChecking=no -W %h:%p <bastion username>@<bastion IP>" <operator instance username>@<operator instance IP> -L <local PC port>:<Worker Node IP>:<ATS NodePort> -o ServerAliveInterval=60 -o ServerAliveCountMax=300
```

3. Access the GUI using URL: <http/https>://<localhost/127.0.0.1>:<local PC port>
For example, ssh -f -N -i id_rsa -o StrictHostKeyChecking=no -o ProxyCommand="ssh -i id_rsa -o StrictHostKeyChecking=no -W %h:%p opc@129.287.66.123" opc@10.1.76.7 -L 5009:10.9.60.118:32071 -o ServerAliveInterval=60 -o ServerAliveCountMax=300

Here, ATS GUI URL is http://localhost:5009

2.21.2 Adding Ingress and Egress Rules to Access the OCI Console

Perform the following procedure to add ingress and egress rules:

1. Log in to the OCI Console using your login credentials.
2. On the OCI Console Dashboard, click **Navigation menu**.

Figure 2-70 Navigation menu



- Click **Networking**, and then click **Virtual cloud networks** on the preview pane.

Figure 2-71 Virtual cloud networks

- Click the required OKE cluster VCN.
A list of all subnets in that VCN is displayed.
- Save the value of the "IPv4 CIDR Block" field for **nf_lb_subnet**.
This information can be used in the subsequent steps as value of <CIDR for Load balancer subnet>.
- Save the value of the "IPv4 CIDR Block" field for **nf_node_subnet**.
This information can be used in the subsequent steps as value of <CIDR for Node subnet>.

Updating Security List for Node Subnet

- To update security list for Node subnet, do the following:
 - Navigate to **nf_node_subnet**, and then to **nf_node_security_list**.
 - Click **Add Ingress Rules**.

Figure 2-72 Add Ingress Rules

- Add ingress rule if it is unavailable to access the OCI Console in **nf_node_subnet** as described in the following table:

Table 2-9 Adding Ingress Rules for Node Subnet

Option Name	Description
Source Type	Indicates the source type. Value: CIDR
Source CIDR	Indicates the source of CIDR. Value: <CIDR for Load balancer subnet> Note: The value for the variable <CIDR for Node subnet> was determined in a previous step.
IP Protocol	Indicates the internet protocol for which this rule is applicable. Value: TCP
Source Port Range	Keep this option at its default value.
Destination Port Range	Keep this option at its default value.
Description	Keep this option at its default value.

Note

To access ATS API, you can use the above mentioned ingress rule.

Updating Security List for Load Balancer Subnet

1. To update security list for Load balancer subnet, do the following:
 - a. Navigate to **nf_lb_subnet**, and then to **nf_lb_security_list**.
 - b. Click **Add Ingress Rules**.
 - c. Add ingress rule if it is unavailable to access the OCI Console in **nf_lb_subnet** by configuring the options as described in the following table:

Table 2-10 Adding Ingress Rules Load Balancer Subnet

Option Name	Description
Source Type	Indicates the source type. Value: CIDR
Source CIDR	Indicates the source of CIDR. Value: <CIDR for Network which will be used to access ATS GUI> Note: Avoid setting its value to 0.0.0.0/0, as it may result in granting access to ATS GUI from all networks, posing a considerable security risk. If you intend for ATS GUI to only be accessible within the Corporate Proxy (VPN), using a Source CIDR value of 0.0.0.0/0 would allow access from outside the Corporate Proxy (VPN) as well.
IP Protocol	Indicates the internet protocol for which this rule is applicable. Value: TCP
Source Port Range	Keep this option at its default value.
Destination Port Range	Indicates the ATS GUI port number. Port 8080 is used for HTTP and 8443 is used for HTTPS. Value: <ATS service Port>
Description	Keep this option at its default value.

- d. To access ATS API, create one more ingress rule if it is unavailable by following the previous step with the following changes:

- In the **Source CIDR** option, provide the <CIDR for Network which will be used to access ATS API> value.
 - In the **Destination Port Range** option, provide the <ATS Service API Port> value, which refers to the ATS API port, that is, 5001.
2. To add egress rule if it is unavailable to access the OCI Console in **nf_lb_subnet**, do the following:
 - a. In the left navigation pane, click **Egress Rules**.
 - b. Click **Add Egress Rules**.
 - c. Configure the following options as described in this table:

Table 2-11 Adding Egress Rules Load Balancer Subnet

Option Name	Description
Destination Type	Indicates the destination type. Value: CIDR
Destination CIDR	Indicates the destination of CIDR. Value: <CIDR for Node subnet> Note: The value for the variable <CIDR for Node subnet> was determined in the previous step.
IP Protocol	Indicates the internet protocol for which this rule is applicable. Value: TCP
Source Port Range	Keep this option at its default value.
Destination Port Range	Keep this option at its default value.
Description	Keep this option at its default value.

Note

To access ATS API, you can use the above mentioned egress rule.

Updating Security List for Tunneling

1. Log in to the OCI Console using your login credentials.
2. On the OCI Console Dashboard, click **Navigation menu**.

Figure 2-73 Navigation menu

3. Click **Networking**, and then click **Virtual cloud networks** on the preview pane.
4. Click the required OKE cluster VCN.
A list of all subnets in that VCN is displayed.
5. Save the value of the "IPv4 CIDR Block" field for **nf_operator_subnet**.
This information can be used in the subsequent steps as value of <CIDR for Operator subnet>.

Updating Security list of Node Subnet

1. To update security list of Node subnet, do the following:
 - a. Navigate to **nf_node_subnet**, and then to **nf_node_security_list**.
 - b. Click **Add Ingress Rules**.
 - c. Add ingress rule if it is unavailable to access the OCI Console in **nf_node_subnet** by configuring the options as described in the following table:

Table 2-12 Adding Ingress Rules for Node Subnet Tunneling

Option Name	Description
Source Type	Indicates the source type. Value: CIDR
Source CIDR	Indicates the source of CIDR. Value: <CIDR for Operator subnet> Note: The value for the variable <CIDR for Operator subnet> was determined in the previous step.
IP Protocol	Indicates the internet protocol for which this rule is applicable. Value: TCP
Source Port Range	Keep this option at its default value.
Destination Port Range	Keep this option at its default value.
Description	Keep this option at its default value.

Note

To access ATS API, you can use the above mentioned ingress rule.

2.22 Support for Transport Layer Security

Currently, ATS is accessible through HTTP, which can raise security risks.

With the support of the TLS feature, Jenkins servers have been upgraded to support HTTPS, ensuring a secure and encrypted connection when accessing the ATS dashboard.

To provide encryption, HTTPS uses an encryption protocol known as Transport Layer Security (TLS), which is a widely accepted standard protocol that provides authentication, privacy, and data integrity between two communicating computer applications.

Now, users can access the ATS GUI with the HTTPS protocol instead of the previously used HTTP protocol.

Figure 2-74 Access with TLS

← → ↻ https://100.77.50.249:31311/login?from=%2F Guest

Oracle Communications Cloud Native Core - Automated Test Suite

Welcome to ATS!

Username

Password

☐ Keep me signed in

Sign in

Note

If this feature is not enabled before installation, HTTP will continue to operate.

2.22.1 Deploy ATS with TLS Enabled

Follow the steps in this section to create a Java KeyStore (JKS) file and enable the ATS GUI with HTTPS during installation.

2.22.1.1 Generate JKS File for Jenkins Server

A JKS file needs to be created in order for Jenkins to provide ATS GUI access through HTTPS.

Perform the following steps to generate a JKS file:

Generate the Root Certificate

If a root certificate, for example, `caroot.cert`, is not already available, a user can generate one. Users may use their own files if they have a CA-signed root certificate and key or their own root certificates. The root certificate is used to sign the application, or ATS certificate.

Follow the steps to create and use self-signed certificates:

1. Generate a root key with the following command:

```
openssl genrsa 2048 > <path_to_root_key>
```

For example,

```
openssl genrsa 2048 > caroot.key
```

2. Generate a "caroot" certificate with the following command:

```
openssl req -new -x509 -nodes -days 1000 -key
    <path_to_root_key> > <path_to_root_certificate>
```

For example,

```
[cloud-user@star23-bastion-1 cert]$ openssl req -new -x509 -nodes -days
1000 -key caroot.key > caroot.cer
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [XX]:IN
State or Province Name (full name) []:KA
Locality Name (eg, city) [Default City]:BLR
Organization Name (eg, company) [Default Company Ltd]:ORACLE
Organizational Unit Name (eg, section) []:CGBU
Common Name (eg, your name or your server's hostname) []:ocats
Email Address []:
[cloud-user@star23-bastion-1 cert]$
```

Generate Application or Client Certificate

Follow the steps to create and edit the `ssl.conf` file:

1. In the **alt_names** section, list the IPs, such as IP.1, IP.2, and so on, that are used to access the ATS GUI:

```
[ req ]
default_bits          = 4096
distinguished_name    = req_distinguished_name
req_extensions        = req_ext

[ req_distinguished_name ]
countryName           = Country Name (2 letter code)
countryName_default   = <Country_Name>
stateOrProvinceName   = State or Province Name (full name)
stateOrProvinceName_default = <State_Name>
localityName          = Locality Name (eg, city)
localityName_default  = <Locality_Name>
organizationName      = Organization Name (eg, company)
organizationName_default = <Org_Name>
commonName            = Common Name (e.g. server FQDN or YOUR name)
commonName_max        = 64
commonName_default    = <helm_name>.<namespace>.svc.cluster.local

[ req_ext ]
keyUsage = critical, digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth, clientAuth
basicConstraints = critical, CA:FALSE
subjectAltName = critical, @alt_names
```

```
[alt_names]
IP.1 = 127.0.0.1
IP.2 = <IP1>
IP.3 = <IP2>
DNS.1 = <helm_name>.<namespace>.svc.cluster.local
```

For example,

```
[ req ]
default_bits          = 4096
distinguished_name = req_distinguished_name
req_extensions        = req_ext

[ req_distinguished_name ]
countryName            = Country Name (2 letter code)
countryName_default    = <Country_Name>
stateOrProvinceName    = State or Province Name (full name)
stateOrProvinceName_default = <State_Name>
localityName           = Locality Name (eg, city)
localityName_default   = <Locality_Name>
organizationName       = Organization Name (eg, company)
organizationName_default = <Org_Name>
commonName             = Common Name (e.g. server FQDN or YOUR name)
commonName_max         = 64
commonName_default     = ocats.scpsvc.svc.cluster.local

[ req_ext ]
keyUsage = critical, digitalSignature, keyEncipherment
extendedKeyUsage = serverAuth, clientAuth
basicConstraints = critical, CA:FALSE
subjectAltName = critical, @alt_names

[alt_names]
IP.1 = 127.0.0.1
IP.2 = 10.75.217.5
IP.3 = 10.75.217.76
DNS.1 = localhost
DNS.2 = ocats.scpsvc.svc.cluster.local
```

Note

- To access the GUI with DNS, make sure that the `commonName_default` is the same as the DNS name being used.
 - The `/etc/hosts` file in the local system should also be updated with the assigned IP and the DNS mentioned above to open the ATS with DNS. Otherwise, it is not required.
- Ensure the DNS is in this format:
`<service_name>.<namespace>.svc.cluster.local`
- Multiple DNSs, such as `DNS.1`, `DNS.2`, and so on, can be added.
- To support the ATS API, add the IP `127.0.0.1` to the list of IPs.

2. Create a Certificate Signing Request (CSR) with the following command:

```
openssl req -config ssl.conf -newkey rsa:2048 -days 1000 -nodes -keyout
<path_to_application_certificate_key> >
<path_to_certificate_signing_request>
```

For example,

```
[cloud-user@star23-bastion-1 cert]$ openssl req -config ssl.conf -newkey
rsa:2048 -days 1000 -nodes -keyout rsa_private_key_pkcs1.key >
ssl_rsa_certificate.csr
Ignoring -days; not generating a certificate
Generating a RSA private key
...+++++
.....+++++
writing new private key to 'rsa_private_key_pkcs1.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [IN]:
State or Province Name (full name) [KA]:
Locality Name (eg, city) [BLR]:
Organization Name (eg, company) [ORACLE]:
Common Name (e.g. server FQDN or YOUR name) [ocats]:
[cloud-user@star23-bastion-1 cert]$
```

3. Use the command to display the components of the file and verify the configurations:

```
openssl req -text -noout -verify -in
ssl_rsa_certificate.csr
```

4. Sign in to this CSR file with the root certificate:

```
openssl x509 -extfile ssl.conf -extensions req_ext -req -in
<path_to_certificate_signing_request> -days 1000 -CA
<path_to_root_certificate> -CAkey <path_to_root_key> -set_serial 04 >
<path_to_application_certificate>
```

For example,

```
[cloud-user@star23-bastion-1 cert]$ openssl x509 -extfile ssl.conf -
extensions req_ext -req -in ssl_rsa_certificate.csr -days 1000 -CA ../
caroot.cer -CAkey ../caroot.key -set_serial 04 > ssl_rsa_certificate.crt
Signature ok
subject=C = IN, ST = KA, L = BLR, O = ORACLE, CN = ocats
Getting CA Private Key
[cloud-user@star23-bastion-1 cert]$
```

5. Use the command to verify that the certificate is properly signed by the root certificate:

```
openssl verify -CAfile <path_to_root_certificate>
               <path_to_application_certificate>
```

For example,

```
[cloud-user@star23-bastion-1 cert]$ openssl verify -CAfile caroot.cer
ssl_rsa_certificate.crt
ssl_rsa_certificate.crt: OK
```

6. Save the generated application certificates and the root certificates.
7. Add the caroot.cer to the browser as a trusted author. For more information, see [Enable ATS GUI with HTTPS](#)
8. Generate the .p12 keystore file with the following command:

```
openssl pkcs12 -inkey <path_to_application_key> -in
               <path_to_application_certificate> -export -out <path_to_p12_certificate>
```

For example,

```
[cloud-user@star23-bastion-1 ocats]$ openssl pkcs12 -inkey
rsa_private_key_pkcs1.key -in ssl_rsa_certificate.crt -export -out
certificate.p12
Enter Export Password:
Verifying - Enter Export Password:
```

9. In the prompt, create a password and save it for future use.
10. Convert the .p12 keystore file into a JKS format file:

```
keytool -importkeystore -srckeystore <path_to_p12_certificate> -
srcstoretype pkcs12 -destkeystore <path_to_jks_file> -deststoretype JKS
```

For example,

```
[cloud-user@star23-bastion-1 cert]$ keytool -importkeystore -srckeystore ./
certificate.p12 -srcstoretype pkcs12 -destkeystore jenkinsserver.jks -
deststoretype JKS
Importing keystore ./certificate.p12 to jenkinsserver.jks...
Enter destination keystore password:
Re-enter new password:
Enter source keystore password:
Entry for alias 1 successfully imported.
Import command completed: 1 entries successfully imported, 0 entries
failed or cancelled
```

11. In the prompt, use the same password used while creating .p12 keystore file.

Note

Verify that the passwords linked to the .p12 keystore and JKS files are the same.

12. The generated file, `jenkinserver.jks`, is provided to the Jenkins server.

2.22.1.2 Enable ATS GUI with HTTPS

Follow the steps to secure or enable TLS on the server.

1. Create a Kubernetes secret by adding the above created files:

```
kubectl create secret generic <secret_name> --from-file=<path_to_jks_file>
--from-file=<path_to_application_certificate> --from-
file=<path_to_application_key> --from-file=<path_to_root_certificate> -n
<namespace>
```

For example,

```
kubectl create secret generic ocats-tls-secret --from-
file=jenkinserver.jks --from-file=ssl_rsa_certificate.crt --from-
file=rsa_private_key_pkcs1.key --from-file=caroot.cer -n scpsvc
```

The sample of created secret:

```
[cloud-user@star23-bastion-1 cert]$ kubectl describe secret ocats-tls-
secret -n scpsvc
Name:          ocats-tls-secret
Namespace:     scpsvc
Labels:        <none>
Annotations:   <none>
```

Type: Opaque

Data

====

```
caroot.cer:          1147 bytes
ssl_rsa_certificate.crt: 1424 bytes
jenkinserver.jks:    2357 bytes
rsa_private_key_pkcs1.key: 1675 bytes
```

2. Apply the changes in `values.yaml` file, the user can install the ATS using the `helm install` command. Change the `atsGuiTLSEnabled` parameter value to true.
3. A user can now start ATS with HTTPS the protocol.
The link to open the ATS GUI format is `https://<IP>:<port>`, for example, `https://10.75.217.25:30301`

The lock symbol in the browser indicates that the server is secured or TLS enabled.

Adding a Certificate in Browser

The created root certificate is added to the browser, either Mozilla Firefox or Chrome.

Note

Future versions of these browsers may involve different menu options. For more information on importing root certificate, see the browser documentation to add a self-signed certificate to the browser as a trusted certificate.

Adding in Windows laptops

1. In the Chrome browser, navigate to the settings and search for security.
2. Click the **security** option that appears next to search.
3. Click the **Manage Device Certificate** option. The Certificate window opens.
4. Click the Trusted root certification authorities bar.
5. Import the caroot certificate.
6. Save and restart the browser.

Adding in Mac laptop

1. In the Chrome browser, navigate to the settings and search for security.
2. Click the **security** option that appears next to search.
3. Click the **Manage Device Certificate** option. The Keychain Access window opens.
4. Search the tab certificate and drag and drop the downloaded caroot certificate.
5. Find the uploaded certificate in the list, usually listed by a temporary name.
6. Double click the certificate and expand the Trust option.
7. In **When using this certificate** option, assign it to "always trust".
8. Close the window and validate if it asks for the password.
9. Save and restart the browser.

Adding for Windows and Mac laptop

1. In the Mozilla Firefox browser, navigate to the settings and search for certificates.
2. Click the **View Certificate** that appears next to search. This opens a Certificate Manager window.
3. Navigate to the Authorities section, click the **Import** button, and upload the caroot certificate.
4. Click the **Trust** options in the pop-up window and click OK.
5. Save and restart the browser.

2.23 Test Results Analyzer

The Test Results Analyzer is a plugin available in ATS to view pipeline test results based on XML reports. It provides the test results report in a graphical format, which includes consolidated and detailed stack trace results in case of any failures. It allows you to navigate to each and every test.

The test result report shows any one of the following statuses for each test case:

- **PASSED:** If the test case passes.

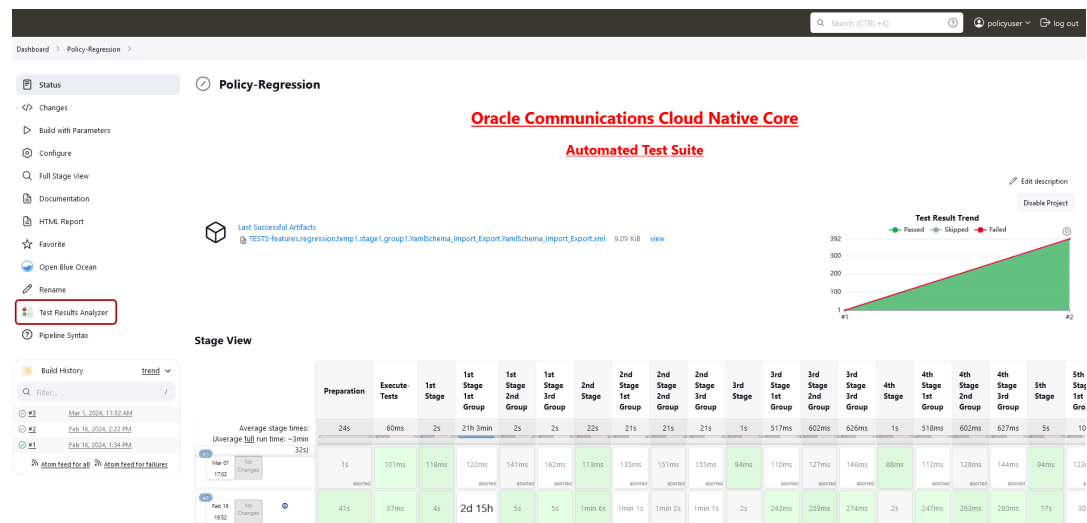
- **FAILED:** If the test case fails.
- **SKIPPED:** If the test case is skipped.
- **N/A:** If the test cases are not executed in the current build.

2.23.1 Accessing Test Results Analyzer Feature

To access the test results analyzer feature:

1. From the NF home page, click any new feature pipeline or regression pipeline where you want to run this plugin.
2. In the left navigation pane, click **Test Results Analyzer**.

Figure 2-75 Test Results Analyzer Option



When the build completes, the test result report appears. A sample test result report is shown below:

Figure 2-76 Sample Test Result Report

Chart	Package/Class/Testmethod	Passed	Transitions	5	4	3	2	1
☐	features.cust_newfeatures.temp1.Binding_Service.Binding_Service	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.Bulk_Import_Export.Bulk_Import_Export	100% (100%)	0	PASSED	PASSED	SKIPPED	SKIPPED	N/A
☐	features.cust_newfeatures.temp1.Diameter_GW_Error_Handling.Diameter_GW_Error_Handling	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.Dynamic_Policy_Override.Dynamic_Policy_Override.This feature aims to dynamically override the parameters of pcc and session rules during SM Policy Association	100% (100%)	0	PASSED	PASSED	SKIPPED	SKIPPED	N/A
☐	features.cust_newfeatures.temp1.IMS_Emergency_Call.IMS_Emergency_Call	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.Operators.Operators	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.PolicyActionForBulkPccRuleRemoval.Policy_Action_Bulk_Pccrule_Removal	100% (100%)	0	PASSED	PASSED	SKIPPED	SKIPPED	N/A
☐	features.cust_newfeatures.temp1.Policy_Conditions_And_Actions.Policy_Conditions_And_Actions	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.Policy_Table.Policy_Table	100% (100%)	0	PASSED	PASSED	PASSED	SKIPPED	N/A
☐	features.cust_newfeatures.temp1.Request_AVF.Request_AVF	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.Sanity.Sanity	0% (0%)	0	FAILED	N/A	N/A	N/A	N/A
☐	features.cust_newfeatures.temp1.SessionRetry.SessionRetry.To verify that SM or User-service retry to alternate FQDN if the	0%	0	FAILED	FAILED	SKIPPED	FAILED	N/A

3. Click any one of the statuses (PASSED, FAILED, SKIPPED) to view the respective feature detail status report.

Note

For N/A status, a detailed status report is not available.

Figure 2-77 Test Result

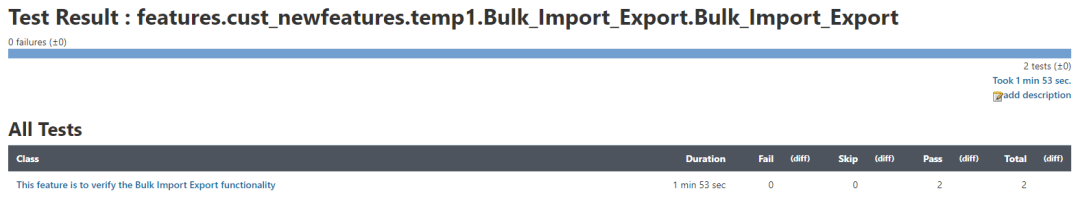
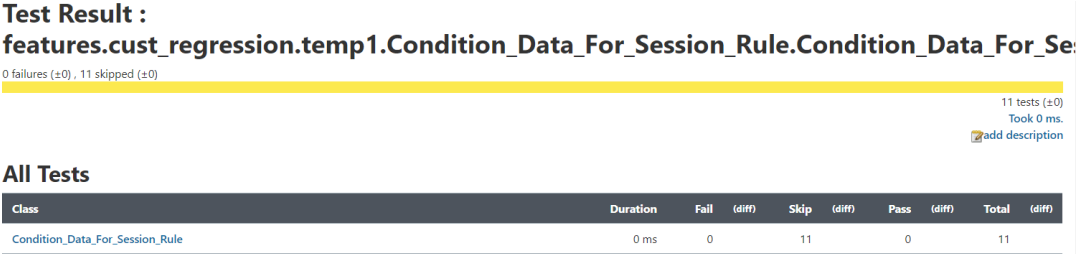


Figure 2-78 Test Result



4. In the case of a rerun, the test cases that passed in the initial run but were skipped in the rerun are considered "passed" in the Test Results Analyzer Report. The following screenshot depicts the scenario:
"Variant2_equal_smPolicySnssaiData,Variant2_exist_smPolicyData,Variant2_exist_smPolicyDnnData_dnn" where the test cases passed in the initial run but skipped in the rerun are considered "passed" in general.

Figure 2-79 Test Results

☐	● Variant2_UDR	13% (20%)	2	FAILED
☐	Variant2_Multi_Block_In_Different_Position	0% (0%)	0	FAILED
☐	Variant2_contain_smPolicyData	0% (0%)	0	FAILED
☐	Variant2_contain_smPolicyDnnData_dnn	0% (0%)	0	FAILED
☐	Variant2_contain_smPolicySnssaiData	0% (0%)	0	FAILED
☐	Variant2_equal_smPolicyDnnData_dnn	0% (0%)	0	FAILED
☐	Variant2_equal_smPolicySnssaiData	100% (100%)	0	PASSED
☐	Variant2_exist_smPolicyData	100% (100%)	0	PASSED
☐	Variant2_exist_smPolicyDnnData_dnn	100% (100%)	0	PASSED
☐	Variant2_exist_smPolicySnssaiData	0% (0%)	0	FAILED

5. Click **PASSED**. The following highlighted message means the test case was passed in the main run but skipped in the rerun.

Figure 2-80 Test Result Info

Passed

features.cust_regression.temp1.Variant2.Variant2.Variant2_UDR.Variant2_equal_smPolicySnssaiData

Standard Output

Passed in Initial Run, hence Skipped here. Please see console logs for more information

@scenario.begin

@Variant2_equal_smPolicySnssaiData

Scenario: Variant2_equal_smPolicySnssaiData

Given Initialize Test Suite ... skipped in 0.000s

Then Waiting for test_suite_to_initialize ... skipped in 0.000s

Then Wait 5 ... skipped in 0.000s

@scenario.end

2.24 Support for Test Case Mapping and Count

The Test Case Mapping and Count feature displays the total number of features, test cases, or scenarios and their mapping to each feature in the ATS GUI.

2.24.1 Access Test Case Mapping and Count Feature

1. On the NF home page, click any new feature or regression pipeline where you want to use this feature.
2. In the left navigation pane, click **Build with Parameters**.
3. On the FEATURE AND TESTCASES section, click **All** from the Features drop-down menu to view the TestCases details mapped to each feature.

Figure 2-81 Test Case Mapping

The screenshot displays the Oracle Communications Cloud Native Core Automated Test Suite - POLICY interface. The left navigation pane includes options like Status, Changes, Build with Parameters, Configure, Full Stage View, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area is titled "Pipeline Policy-NewFeatures" and shows "EXECUTION OPTIONS" with fields for TestSuite (NewFeatures), SUT (PCF), Configuration_Type (Product_Config), and FilterWithTags (No). Below this is the "FEATURES AND TESTCASES" section. A red box highlights the "Features" drop-down menu, which is currently set to "All". Another red box highlights the "TestCases" section, which shows a list of test cases grouped by stage and group. The test cases include names like "Bulwark_Support_SM_Creates_Delete_UpdateIdentify_PDSNotification_RaaSCap_verify" and "Internal_Binding_Header_AMPolicy_ServiceName_Instance". At the bottom, there are "Build" and "Cancel" buttons.

4. Click **Select** from the **Features** drop-down menu to view the the test cases details.

Figure 2-82 Test Cases Details When Select a Single/MultipleFeatures

FEATURES AND TESTCASES

Features Select ▾

Stage: stage1 ☐

Group: group1 ☐

☒ Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_verify ☐ internal_Binding_Header_AMPolicy_Svcname_nf_instance
☐ internal_SM_Multi_ServerHeader_with_Retry_using_SCP ☐ internal_SM_Rx_AAR-L_without_MCD

Stage: stage2 ☐

Group: group3 ☐

☐ Pending_Transaction_N7_Fast_Lock_Bulwark

Stage: stage3 ☐

Group: group3 ☐

☐ internal_max_rx_session

TestCases All ▾

Total Test Cases are : 1

Stage: stage1 Group: group1

Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_verify

• Bulwark_CHF_UDR_notification_RedCap_verify

▶ Build Cancel

3

Installing ATS for Different Network Functions

This section describes how to install ATS for different network functions. It includes:

- [Installing ATS for BSF](#)
- [Installing ATS for NEF](#)
- [Installing ATS for NRF](#)
- [Installing ATS for NSSF](#)
- [Installing ATS for Policy](#)
- [Installing ATS for SCP](#)
- [Installing ATS for SEPP](#)
- [Installing ATS for UDR](#)

3.1 Installing ATS for BSF

The BSF ATS installation procedure covers two steps:

1. Locating and downloading the ATS package for BSF.
2. Deploying ATS and stub pods in Kubernetes cluster

This includes installation of three stubs (nf1stub, nf11stub, and nf12stub), ocdns-bind stub, and BSF ATS in BSF namespace.

3.1.1 Resource Requirements

This section describes the ATS resource requirements for Binding Support Function.

Overview - Total Number of Resources

The following table describes the overall resource usage in terms of CPUs, memory, and storage:

Table 3-1 BSF - Total Number of Resources

Resource Name	Non-ASM CPU	Non-ASM Memory (GB)	ASM CPU	ASM Memory (GB)
BSF Total	41	36	73	52
ATS Total	11	11	23	17
cnDBTier Total	107.1	175.2	137.1	190.2
Grand Total BSF ATS	159.1	222.2	233.1	259.2

BSF Pods Resource Requirements Details

This section describes the resource requirements, which are needed to deploy BSF ATS successfully.

Table 3-2 BSF Pods Resource Requirements Details

BSF Microservices	Max CPU	Memory (GB)	Max Replica	Isito ASM CPU	Isito ASM Memory (GB)	Non-ASM Total CPU	Non-ASM Memory (GB)	ASM Total CPU	ASM Total Memory (GB)
oc-app-info	1	1	1	2	1	1	1	3	2
oc-diam-gateway	4	2	1	2	1	4	2	6	3
alternate-route	2	4	1	2	1	2	4	4	5
oc-config-server	4	2	1	2	1	4	2	6	3
ocgresss_gateway	4	6	1	2	1	4	6	6	7
ocingress_gateway	4	6	1	2	1	4	6	6	7
nrf-client-mngt	1	1	2	2	1	2	2	6	4
oc-audit	2	1	1	2	1	2	1	4	2
oc-config-mgmt	4	2	2	2	1	8	4	12	6
oc-query	2	1	2	2	1	4	2	8	4
oc-perf-info	1	1	2	2	1	2	2	6	4
bsf-management-service	4	4	1	2	1	4	4	6	5
BSF Totals						41	36	73	52

ATS Resource Requirements details for BSF

This section describes the ATS resource requirements, which are needed to deploy BSF ATS successfully.

Table 3-3 ATS Resource Requirements Details

ATS Microservices	Max CPU	Max Memory (GB)	Max Replica	Isito ASM CPU	Isito ASM Memory (GB)	Non-ASM Total CPU	Non-ASM Total Memory (GB)	ASM Total CPU	ASM Total Memory (GB)
ocstub1-py	2	2	1	2	1	2	2	4	3
ocstub2-py	2	2	1	2	1	2	2	4	3

Table 3-3 (Cont.) ATS Resource Requirements Details

ATS Microservices	Max CPU	Max Memory (GB)	Max Replica	Isito ASM CPU	Isito ASM Memory (GB)	Non-ASM Total CPU	Non-ASM Total Memory (GB)	ASM Total CPU	ASM Total Memory (GB)
ocstub3-py	2	2	1	2	1	2	2	4	3
ocats-bsf	3	3	1	2	1	3	3	5	4
ocdns-bind	1	1	1	2	1	1	1	3	2
ocdiam-sim	1	1	1	2	1	1	1	3	2
ATS Totals						11	11	23	17

cnDBTier Resource Requirements Details for BSF ATS

This section describes the cnDBTier resource requirements, which are needed to deploy BSF ATS successfully.

Note

For cnDBTier pods, a minimum of 4 worker nodes are required.

Table 3-4 cnDBTier Resource Requirements Details

cnDBTier Microservices	Min CPU	Min Memory (GB)	Min Replica	Isito ASM CPU	Isito ASM Memory (GB)	Total CPU	Total Memory (GB)	ASM Total CPU	ASM Total Memory (GB)
db_monitor_svc	1	1	1	2	1	1	1	3	2
db_replication_svc	2	12	1	2	1	2	12	4	13
db_backup_manager_svc	0.1	0.2	1	2	1	0.1	0.2	2.1	1.2
ndbappmysqld	8	10	4	2	1	32	40	40	44
ndbmngmd	4	10	2	2	1	8	20	12	22
ndbmtl	10	18	4	2	1	40	72	48	76
ndbmysqld	8	10	2	2	1	16	20	20	22
db_infra_monitor_svc	8	10	1	2	1	8	10	8	10
cnDBTier Total						107.1	175.2	137.1	190.2

3.1.2 Downloading the ATS Package

This section provides information on how to locate and download BSF ATS package file from My Oracle Support (MOS).

Locating and Downloading BSF ATS Package

To locate and download the ATS Image from MOS, perform the following steps:

1. Log in to [My Oracle Support](#) using the appropriate credentials.
2. Select the **Patches and Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Binding Support Function <release_number>* from **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required patch from the search results. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the **<p*****_<release_number>_Tekelec>.zip** file to download the BSF ATS package file.
10. Untar the gzip file `ocats-bsf-tools-24.1.0.0.0.tgz` to access the following files:

```
ocats-bsf-pkg-24.1.0.0.0.tgz
ocdns-pkg-24.1.0.tgz
ocstub-pkg-24.1.0.tgz
ocdiam-sim-24.1.0.tgz
```

The contents included in each of these files are as follow:

```
ocats-bsf-tools-24.1.0.0.0.tgz
|
|_ _ _ _ _ ocats-bsf-pkg-24.1.0.tgz
| | _ _ _ _ _ ocats-bsf-24.1.0.tgz (Helm Charts)
| | _ _ _ _ _ ocats-bsf-images-24.1.0.tar (Docker Images)
| | _ _ _ _ _ ocats-bsf-data-24.1.0.tgz (BSF ATS and Jenkins job Data)
|
|_ _ _ _ _ ocstub-pkg-24.1.0.0.0.tgz
| | _ _ _ _ _ ocstub-py-24.1.0.tgz(Helm Charts)
| | _ _ _ _ _ ocstub-py-image-24.1.0.tar (Docker Images)
|
|_ _ _ _ _ ocdns-pkg-24.1.0.0.0.tgz
| | _ _ _ _ _ ocdns-bind-24.1.0.tgz(Helm Charts)
| | _ _ _ _ _ ocdns-bind-image-24.1.0.tar (Docker Images)
|
|_ _ _ _ _ ocdiam-pkg-24.1.0.0.0.tgz
| | _ _ _ _ _ ocdiam-sim-24.1.0.tgz(Helm Charts)
| | _ _ _ _ _ ocdiam-sim-image-24.1.0.tar (Docker Images)
```

11. Copy the tar file from the downloaded package to OCCNE, OCI, or Kubernetes cluster where you want to deploy ATS.

3.1.3 Pushing the Images to Customer Docker Registry

This section describes the pre-deployment steps for deploying ATS and stub pods.

Preparing to deploy ATS and Stub Pods in Kubernetes Cluster

To deploy ATS and Stub pods in a Kubernetes Cluster, perform the following steps:

1. Run the following command to extract the tar file content:
`tar -zxvf ocats-bsf-tools-24.1.0.0.tgz`

The output of this command is:

```
ocats-bsf-pkg-24.1.0.tgz
ocstub-pkg-24.1.0.tgz
ocdns-pkg-24.1.0.tgz
ocdiam-pkg-24.1.0.0.tgz
```

2. Go to the `ocats-bsf-tools-24.1.0.0` folder and run the following command to extract the helm charts and docker images of ATS:
`tar -zxvf ocats-bsf-pkg-24.1.0.0.tgz`

The output of this command is:

```
ocats-bsf-24.1.0.tgz
ocats-bsf-images-24.1.0.tar
ocats-bsf-data-24.1.0.tgz
```

3. Run the following command in your cluster to load the ATS docker image:
`docker load --input ocats-bsf-images-24.1.0.tar`
4. Run the following commands to tag and push the ATS images

```
docker tag ocats-bsf:24.1.0 <registry>/ocats-bsf:24.1.0
docker push <registry>/ocats-bsf:24.1.0
```

Example:

```
docker tag ocats-bsf:24.1.0 localhost:5000/ocats-bsf:24.1.0
docker push localhost:5000/ocats-bsf:24.1.0
```

5. Run the following command to untar the helm charts, `ocats-bsf-24.1.0.tgz`
`tar -zxvf ocats-bsf-24.1.0.tgz`
6. Update the registry name, image name and tag in the `ocats-bsf/values.yaml` file as required. For this, you need to update the `image.repository` and `image.tag` parameters in the `ocats-bsf/values.yaml` file.
7. In the `ocats-bsf/values.yaml` file, the `atsFeatures` parameter is configured to control ATS feature deliveries.

```
atsFeatures: ## DO NOT UPDATE this section without My Oracle Support
              team's support
```

```

testCaseMapping: true           # To display Test cases on GUI along
with Features
logging: true                   # To enable feature to collect
applogs in case of failure
lightWeightPerformance: false   # The Feature is not implemented yet
executionWithTagging: true      # To enable Feature/Scenario
execution with Tag
scenarioSelection: false        # The Feature is not implemented yet
parallelTestCaseExecution: true # To run ATS features parallel
parallelFrameworkChangesIntegrated: true # To run ATS features parallel
mergedExecution: false          # To execute ATS Regression and
NewFeatures pipelines together in merged manner
individualStageGroupSelection: false # The Feature is not implemented
yet
parameterization: true          # When set to false, the
Configuration_Type parameter on the GUI will not be available.
atsApi: true                    # To trigger ATS using ATS API
healthcheck: true               # TO enable/disable ATS Health Check.
atsGuiTLSEnabled: false         # To run ATS GUI in https mode.
atsCommunicationTLSEnabled: false #If set to true, ATS will get
necessary variables to communicate with SUT, Stub or other NFs with TLS
enabled. It is not required in ASM environment.

```

Note

It is recommended to avoid altering atsFeatures flags.

3.1.4 Configuring ATS

3.1.4.1 Enabling Static Port

To enable static port, in the `ocats-bsf/values.yaml` file under the service section, set the value of `staticNodePortEnabled` parameter to `true` and enter a valid `nodePort` value for `staticNodePort` parameter.

```

service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
  ports:
    http:
      port: "8080"
      staticNodePortEnabled: false
      staticNodePort: ""

```

3.1.4.2 Enable Static API Node Port

To enable static API node port, in the `ocats-bsf/values.yaml` file under the **service** section, set the value of **staticNodePortEnabled** parameter to true and enter a valid nodePort value for **staticNodePort** parameter. The following is a snippet of the service section in the yaml file:

```
service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
  ports:
    api:
      port: "5001"
      staticNodePortEnabled: false
      staticNodePort: ""
```

3.1.4.3 Service Account Requirements

To run BSF-ATS, use the following rules to create a service account:

```
rules:
- apiGroups: ["extensions"]
  resources: ["deployments", "replicasets"]
  verbs: ["watch", "get", "list", "update"]
- apiGroups: ["apps"]
  resources: ["deployments", "replicasets"]
  verbs: ["watch", "get", "list", "update"]
- apiGroups: [""]
  resources: ["pods", "services", "secrets", "configmaps"]
  verbs: ["watch", "get", "list", "delete", "update", "create"]
- apiGroups: [""]
  resources: ["pods/log"]
  verbs: ["get", "list"]
```

3.1.4.4 Enabling Aspen Service Mesh

This section provides information on how to enable Aspen service mesh while deploying ATS for Binding Support Function. The configurations mentioned in this section are optional and should be performed only if ASM is required.

To enable service mesh for BSF ATS, perform the following steps:

1. In the **service** section of the `values.yaml` file, the **serviceMeshCheck** parameter is set to false by default. To enable service mesh, set the value for **serviceMeshCheck** to true. The following is a snippet of the service section in the yaml file:

```
service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
```

```

ports:
  https:
    port: "8443"
    staticNodePortEnabled: false
    staticNodePort: ""
  http:
    port: "8080"
    staticNodePortEnabled: false
    staticNodePort: ""
  api:
    port: "5001"
    staticNodePortEnabled: false
    staticNodePort: ""
serviceMeshCheck: true

```

2. If the ASM is not enabled on the global level for the namespace, run the following command to enable it before deploying the ATS:

```
kubectl label --overwrite namespace <namespace_name> istio-
injection=enabled
```

For example:

```
kubectl label --overwrite namespace ocbsf istio-injection=enabled
```

3. Uncomment and add the following annotation under the **lbDeployments** and **nonlbDeployments** section of the global section in `values.yaml` file as follows:

```

traffic.sidecar.istio.io/excludeInboundPorts: "9000"
traffic.sidecar.istio.io/excludeOutboundPorts: "9000"

```

The following is a snippet from the `values.yaml` of BSF:

```

/home/cloud-user/ocats-bsf/ocats-bsf-tools-24.1.0.0.0/ocats-bsf-
pkg-24.1.0.0.0/ocats-bsf/
vim values.yaml

```

```

customExtension:
  allResources:
    labels: {}
    annotations: {
      #Enable this section for service-mesh based installation
      traffic.sidecar.istio.io/excludeInboundPorts: "9000",
      traffic.sidecar.istio.io/excludeOutboundPorts: "9000"
    }

```

```

lbDeployments:
  labels: {}
  annotations: {
    traffic.sidecar.istio.io/excludeInboundPorts: "9000",
    traffic.sidecar.istio.io/excludeOutboundPorts: "9000"}

```

4. If service mesh is enabled, then create a destination rule for fetching the metrics from the Prometheus. In most of the deployments, Prometheus is kept outside the service mesh so you need a destination rule to communicate between TLS enabled entity (ATS) and non-

TLS entity (Prometheus). You can create a destination rule using the following sample yaml file:

```
kubectl apply -f - <<EOF

apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: prometheus-dr
  namespace: ocats
spec:
  host: oso-prometheus-server.pcf.svc.cluster.local
  trafficPolicy:
    tls:
      mode: DISABLE
EOF
```

In the destination rule:

- **name** indicates the name of destination rule.
 - **namespace** indicates where the ATS is deployed.
 - **host** indicates the hostname of the prometheus server.
5. Update the `ocbsf_custom_values_servicemesh_config_24.1.0.yaml` with the below additional configuration under `virtualService` section for Egress Gateway:

```
virtualService:
- name: nrfvirtual1
  host: ocbsf-ocbsf-egress-gateway
  destinationhost: ocbsf-ocbsf-egress-gateway
  port: 8000
  exportTo: |-
    [ "." ]
  attempts: "0"
```

Where,

host or destination name uses the format - `<release_name>-<egress_svc_name>`.

You must update the host or destination name as per the deployment.

6. Perform helm upgrade on the `ocbsf-servicemesh-config` release using the modified `ocbsf_custom_values_servicemesh_config_24.1.0.yaml` file.

```
helm upgrade <helm_release_name_for_servicemesh> -n <namespace>
<servicemesh_charts> -f <servicemesh-custom.yaml>
```

For example,

```
helm upgrade ocbsf-servicemesh-config ocbsf-servicemesh-config-24.1.0.tgz -
n ocbsf -f ocbsf_custom_values_servicemesh_24.1.0.yaml
```

7. Configure DNS for Alternate Route service. For more information, see [Post-Installation Steps](#).

3.1.4.5 Enabling Health Check

This section describes how to enable Health Check for ATS.

To enable Health Check, in the `ocats-bsf/values.yaml` file, set the value of `healthcheck` parameter to `true` and enter a valid value to select either **Webscale** or **OCCNE** environment.

To select OCCNE environment, set the `envtype` to **OCCNE** and update the values of the following parameters:

- `webscale` - Update the value as `false`
- `envtype` - `T0NDTKU=` (i.e `envtype=$(echo -n 'OCCNE' | base64)`)
- `occnehostip` - OCCNE Host IP address
- `occnehostusername` - OCCNE Host Username
- `occnehostpassword` - OCCNE Host Password

To select OCCNE environment, update the values of the following two parameters:

- `webscale` - Update the value as `true`
- `envtype` - `T0NDTKU=` (i.e `envtype=$(echo -n 'OCCNE' | base64)`)

After the configurations are done, encrypt the parameters and provide the values as shown in the following snippet:

```
atsFeatures: ## DO NOT UPDATE this section without Engineering team's
permission
  healthcheck: true                # TO enable/disable ATS Health Check.

webscale: false
healthchecksecretname: "healthchecksecret"
occnehostip: "MTAuMTcuMjE5LjY1"
occnehostusername: "dXNlcm5hbWU="
occnehostpassword: "KioqKg=="
envtype: "T0NDTKU="
```

To select WEBSCALE environment, update the values of the following two parameters:

- `webscale` - Update the value as `true`
- `envtype` - `V0VCU0NBTEU=` (i.e `envtype=$(echo -n 'WEBSCALE' | base64)`)

After the configurations are done, encrypt the parameters and provide the values as shown in the following snippet:

```
atsFeatures: ## DO NOT UPDATE this section without Engineering team's
permission
  healthcheck: true                # TO enable/disable ATS Health Check.

webscale: true
healthchecksecretname: "healthchecksecret"
occnehostip: ""
occnehostusername: ""
```



```

occnehostpassword: ""
envtype: "V0VCU0NBTEU="
webscalejumpip: "MTAuNzAuMTE3LjQy"
webscalejumpusername: "dXNlcm5hbWU="
webscalejumppassword: "KioqKg=="
webscaleprojectname: "KioqKg=="
webscalelabserverFQDN: "KioqKg=="
webscalelabserverport: "KioqKg=="
webscalelabserverusername: "KioqKg=="
webscalelabserverpassword: "KioqKg=="

```

① Note

Once the ATS is deployed with HealthCheck feature enabled or disabled, then it cannot be changed. To change the configuration, you are required to re-install.

3.1.4.6 Enabling Persistent Volume

① Note

The steps provided in this section are optional and required only if Persistent Volume needs be to enabled.

ATS supports Persistent storage to retain ATS historical build execution data, test cases and one-time environment variable configurations. With this enhancement, the user can decide whether to use persistent volume based on their resource requirements. By default, the persistent volume feature is not enabled.

To enable persistent storage, perform the following steps:

1. Create a PVC using PersistentVolumeClaim.yaml file and associate the same to the ATS pod.

Sample PersistentVolumeClaim.yaml file:

```

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: <Enter the PVC Name>
  annotations:
spec:
  storageClassName: <Provide the Storage Class Name>
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: <Provide the size of the PV>

```

- a. Set **PersistentVolumeClaim** to the PVC file name.
- b. Enter the **storageClassName** to the Storage Class Name.
- c. Set **storage** to and size of the persistent volume.

Sample PVC configuration:

```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: bsf-pvc-24.1.0
annotations:
spec:
  storageClassName: standard
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 1Gi
```

2. Run the following command to create PVC:

```
kubectl apply -f <filename> -n <namespace>
```

For example:

```
kubectl apply -f PersistentVolumeClaim.yaml -n ocbsf
```

Output:

```
persistentvolumeclaim/bsf-pvc-24.1.0 created
```

3. Once the PVC is created , run the following command to verify that it is bound to the persistent volume and is available.

```
kubectl get pvc -n <namespace used for pvc creation>
```

For example:

```
kubectl get pvc -n ocbsf
```

Sample output:

NAME	STATUS	VOLUME
CAPACITY	ACCESS MODES	STORAGECLASS AGE
bsf-pvc-24.1.0	Bound	pvc-65484045-3805-4064-9fc3-f9eeeaccc8b8
1Gi	RWO	standard 11s

Verify that the STATUS is Bound and rest of the parameters like NAME, CAPACITY, ACCESS MODES, and STORAGECLASS are as mentioned in the PersistentVolumeClaim.yaml file.

Note

Do not proceed further with the next step if there is an issue with the PV creation and contact your administrator to get the PV Created.

4. Enable PVC:
 - a. Set the **PVEnabled** flag to **true**.
 - b. Set **PVClaimName** to the PVC created in Step 1.

```
deployment:
  customExtension:
    labels: {}
    annotations: {}
    PVEnabled: true
    PVClaimName: "ocbsf-pvc-24.1.0"
```

Note

Make sure that ATS is deployed before proceeding to the further steps.

5. Copy the `<nf_main_folder>` and `<jenkins jobs>` folders from the tar file to their ATS pod and restart the pod.
 - a. Extract the tar file.

```
tar -xvf ocats-bsf-data-24.1.0.tgz
```

- b. Run the following commands to copy the desired folder.

```
kubectl cp ocats-bsf-data-24.1.0/ocbsf_tests <namespace>/<pod-name>:/var/lib/jenkins/
```

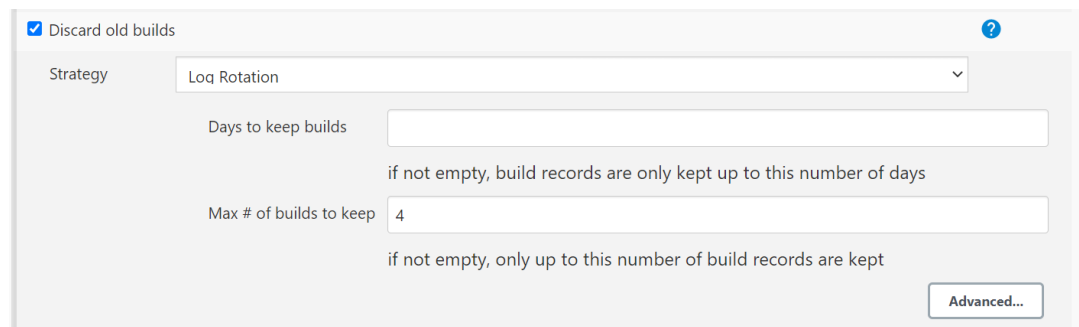
```
kubectl cp ocats-bsf-data-24.1.0/jobs <namespace>/<pod-name>:/var/lib/jenkins/.jenkins/
```

- c. Restart the pod.

```
kubectl delete po <pod-name> -n <namespace>
```

6. Once the Pod is up and running, log in to the Jenkins console and configure the **Discard old Builds** option to configure the number of Jenkins builds, which must be retained in the persistent volume.

Figure 3-1 Discarding Old Builds



☒ Discard old builds ?

Strategy: Log Rotation

Days to keep builds:

if not empty, build records are only kept up to this number of days

Max # of builds to keep: 4

if not empty, only up to this number of build records are kept

[Advanced...](#)

Note

If **Discard old Builds** is not configured, Persistent Volume can get filled when there are huge number of builds.

For more details on Persistent Volume Storage, see [Persistent Volume for 5G ATS](#).

3.1.5 Deploying ATS and Pods

3.1.5.1 Deploying ATS in Kubernetes Cluster

Important

This Procedure is for Backwards porting purpose only and should not be considered as the Subsequent Release POD Deployment Procedure.

Prerequisite: Make sure that the old PVC, which contains the old release POD data is available.

To deploy ATS, perform the following steps:

1. Run the following command to deploy ATS using the updated helm charts:

Note

Ensure that all the the components, that is, ATS, stub pods and CNC BSF are deployed in the same namespace.

Using Helm 3

```
helm install -name <release_name> ocats-bsf-24.1.0.tgz --namespace
<namespace_name> -f <values-yaml-file>
```

For example:

```
helm install -name ocats ocats-bsf-24.1.0.tgz --namespace ocbsf -f ocats-
bsf/values.yaml
```

2. Run the following command to verify ATS deployment:

```
helm ls -n ocbsf
```

The output of the command is as follows:

NAME	REVISION	UPDATED		
STATUS	CHART	APP VERSION	NAMESPACE	

ocats	1	Mon Nov 14 14:56:11 2020
DEPLOYED	ocats-bsf-24.1.0	1.0 ocbsf

If the deployment is successful, the status is **Deployed**.

3.1.5.2 Deploying Stub Pod in Kubernetes Cluster

To deploy Stub Pod in Kubernetes cluster, perform the following steps:

1. Navigate to ocats-bsf-tools-24.1.0.0.0 folder and run the following command:

```
tar -zxvf ocstub-pkg-24.1.0.0.0.tgz
```

The output of the command shows:

- ocstub-py-24.1.0.tgz
 - ocstub-py-image-24.1.0.tar
2. Deploy the additional stubs required to validate the session retry feature. You can use nf11stub or nf12stub as alternate FQDN for nf1stub.

- a. Run the following command to load the stub image.

```
docker load --input ocstub-py-image-24.1.0.tar
```

- b. Tag and push the image to your docker registry using below commands.

```
docker tag ocats-bsf:24.1.0 localhost:5000/ocats-bsf:23.4.0
docker push localhost:5000/ocats-bsf:24.1.0
```

- c. Untar the helm charts ocstub-py-24.1.0.tgz and update the registry name, image name and tag (if required) in ocstub-py/values.yaml file.
- d. If required, change apiVersion to apps/v1 in ocstub-py/templates/deployment.yaml file.

```
apiVersion: apps/v1
```

- e. Deploy the stub:

```
helm install -name <release_name> ocstub-py --set env.NF=<NF> --setenv.LOG_LEVEL=<DEBUG/INFO> --set service.name=<service_name> --set service.appendReleaseName=false --namespace=<namespace_name> -f <valuesyaml-file>
```

For example:

```
helm install -name nf1stub ocstub-py --set env.NF=BSF --set env.LOG_LEVEL=DEBUG --set service.name=nf1stub --set service.appendReleaseName=false --namespace=ocbsf -f ocstub-py/values.yaml
```

```
helm install -name nf11stub ocstub-py --set env.NF=BSF --set
```

```
env.LOG_LEVEL=DEBUG --set service.name=nf11stub --set
service.appendReleaseName=false --namespace=ocbsf -f ocstub-py/
values.yaml
```

```
helm install-name nf12stub ocstub-py --set env.NF=BSF --set
env.LOG_LEVEL=DEBUG --set service.name=nf12stub --set
service.appendReleaseName=false --namespace=ocbsf -f ocstub-py/
values.yaml
```

- f. Run the following command to verify the stub deployment:

```
helm ls -n ocbsf
```

Sample output:

```
[cloud-user@platform-bastion-1 ocstub-pkg-24.1.0.0.0]$ helm ls -n ocbsf
```

NAME	REVISION	UPDATED
STATUS	CHART	APP VERSION NAMESPACE
nf11stub	1	Thu Jul 29 05:55:48 2021
DEPLOYED	ocstub-py-24.1.0	1.0 ocbsf
nf12stub	1	Thu Jul 29 05:55:50 2021
DEPLOYED	ocstub-py-24.1.0	1.0 ocbsf
nf1stub	1	Thu Jul 29 05:55:47 2021
DEPLOYED	ocstub-py-24.1.0	1.0 ocbsf

- g. Run the following command to verify the ATS and Stubs deployment status:

```
helm status -n ocbsf
```

- h. Run the following command to verify if all the services are installed.

```
kubectl get po -n ocbsf
```

Sample output:

```
[cloud-user@platform-bastion-1 ocstub-pkg-24.1.0.0.0]$ kubectl get po -
n ocbsf
```

NAME	READY	STATUS
RESTARTS AGE		
nf11stub-ocstub-py-7bffd6dcd7-ftm5f	1/1	Running
0 3d23h		
nf12stub-ocstub-py-547f7cb99f-7mpll	1/1	Running
0 3d23h		
nf1stub-ocstub-py-bdd97cb9-xjrkx	1/1	Running
0 3d23h		

3.1.5.3 Deploying DNS Stub in Kubernetes Cluster

Note

Ensure there are sufficient resources and limit for DNS Stub. Set the resource request and limit values in the **resources** section of the **values.yaml** file as follows:

```
resources: {}
# We usually recommend not to specify default resources and to leave
this as a conscious
# choice for the user. This also increases chances charts run on
environments with little
# resources, such as Minikube. If you do want to specify resources,
uncomment the following
# lines, adjust them as necessary, and remove the curly braces after
'resources:'. # limits:
#   cpu: 1000m
#   memory: 1024Mi
# requests:
#   cpu: 500m
#   memory: 500Mi
```

To deploy DNS Stub in Kubernetes cluster, perform the following steps:

1. Go to the `ocats-bsf-tools-24.1.0.0.0` folder and run the following command to extract the `ocstub` tar file content:

```
tar -zxvf ocdns-pkg-24.1.0.0.0.tgz
```

Sample output:

```
[cloud-user@platform-bastion-1 ocdns-pkg-24.1.0.0.0]$ ls -ltrh
total 211M
-rw-----. 1 cloud-user cloud-user 211M Mar 14 14:49 ocdns-bind-
image-24.1.0.tar
-rw-r--r--. 1 cloud-user cloud-user 2.9K Mar 14 14:49 ocdns-bind-24.1.0.tgz
```

2. Run the following command in your cluster to load the DNS STUB image:
`docker load --input ocdns-bind-image-24.1.0.tar`

3. Run the following commands to tag and push the DNS STUB image:

```
docker tag ocdns-bind:24.1.0 localhost:5000/ocdns-bind:24.1.0
docker push localhost:5000/ocdns-bind:24.1.0
```

4. Run the following command to untar the helm charts, `ocdns-bind-24.1.0.tgz`.
`tar -zxvf ocdns-bind-24.1.0.tgz`
5. Update the registry name, image name and tag (if required) in the `ocdns-bind/values.yaml` file as required. For this, open the **values.yaml** file and update the `image.repository` and `image.tag` parameters.

6. Run the following command to deploy the DNS Stub.

Using Helm3:

```
helm install -name ocdns ocdns-bind-24.1.0.tgz --namespace ocbsf -f ocdns-bind/values.yaml
```

7. Capture the cluster name of the deployment, namespace where nfstubs are deployed, and the cluster IP of DNS Stub.

To capture the DNS Stub cluster IP:

```
kubectl get svc -n ocbsf | grep dns
```

Sample output:

```
[cloud-user@platform-bastion-1 ocdns-pkg-24.1.0.0.0]$ kubectl get svc -n ocbsf | grep dns
```

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP
ocdns	ClusterIP	10.233.11.45	<none>
UDP, 6236/TCP		19h	53/

To capture the cluster name:

```
kubectl -n kube-system get configmap kubeadm-config -o yaml | grep clusterName
```

Sample output:

```
clusterName: platform
```

3.1.5.4 Deploying ocdiam Simulator in Kubernetes Cluster

To deploy ocdiam Simulator in Kubernetes cluster, perform the following steps:

1. Go to the ocats-bsf-tools-24.1.0.0.0 folder and run the following command to extract the ocstub tar file content:

```
tar -zxvf ocdiam-pkg-24.1.0.0.0.tgz
```

Sample output:

```
[cloud-user@platform-bastion-1 ocdiam-pkg-24.1.0.0.0]$ ls -ltrh
```

total	908M
-rw-----	1 cloud-user cloud-user 908M Mar 14 14:49 ocdiam-sim-image-24.1.0.tar
-rw-r--r--	1 cloud-user cloud-user 3.8K Mar 14 14:49 ocdiam-sim-24.1.0.tgz

2. Run the following command in your cluster to load the DNS STUB image:

```
docker load --input ocdiam-sim-image-24.1.0.tar
```


- Run the following commands to tag and push the DNS STUB image:

```
docker tag ocddiam-sim:24.1.0 localhost:5000/ocddiam-sim:24.1.0
docker push localhost:5000/ocddiam-sim:24.1.0
```

- Run the following command to untar the helm charts, ocddiam-sim-24.1.0.tgz.

```
tar -zxvf ocddiam-sim-24.1.0.tgz
```

- Update the registry name, image name and tag (if required) in the ocddiam-sim/values.yaml file as required. For this, open the **values.yaml** file and update the image.repository and image.tag parameters.

- Run the following command to deploy the Diam Sim.

Using Helm3:

```
helm install -name ocddiam-sim ocddiam-sim --namespace ocbscf -f ocddiam-sim/
values.yaml
```

Output:

```
ocddiam-sim-69968444b6-fg6ks          1/1      Running    0
5h47m
```

Sample of BSF namespace with BSF and ATS after installation:

```
[cloud-user@platform-bastion-1 ocstub-pkg-24.1.0.0.0]$ kubectl get po -n ocbsf
NAME                                READY   STATUS
RESTARTS   AGE
ocbsf-appinfo-6fc99ffb85-f96j2      1/1     Running
1          3d23h
ocbsf-bsf-management-service-df6b68d75-m77dv  1/1     Running
0          3d23h
ocbsf-oc-config-79b5444f49-7pwzx     1/1     Running
0          3d23h
ocbsf-oc-diam-connector-77f7b855f4-z2p88  1/1     Running
0          3d23h
ocbsf-oc-diam-gateway-0              1/1     Running
0          3d23h
ocbsf-ocats-bsf-5d8689bc77-cxdvx     1/1     Running
0          3d23h
ocbsf-ocbsf-egress-gateway-644555b965-pkxsb  1/1     Running
0          3d23h
ocbsf-ocbsf-ingress-gateway-7558b7d5d4-lfs5s  1/1     Running
4          3d23h
ocbsf-ocbsf-nrf-client-nfmanagement-d6b955b48-4pptk  1/1     Running
0          3d23h
ocbsf-ocdns-ocdns-bind-75c964648-j5f5d     1/1     Running
0          3d23h
ocbsf-ocpm-cm-service-7775c76c45-xgztj     1/1     Running
0          3d23h
ocbsf-ocpm-queryservice-646cb48c8c-d72x4    1/1     Running
0          3d23h
ocbsf-performance-69fc459ff6-frrvs         1/1     Running
4          3d23h
```

ocbsfnf11stub-7bffd6dcd7-ftm5f	1/1	Running
0 3d23h		
ocbsfnf12stub-547f7cb99f-7mpll	1/1	Running
0 3d23h		
ocbsfnf1stub-bdd97cb9-xjrkx	1/1	Running
0 3d23h		
ocdiam-sim-69968444b6	1/1	Running
0 3d23h		

3.1.6 Post-Installation Steps

The section describes post-installation steps that users should perform after deploying ATS and stub pods.

Alternate Route Service Configurations

To edit the Alternate Route Service deployment file (ocpcf-ocbsf-alternate-route) that points to DNS Stub, perform the following steps:

1. Run the following command to get searches information from dns-bind pod to enable communication between Alternate Route and dns-bind service:

```
kubectl exec -it <dns-bind pod> -n <NAMESPACE> -- /bin/bash -c 'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
```

The following output is displayed after running the command:

Figure 3-2 Sample Output

```
[flanders@bastion-1 ~]$ kubectl exec -it ocdns-stub-ocdns-bind-7467cf56c8-5dgg2 -n flanders-ns -- /bin/bash -c 'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
flanders-ns.svc.inferno.lab.us.oracle.com
svc.inferno.lab.us.oracle.com
inferno.lab.us.oracle.com
```

By default alternate service will point to CoreDNS and you will see following settings in deployment file:

Figure 3-3 Alternate Route Service Deployment File

```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

2. Run the following command to edit the deployment file and add the following content in alternate service to query DNS stub:

```
$kubectl edit deployment ocpcf-occpn-alternate-route -n ocpcf
```

- a. Add the IP Address of the nameserver that you have recorded after installing the DNS stub (cluster IP Address of DNS Stub).
- b. Add the search information one by one which you recorded earlier.

- c. Set dnsPolicy to "None".

```
dnsConfig:
  nameservers:
    - 10.233.33.169      // cluster IP of DNS Stub
  searches:
    - ocpcf.svc.occne15-ocpcf-ats
    - svc.occne15-ocpcf-ats
    - occne15-ocpcf-ats
  dnsPolicy: None
```

For example:

Figure 3-4 Example

```
terminationMessagePath: /dev/termination-log
terminationMessagePolicy: File
dnsConfig:
  nameservers:
    - 10.233.31.245
  searches:
    - flanders-ns.svc.inferno.lab.us.oracle.com
    - svc.inferno.lab.us.oracle.com
    - inferno.lab.us.oracle.com
  dnsPolicy: None
restartPolicy: Always
schedulerName: default-scheduler
```

NRF client configmap

1. In the **application-config** configmap, configure the following parameters with the respective values:
 - primaryNrfApiRoot=nf1stub.<namespace_gostubs_are_deployed_in>.svc:8080
Example: primaryNrfApiRoot=nf1stub.ocats.svc:8080
 - secondaryNrfApiRoot=nf11stub.<namespace_gostubs_are_deployed_in>.svc:8080
Example: secondaryNrfApiRoot=nf11stub.ocats.svc:8080
 - virtualNrfFqdn = nf1stub.<namespace_gostubs_are_deployed_in>.svc
Example: virtualNrfFqdn=nf1stub.ocats.svc

Note

To get all configmaps in your namespace, run the following command:
 kubectl get configmaps -n <BSF_namespace>

2. (Optional) If persistent volume is used, follow the post-installation steps provided in the [Persistent Volume for 5G ATS](#) section.

3.2 Installing ATS for NEF

Following are the ATS installation procedures for NEF:

1. [Downloading the ATS Package](#)
2. [Pushing the Images to Customer Docker Registry](#)
3. [Configuring ATS](#)
 - a. [Enabling Static Port](#)
4. [Deploying ATS, Stub and CNC Console in Kubernetes Cluster](#)
5. [Verifying ATS Deployment](#)

3.2.1 Resource Requirements

This section describes the ATS resource requirements for NEF.

Overview - Total Number of Resources

The following table describes the overall resource usage in terms of CPUs, memory, and storage for the following:

- NEF SUT
- cnDBTier
- ATS

Table 3-5 NEF - Total Number of Resources

Resource Name	CPU	Memory (GB)	Storage (GB)
NEF SUT Totals	21.6	21.6	4
cnDBTier Totals	40	40	20
ATS Totals	4	3	0
Grand Total NEF ATS	65.6	64.6	24

NEF Pods Resource Requirements Details

This section describes the resource requirements, which are needed to deploy NEF ATS successfully.

Table 3-6 NEF Pods Resource Requirements Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	# Replicas (regular deployment)	# Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
oc_nef_ccfclient_manager	0.7	0.7	0	2	1	0.7	0.7	0
oc_nef_monitoring_events	1	1	0	2	1	1	1	0
oc_nef_quality_of_service	1	1	0	2	1	1	1	0
oc_nef_aef_apirouter	1	1	0	2	1	1	1	0
oc_nef_apd_manager	0.7	0.7	0	1	1	0.7	0.7	0

Table 3-6 (Cont.) NEF Pods Resource Requirements Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	# Replicas (regular deployment)	# Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
oc_nef_5gcagent	0.7	0.7	0	2	1	0.7	0.7	0
oc_nef_expiry_auditor	0.7	0.7	0	2	1	0.7	0.7	0
oc_capif_afmgr	0.7	0.7	0	2	1	0.7	0.7	0
oc_capif_apimgr	0.7	0.7	0	2	1	0.7	0.7	0
oc_capif_eventmgr	0.7	0.7	0	2	1	0.7	0.7	0
ocingress_gateway	1	1	0	2	1	1	1	0
ocegress_gateway	1	1	0	2	1	1	1	0
nrf-client	1	1	0	1	1	1	1	0
oc-app-info	1	1	0	1	1	1	1	0
oc-perf-info	1	1	0	1	1	1	1	0
oc-config-server	1	1	0	1	1	1	1	0
ocnef-trafficinfluence	0.5	0.5	0	1	1	0.5	0.5	0
nrf-client-discovery	1	1	0	1	1	1	1	0
oc_nef_diam_gateway	1	1	0	1	1	1	1	0
oc_nef_device_trigger	1	1	0	1	1	1	1	0
poolmanager	1	1	0	1	1	1	1	0
msisdnllessmosms	1	1	0	1	1	1	1	0
consoledataservice	0.5	0.5	0	1	1	0.5	0.5	0
NEF SUT Totals						19.9	19.9	0

ATS Resource Requirements details for NEF

This section describes the ATS resource requirements, which are needed to deploy NEF ATS successfully.

Table 3-7 ATS Resource Requirements Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	# Replicas (regular deployment)	# Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
ATS Behave	2	1	0	0	1	2	1	0
ocstub-nef-af	0.5	1	0	0	1	0.5	1	0
ocstub-nef-nrf	0.5	0.5	0	0	1	0.5	0.5	0
ocstub-nef-udm	0.5	1	0	0	1	0.5	1	0
ocstub-nef-gmlc	0.5	1	0	0	1	0.5	1	0

Table 3-7 (Cont.) ATS Resource Requirements Details

Microservice	CPUs Require d per Pod	Memory Require d per Pod (GB)	Storage PVC Require d per Pod (GB)	# Replica s (regular deploy ment)	# Replica s (ATS deploy ment)	CPUs Require d - Total	Memory Require d - Total (GB)	Storage PVC Require d - Total (GB)
ocstub-nef-bsf	1	1	0	2	1	1	1	0
ocstub-nef-pcf	1	1	0	2	1	1	1	0
ocstub-nef-udr	1	1	0	2	1	1	1	0
ocstub-diam-nef	1	0.5	0	2	1	1	1	0
ATS Totals						8	8.5	0

cnDBTier Resource Requirements Details for NEF

This section describes the cnDBTier resource requirements, which are needed to deploy NEF ATS successfully.

Note

For cnDBTier pods, a minimum of 4 worker nodes are required.

Table 3-8 cnDBTier Resource Requirements

Microservice	CPUs Require d per Pod	Memory Require d per Pod (GB)	Storage PVC Require d per Pod (GB)	# Replica s (regular deploy ment)	# Replica s (ATS deploy ment)	CPUs Require d - Total	Memory Require d - Total (GB)	Storage PVC Require d - Total (GB)
vrt-launcher-dt-1.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-dt-2.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-dt-3.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-dt-4.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-mt-1.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-mt-2.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-mt-3.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-sq-1.cluster.local	4	4	2	2	1	4	4	2
vrt-launcher-sq-2.cluster.local	4	4	2	2	1	4	4	2

Table 3-8 (Cont.) cnDBTier Resource Requirements

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	# Replicas (regular deployment)	# Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
vrt-launcher-db-installer.cluster.local	4	4	2	2	1	4	4	2
cnDBTier Totals						40	40	20

3.2.2 Downloading the ATS Package

Locating and Downloading ATS Images

To locate and download the ATS Image from MOS:

1. Log in to [My Oracle Support](#) using the appropriate credentials.
2. Select the **Patches & Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Network Exposure Function <release_number>* from the **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required ATS patch from the list. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the **<p*****_<release_number>_Tekelec>.zip** file to download the NEF ATS package file.
10. Untar the zip file to access all the ATS Images. The **<p*****_<release_number>_Tekelec>.zip** directory has following files:

```

ocats-nef-tools-pkg-24.1.0.0.0.tgz
ocats-nef-tools-pkg-24.1.0.0.0-README.txt
ocats-nef-tools-pkg-24.1.0.0.0.tgz.sha256
ocats-nef-custom-configtemplates-24.1.0.0.0.zip
ocats-nef-custom-configtemplates-24.1.0.0.0-README.txt

```

The ocats-nef-tools-pkg-24.1.0.0.0-README.txt file has all the information required for the package.

The ocats-nef-tools-pkg-24.1.0.0.0.tgz file has following images and charts packaged as tar files:

```
ocats-nef-tools-pkg-24.1.0.0.0.tgz
```

|

```

|_ _ _ _ _ ocats-nef-pkg-24.1.0.0.0.tgz
|
|_ _ _ _ _ ocats-nef-24.1.0.tgz (Helm Charts)
|
|_ _ _ _ _ ocats-nef-image-24.1.0.tar (Docker Images)
|
|_ _ _ _ _ OCATS-NEF-Readme.txt
|
|_ _ _ _ _ ocats-nef-24.1.0.tgz.sha256
|
|_ _ _ _ _ ocats-nef-image-24.1.0.tar.sha256
|
|_ _ _ _ _ ats_data-24.1.0.tar (ATS test scripts and Jenkins
data)
|
|_ _ _ _ _ ats_data-24.1.0.tar.sha256
|
|
|
|_ _ _ _ _ ocstub-nef-pkg-24.1.0.0.0.tgz
|
|_ _ _ _ _ ocstub-nef-24.1.0.tgz (Helm Charts)
|
|_ _ _ _ _ ocstub-nef-image-24.1.0.tar (Docker Images)
|
|_ _ _ _ _ ocstub-diam-nef-image-24.1.0.tar
|
|_ _ _ _ _ ocstub-diam-nef-image-24.1.0.tar.sha256
|
|_ _ _ _ _ OCSTUB-NEF-Readme.txt
|
|_ _ _ _ _ ocstub-nef-24.1.0.tgz.sha256
|
|_ _ _ _ _ ocstub-nef-image-24.1.0.tar.sha256

```

In addition to the above images and charts, there is a **ocats-nef-custom-configtemplates-24.1.0.0.0.zip** file in the package zip file. The **ocats-nef-custom-configtemplates-24.1.0.0.0-README.txt** file has information about this zip file.

ocats-nef-custom-configtemplates-24.1.0.0.0.zip

```

|
|_ _ _ _ _ ocats-nef-custom-values.yaml (Custom values file for
installation)
|
|_ _ _ _ _ ocats-nef-custom-serviceaccount.yaml (Template to create
custom service account)
|

```



```
|_ _ _ ocstub-nef-custom-values.yaml (Custom values file for stub installation)
```

11. Copy the tar file to the CNE, OCI, or Kubernetes cluster where you want to deploy ATS.

3.2.3 Pushing the Images to Customer Docker Registry

Preparing to deploy ATS and Stub Pod in Kubernetes Cluster

To deploy ATS and Stub Pod in Kubernetes Cluster:

1. Verify the checksums of tarballs mentioned in file Readme.txt.
2. Run the following command to extract tar file content.

```
tar -xvf ocats-nef-tools-pkg-24.1.0.0.0.tgz
```

The output of this command is:

```
ocats-nef-pkg-24.1.0.0.0.tgz
ocstub-nef-pkg-24.1.0.0.0.tgz
```

3. Run the following command to extract the helm charts and docker images of ATS.

```
tar -xvf ocats-nef-pkg-24.1.0.0.0.tgz
```

The output of this command is:

```
ocats-nef-image-24.1.0.tar
ocats-nef-24.1.0.tgz
OCATS-NEF-Readme.txt
```

Note

The OCATS-NEF-Readme.txt file has all the information required for the package.

4. Run the following command to untar the ocstub package.

```
tar -xvf ocstub-nef-pkg-24.1.0.0.0.tgz
```

The output of this command is:

```
ocstub-nef-image-24.1.0.tar
ocstub-nef-24.1.0.tgz
ocstub-diam-nef-image-24.1.0.tar
OCSTUB-NEF-Readme.txt
```

5. Run the following command to extract the content of the custom configuration templates:

```
unzip ocats-nef-custom-configtemplates-24.1.0.0.0.zip
```

The output of this command is:

```
ocats-nef-custom-values.yaml (Custom yaml file for deployment of OCATS-NEF)
ocats-nef-custom-serviceaccount.yaml (Custom yaml file for service account creation to help the customer if required)
ocstub-nef-custom-values.yaml (Custom yaml file for deployment of OCSTUB-NEF)
```

6. Run the following commands in your cluster to load the ATS docker image, 'ocats-nef-image-24.1.0.tar', and push it to your registry.

```
$ docker load -i ocats-nef-image-24.1.0.tar
```

```
$ docker tag ocats/ocats-nef:24.1.0 <local_registry>/ocats/ocats-nef:24.1.0
```

```
$ docker push <local_registry>/ocats/ocats-nef:24.1.0
```

7. Run the following commands in your cluster to load the ATS docker image, Stub docker image, 'ocstub-nef-image-24.1.0.tar' and push it to your registry.

```
$ docker load -i ocstub-nef-image-24.1.0.tar
```

```
$ docker tag ocats/ocstub-nef:24.1.0 <local_registry>/ocats/ocstub-nef:24.1.0
```

```
$ docker push <local_registry>/ocats/ocstub-nef:24.1.0
```

```
$ docker load -i ocstub-diam-nef-image-24.1.0.tar
```

```
$ docker tag ocats/ocstub-diam-nef:24.1.0 <local_registry>/ocats/ocstub-diam-nef:24.1.0
```

```
$ docker push <local_registry>/ocats/ocstub-diam-nef:24.1.0
```

8. Update the image name and tag in the ocats-nef-custom-values.yaml and ocstub-nef-custom-values.yaml files as required. For this, open the ocats-nef-custom-values.yaml and ocstub-nef-custom-values.yaml file and update the image.repository and image.tag parameters.

3.2.4 Configuring ATS

3.2.4.1 Enabling Static Port

1. To enable static port:

Note

ATS supports static port. By default, this feature is not available.

- In the **ocats-nef-custom-values.yaml** file under service section, set the **staticNodePortEnabled** parameter value to 'true' and **staticNodePort** parameter value with valid nodePort.

```
service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
  port: "8080"
```

```
staticNodePortEnabled: true
staticNodePort: "32385"
```

3.2.5 Deploying ATS, Stub and CNC Console in Kubernetes Cluster

Note

It is important to ensure that all the four components; ATS, Stub, NEF and CNC Console are in the same namespace.

To run NEF test cases, you need **seven** stubs. The service name of the stubs should be `ocnefsim-ocstub-svc-af`, `ocnefsim-ocstub-svc-nrf`, `ocnefsim-ocstub-svc-bsf`, `ocnefsim-ocstub-svc-gmlc`, `ocnefsim-ocstub-svc-pcf`, `ocnefsim-ocstub-svc-udm`, and `ocnefsim-ocstub-svc-udr`.

ATS and Stub supports Helm2 and Helm3 for deployment.

If the namespace does not exist, run the following command to create a namespace:

```
kubectl create namespace ocnef
```

Important

- It is mandatory to use the `<release_name>` as `ocnefsim` while installing stubs.
- The ATS deployment with NEF does not support the Persistent Volume (PV) feature. Therefore, the default value of the deployment `PVEnabled` parameter in the `ocats-nef-custom-values.yaml` must not be changed. By default, the parameter value is set to **false**.

Using Helm 2 for Deploying ATS:

```
helm install ocats-nef-24.1.0.tgz --name <release_name> --namespace
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm install ocats-nef-24.1.0.tgz --name ocats --namespace ocnef -f ocats-nef-
custom-values.yaml
```

Using Helm 2 for Deploying Stubs:

```
helm install ocstub-nef-24.1.0.tgz --name <release_name> --namespace
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm install ocstub-nef-24.1.0.tgz --name ocnefsim --namespace ocnef -f
ocstub-nef-custom-values.yaml
```

Using Helm 3 for Deploying ATS:

```
helm3 install -name <release_name> ocats-nef-24.1.0.tgz --namespace  
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm3 install -name ocats ocats-nef-24.1.0.tgz --namespace ocnef -f ocats-nef-  
custom-values.yaml
```

Using Helm 3 for Deploying Stubs:

```
helm3 install -name <release_name> ocstub-nef-24.1.0.tgz --namespace  
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm3 install -name ocnefsim ocstub-nef-24.1.0.tgz --namespace ocnef -f  
ocstub-nef-custom-values.yaml
```

3.2.5.1 Deploy and Configure CNC Console

Perform the following steps to deploy and configure the CNC Console:

1. Install the CNC Console 24.1.0 in the same namespace where NEF ATS is installed. For further information, see *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.
2. Create user and assign the necessary roles to access the NEF CNC Console GUI.

Note

Following are the different roles to be assigned:

- ADMIN
- NEF_WRITE
- NEF_READ
- default-roles-cncc

3.2.5.2 Creating Console secrets

Run the following command to create the console secrets that contains **iam** and **console** passwords created in [Deploy and Configure CNC Console](#).

Command:

```
kubectl create secret generic ocats-console-secret --from-  
literal=cnc_console_password=<cncc-console-password> --from-  
literal=cnc_iam_password=<cnc_iam_password> -n <namespace>
```

For example:

```
kubectl create secret generic ocats-console-secret --from-  
literal=cnc_console_password=Nefuser@1 --from-literal=cnc_iam_password=abc123  
-n ocnef
```

3.2.6 Verifying ATS Deployment

Run the following command to verify ATS deployment.

```
helm status <release_name>
```

Once ATS and Stub are deployed, run the following commands to check the pod and service deployment:

To check pod deployment:

```
kubectl get pod -n ocnef
```

To check service deployment:

```
kubectl get service -n ocnef
```

3.3 Installing ATS for NRF

Following are the ATS installation procedures for NRF:

1. [Downloading the ATS Package](#)
2. [Pushing the Images to Customer Docker Registry](#)
3. [Configuring ATS](#)
 - a. [Enabling Static Port](#)
 - b. [Enabling Aspen Service Mesh](#)
 - c. [Enabling Persistent Volume](#)
 - d. [Enabling NF FQDN Authentication](#)
4. [Deploying ATS and Stub in Kubernetes Cluster](#)
5. [Verifying ATS Deployment](#)
6. [Post-Installation Steps \(if Persistent Volume is Used\)](#)

3.3.1 Resource Requirements

This section describes the ATS resource requirements for NRF.

Overview - Total Number of Resources

The following table describes the overall resource usage in terms of CPUs, memory, and storage for the following:

- NRF SUT
- cnDBTier

- ATS

Table 3-9 NRF - Total Number of Resources

Resource Name	CPU	Memory (Gi)	Storage (Mi)
NRF SUT Totals	61	69	0
DBTier Totals	40.5	50.5	720
ATS Totals	7	6	0
Grand Total NRF ATS	108.5	125.5	720

NRF Pods Resource Requirements Details

For NRF Pods resource requirements, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

ATS Resource Requirements details for NRF

This section describes the ATS resource requirements, which are needed to deploy NRF ATS successfully.

Table 3-10 ATS Resource Requirements Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	# Replicas (regular deployment)	# Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
ATS Behave	2	1	1	1	1	2	1	0
ATS Stub (Python)	1	1	1	1	5	5	5	0
ATS Totals						7	6	0

cnDBTier Resource Requirements Details for NRF

This section describes the cnDBTier resource requirements, which are needed to deploy NRF ATS successfully.

Note

For cnDBTier pods, a minimum of 4 worker nodes are required.

Table 3-11 cnDBTier Services Resource Requirements

Service Name	Min Pod Replica #	Min CPU/Pod	Min Memory/Pod (in Gi)	PVC Size (in Gi)	Min Ephemeral Storage (Mi)
MGMT (ndbmcmd)	2	4	6	15	90
DB (ndbmtd)	4	5	5	4	90
SQL (ndbmssql)	2	4	5	8	90
SQL (ndbapssql)	2	2	3	1	90

Table 3-11 (Cont.) cnDBTier Services Resource Requirements

Service Name	Min Pod Replica #	Min CPU/Pod	Min Memory/Pod (in Gi)	PVC Size (in Gi)	Min Ephemeral Storage (Mi)
Monitor Service (db-monitor-svc)	1	0.4	490 (Mi)	NA	90
Backup Manager Service (db-backup-manager-svc)	1	0.1	130(Mi)	NA	90
Replication Service - Leader	1	2	2	2	90
Replication Service - Other	0	1	2	0	90

3.3.2 Downloading the ATS Package

Locating and Downloading ATS Images

To locate and download the ATS image from MOS:

1. Log in to [My Oracle Support](#) using the appropriate credentials.
2. Select the **Patches & Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Core Network Repository Function <release_number>* from the **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required ATS patch from the list. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the `<p*****_<release_number>_Tekelec>.zip` file to download the NRF ATS package file.
10. Untar the zip file to access all the ATS images. The `<p*****_<release_number>_Tekelec>.zip` directory has the following files:

```
ocats_ocnrf_csar_24_1_3_0_0.zip
ocats_ocnrf_csar_24_1_3_0_0.zip.sha256
ocats_ocnrf_csar_mcafee-24.1.3.0.0.log
```

11.

Note

The above zip file contains all the images and custom values required for 24.1.3 release of OCATS-OCNRF.

The `ocats_ocnrf_csar_24_1_3_0_0.zip` file has the following files and folders:

```
├── Definitions
│   ├── ocats_ocnrf_ats_tests.yaml
│   └── ocats_ocnrf.yaml
└── Files
```

```

|   |   |   | ChangeLog.txt
|   |   |   | Helm
|   |   |   | |   | ocats-ocnrf-24.1.3.tgz
|   |   |   | |   | Licenses
|   |   |   | |   | ocats-nrf-24.1.3.tar
|   |   |   | |   | Oracle.cert
|   |   |   | |   | ocstub-py-24.1.3.tar
|   |   |   | |   | Tests
|   |   |   | |   |
|   |   |   | |   | ocats_ocnrf.mf
|   |   |   | |   | Scripts
|   |   |   | |   | |   | ocats_ocnrf_custom_serviceaccount_24.1.3.yaml
|   |   |   | |   | |   | ocats_ocnrf_custom_values_24.1.3.yaml
|   |   |   | |   | |   | ocats_ocnrf_tests_jenkinsjobs_24.1.3.tgz
|   |   |   | |   |
|   |   |   | |   | TOSCA-Metadata
|   |   |   | |   | |   | TOSCA.meta

```

12. Copy the zip file to Kubernetes cluster where you want to deploy ATS.

3.3.3 Pushing the Images to Customer Docker Registry

Preparing to Deploy ATS and Stub Pod in Kubernetes Cluster

To deploy ATS and Stub Pod in Kubernetes Cluster:

1. Run the following command to extract tar file content:

```
unzip ocats_ocnrf_csar_24_1_3_0_0.zip
```

The following docker image tar files are located at **Files** folder:

- ocats-nrf-24.1.3.tar
 - ocstub-py-.tar
2. Run the following commands in your cluster to load the ATS docker image, 'ocats-nrf-24.1.3.tar' and Stub docker image 'ocstub-py-.tar', and push it to your registry.

```
$ docker load -i ocats-nrf-24.1.3.tar
```

```
$ docker load -i ocstub-py-.tar
```

```
$ docker tag ocats/ocats-nrf:24.1.3 <local_registry>/ocats/ocats-nrf:24.1.3
```

```
$ docker tag ocats/ocstub-py: <local_registry>/ocats/ocstub-py:
```

```
$ docker push <local_registry>/ocats/ocats-nrf:24.1.3
```

```
$ docker push <local_registry>/ocats/ocstub-py:
```

3. Create a copy of the custom values located at **Scripts/ocats_ocnrf_custom_values_24.1.3.yaml** and update it for image name, tag and other parameters as per the requirement.

3.3.4 Configuring ATS

3.3.4.1 Enabling Static Port

1. To enable static port:

Note

ATS supports static port. By default, this feature is not available.

- In the **ocats-ocnrf-custom-values.yaml** file under service section, set the **staticNodePortEnabled** parameter value to 'true' and **staticNodePort** parameter value with valid nodePort.

```
service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
  port: "8080"
  staticNodePortEnabled: true
  staticNodePort: "32385"
```

3.3.4.2 Enabling Aspen Service Mesh

To enable service mesh for ATS:

1. To enable service mesh, set the value for **serviceMeshCheck** to true. The following is a snippet of the service section in the yaml file:

```
service:
  customExtension:
    labels: {}
    annotations: {}
  type: LoadBalancer
  port: "8080"
  staticNodePortEnabled: true
  staticNodePort: "32385"
  serviceMeshCheck: true
```

2. If the ASM is not enabled on the global level for the namespace, run the following command to enable it before deploying the ATS:

```
kubectl label --overwrite namespace <namespace_name> istio-
injection=enabled
```

For example:

```
kubectl label --overwrite namespace ocnrf istio-injection=enabled
```

3. Add the following annotations under the **lbDeployments** and **nonlbDeployments** section of the global section in **ocats-nrf-custom-values.yaml** file for ATS deployment as follows:

```
traffic.sidecar.istio.io/excludeInboundPorts: "8080"
traffic.sidecar.istio.io/excludeOutboundPorts: "9090"
```

For example:

```
lbDeployments:
  labels: {}
  annotations:
    traffic.sidecar.istio.io/excludeInboundPorts: "8080"
    traffic.sidecar.istio.io/excludeOutboundPorts: "9090"
```

```
lbDeployments:
  labels: {}
  annotations:
    traffic.sidecar.istio.io/excludeInboundPorts: "8090"
    traffic.sidecar.istio.io/excludeOutboundPorts: "9090"
```

```
nonlbServices:
  labels: {}
  annotations: {}
```

```
nonlbDeployments:
  labels: {}
  annotations:
    traffic.sidecar.istio.io/excludeInboundPorts: "8090"
    traffic.sidecar.istio.io/excludeOutboundPorts: "9090"
```

4. Add the following annotations in OCNRF deployment to work with ATS in service mesh environment:

For example:

```
oracle.com/cnc: "true"
traffic.sidecar.istio.io/excludeInboundPorts: "9090,8095,8096,7,53"
traffic.sidecar.istio.io/excludeOutboundPorts: "9090,8095,8096,7,53"
```

```
lbDeployments:
  labels: {}
  annotations:
    oracle.com/cnc: "true"
    traffic.sidecar.istio.io/excludeInboundPorts: "9090,8095,8096,7,53"
    traffic.sidecar.istio.io/excludeOutboundPorts:
"9090,8095,8096,7,53"
```

```
nonlbServices:
  labels: {}
  annotations: {}
```

```
nonlbDeployments:
```

```
labels: {}
annotations:
  oracle.com/cnc: "true"
  traffic.sidecar.istio.io/excludeInboundPorts: "9090,8095,8096,7,53"
  traffic.sidecar.istio.io/excludeOutboundPorts:
    "9090,8095,8096,7,53"
```

① Note

If the above annotations are not provided in NRF deployment under lbDeployments and nonlbDeployments, all the metrics and alerts related test cases will fail.

3.3.4.3 Enabling Persistent Volume

ATS supports Persistent storage to retain ATS historical build execution data, test cases, and one-time environment variable configurations.

To enable persistent storage:

1. Create a PVC and associate the same to the ATS pod.
2. Set the **PVEnabled** flag to **true**.
3. Set **PVClaimName** to PVC that is created for ATS.

```
deployment:
  customExtension:
    labels: {}
    annotations: {}
    PVEnabled: true
    PVClaimName: "ocats-nrf-24.1.3-pvc"
```

For more details on Persistent Volume Storage, you can refer to [Persistent Volume for 5G ATS](#).

3.3.4.4 Enabling NF FQDN Authentication

① Note

This procedure is applicable only if the NF FQDN Authentication feature is being tested else, proceed to the "[Deploying ATS and Stub in Kubernetes Cluster](#)" section.

You must enable this feature while deploying Service Mesh. For more information on how to enable NF FQDN Authentication feature, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

However, there is some change in the ATS deployment process, which is as follows:

1. Use previously unzipped file "**ocats-nrf-custom-serviceaccount-24.1.3.yaml**" to create a service account. Add the following annotation in the "**ocats-nrf-custom-serviceaccount-24.1.3.yaml**" file where the kind is ServiceAccount.

```
"certificate.aspenmesh.io/customFields": '{ "SAN": { "DNS": [ "<NF-FQDN>" ] } }'
```

Sample format:

```
apiVersion: v1
kind: ServiceAccount
metadata:
  name: ocats-custom-serviceaccount
  namespace: ocnrf
  annotations:
    "certificate.aspenmesh.io/customFields": '{ "SAN": { "DNS":
[ "AMF.d5g.oracle.com" ] } }'
```

Note

"AMF.d5g.oracle.com" is the NF FQDN that you must provide in the serviceaccount DNS field.

2. Run the following command to create a service account:
`kubectl apply -f ocats-nrf-custom-serviceaccount-24.1.3.yaml`
3. Update the service account name in the **ocats-ocnrf-custom-values-24.1.3.yaml** file as follows:

```
ocats-nrf:
  serviceAccountName: "ocats-custom-serviceaccount"
```

3.3.5 Deploying ATS and Stub in Kubernetes Cluster

Note

It is important to ensure that all the three components; ATS, Stub and NRF are in the same namespace.

ATS and Stub supports Helm3 for deployment.

If the namespace does not exist, run the following command to create a namespace:

```
kubectl create namespace ocnrf
```

Using Helm for Deploying ATS:

```
helm install <release_name> ocats-ocnrf-24.1.3.tgz --namespace
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm install ocats ocats-ocnrf-24.1.3.tgz --namespace ocnrf -f ocats-ocnrf-
custom-values.yaml
```

Note

The above helm install command will deploy ATS along with the stub servers required for ATS executions, which include 1 ATS pod and 5 stub server pods.

3.3.6 Verifying ATS Deployment

Run the following command to verify ATS deployment.

```
helm status <release_name>
```

Once ATS and Stub are deployed, run the following command to check the pod and service deployment.

Checking Pod Deployment:

```
kubectl get pod -n ocnrf
```

Checking Service Deployment:

```
kubectl get service -n ocnrf
```

Figure 3-5 Checking Pod Deployment without Service Mesh

NAME	READY	STATUS	RESTARTS	AGE
ocats-ocats-nrf-658df6b9c9-vn9jv	1/1	Running	0	2d2h
ocnrf-alternate-route-688cb7bbc5-82whg	1/1	Running	0	7m34s
ocnrf-appinfo-74d5cf9c75-kvffz	1/1	Running	0	7m35s
ocnrf-egressgateway-6f94cc46cf-qxzjc	1/1	Running	0	7m29s
ocnrf-ingressgateway-5bd6779bc6-4z296	1/1	Running	0	7m24s
ocnrf-nfaccessstoken-7676857798-7fw8h	1/1	Running	0	7m35s
ocnrf-nfdiscovery-7ffdcdbd865-bqngj	1/1	Running	0	7m28s
ocnrf-nfregistration-68b597df4c-tz8rj	1/1	Running	0	7m26s
ocnrf-nfsubscription-65f89d7cc7-klxns	1/1	Running	0	7m28s
ocnrf-nrfartisan-d56dc784-bq7tp	1/1	Running	0	7m37s
ocnrf-nrfauditor-6bb5c8fb87-896xs	1/1	Running	0	7m38s
ocnrf-nrfcachedata-695985f988-jp7h9	1/1	Running	0	7m29s
ocnrf-nrfconfiguration-6dbcd88f97-22sl4	1/1	Running	0	7m36s
stub1-7bd7bf7574-nkjnm	1/1	Running	0	2d2h
stub2-7b9ddcbc49-579fv	1/1	Running	0	2d2h
stub3-f7794fbfb-qg5fm	1/1	Running	0	2d2h
stub4-7bdcd98667-f2gbn	1/1	Running	0	2d2h
stub5-899bcb978-2gh6m	1/1	Running	0	2d2h

Figure 3-6 Checking Service Deployment without Service Mesh

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
amf1-cluster1-net2-amf-5gc-mnc016-mcc310-3gppnetwork-org	LoadBalancer	10.233.3.237	<pending>	8080:32014/TCP, 8081:30259/TCP, 8443:31359/TCP	3d2h
notify-stub-service	LoadBalancer	10.233.58.68	<pending>	8080:31969/TCP, 8081:32610/TCP, 8443:32356/TCP	3d2h
notify-stub-service02	LoadBalancer	10.233.43.100	<pending>	8080:30213/TCP, 8081:32087/TCP, 8443:30194/TCP	3d2h
notify-stub-service03	LoadBalancer	10.233.31.107	<pending>	8080:30238/TCP, 8081:31109/TCP, 8443:32255/TCP	3d2h
ocats-ocats-nrf	LoadBalancer	10.233.16.95	<pending>	8080:32653/TCP	3d2h
ocnrf-alternate-route	ClusterIP	10.233.40.203	<none>	80/TCP	45h
ocnrf-altsvc-cache	ClusterIP	None	<none>	8080/TCP	45h
ocnrf-egressgateway	ClusterIP	10.233.49.192	<none>	8080/TCP	45h
ocnrf-egw-cache	ClusterIP	10.233.23.211	<none>	8080/TCP	45h
ocnrf-igw-cache	ClusterIP	None	<none>	8080/TCP	45h
ocnrf-ingressgateway	LoadBalancer	10.233.41.30	<pending>	80:31501/TCP	90d
ocnrf-nfaccessstoken	ClusterIP	10.233.49.192	<none>	8080/TCP	45h
ocnrf-nfdiscovery	ClusterIP	10.233.31.176	<none>	8080/TCP	45h
ocnrf-nfregistration	ClusterIP	10.233.47.105	<none>	8080/TCP	45h
ocnrf-nfsubscription	ClusterIP	10.233.56.47	<none>	8080/TCP	45h
ocnrf-nrfartisan	ClusterIP	10.233.49.170	<none>	8080/TCP	45h
ocnrf-nrfauditor	ClusterIP	10.233.5.30	<none>	8080/TCP	45h
ocnrf-nrfcachedata	ClusterIP	10.233.25.20	<none>	8080/TCP	45h
ocnrf-nrfconfiguration	LoadBalancer	10.233.27.134	<pending>	8080:31941/TCP	90d
ocnrf-ocnrf-app-info	ClusterIP	10.233.25.191	<none>	5906/TCP	45h
ocnrf-ocnrf-perf-info	ClusterIP	10.233.5.8	<none>	5905/TCP	45h
slf-stub-service01	LoadBalancer	10.233.62.84	<pending>	80:30178/TCP, 8081:30891/TCP, 8443:30223/TCP	3d2h

If ATS is deployed with side car of service mesh, ensure that both ATS and Stub pods have two containers in ready state and shows "2/2" as follows:

Figure 3-7 ATS and Stub Deployed with Service Mesh

NAME	READY	STATUS	RESTARTS	AGE
ocats-ocats-nrf-855d874786-6w7n8	2/2	Running	0	3d2h
ocnrf-alternate-route-57f4978469-ffn9w	2/2	Running	0	45h
ocnrf-appinfo-857c79bdc4-ngb5t	2/2	Running	0	45h
ocnrf-egressgateway-7d5dc479b-tfjffj	2/2	Running	0	45h
ocnrf-ingressgateway-6bfb85865f-d29xr	2/2	Running	0	45h
ocnrf-nfaccessstoken-5bbdb6f4c9-f5c6q	2/2	Running	0	45h
ocnrf-nfdiscovery-68db8f4c8d-dwfgw	2/2	Running	0	45h
ocnrf-nfregistration-759c668c77-rrk7q	2/2	Running	0	45h
ocnrf-nfsubscription-7c54b77669-v8msn	2/2	Running	0	45h
ocnrf-nrfartisan-5b8f48bff7-z985q	2/2	Running	0	45h
ocnrf-nrfauditor-6cdb597c67-qmgrp7	2/2	Running	0	45h
ocnrf-nrfcachedata-7c8967fbc8-npbwh	2/2	Running	0	45h
ocnrf-nrfcachedata-7c8967fbc8-sp4pl	2/2	Running	0	45h
ocnrf-nrfconfiguration-75b8c67dfb-r8qht	2/2	Running	0	45h
ocnrf-performance-5cf895dc75-4252k	2/2	Running	0	45h
ocnrf-performance-5cf895dc75-g9f8x	2/2	Running	0	45h
stub1-7698cf45d5-hncsh	2/2	Running	0	3d2h
stub2-58bd5bfbb4-h6xq9	2/2	Running	0	3d2h
stub3-5b596bbb59-488rm	2/2	Running	0	3d2h
stub4-5b84bc67d6-gmnqs	2/2	Running	0	3d2h
stub5-66cd68f97c-1s5kn	2/2	Running	0	3d2h

Figure 3-8 ATS and Stub Deployed with Service Mesh

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE
amf1-cluster1-net2-amf-5gc-mnc016-mcc310-3gppnetwork-org	LoadBalancer	10.233.3.237	<pending>	8080:32014/TCP, 8081:30259/TCP, 8443:31359/TCP	3d2h
notify-stub-service	LoadBalancer	10.233.58.68	<pending>	8080:31969/TCP, 8081:32610/TCP, 8443:32356/TCP	3d2h
notify-stub-service02	LoadBalancer	10.233.43.100	<pending>	8080:30213/TCP, 8081:32087/TCP, 8443:30194/TCP	3d2h
notify-stub-service03	LoadBalancer	10.233.31.107	<pending>	8080:30238/TCP, 8081:31109/TCP, 8443:32255/TCP	3d2h
ocats-ocats-nrf	LoadBalancer	10.233.16.95	<pending>	8080:32653/TCP	3d2h
ocnrf-alternate-route	ClusterIP	10.233.40.203	<none>	80/TCP	45h
ocnrf-altsvc-cache	ClusterIP	None	<none>	8080/TCP	45h
ocnrf-egressgateway	ClusterIP	10.233.23.211	<none>	8080/TCP	45h
ocnrf-egw-cache	ClusterIP	None	<none>	8080/TCP	45h
ocnrf-igw-cache	ClusterIP	None	<none>	8080/TCP	45h
ocnrf-ingressgateway	LoadBalancer	10.233.41.30	<pending>	80:31501/TCP	90d
ocnrf-nfaccessstoken	ClusterIP	10.233.49.192	<none>	8080/TCP	45h
ocnrf-nfdiscovery	ClusterIP	10.233.31.176	<none>	8080/TCP	45h
ocnrf-nfregistration	ClusterIP	10.233.47.165	<none>	8080/TCP	45h
ocnrf-nfsubscription	ClusterIP	10.233.56.47	<none>	8080/TCP	45h
ocnrf-nrfartisan	ClusterIP	10.233.49.170	<none>	8080/TCP	45h
ocnrf-nrfauditor	ClusterIP	10.233.5.30	<none>	8080/TCP	45h
ocnrf-nrfcachedata	ClusterIP	10.233.28.20	<none>	8080/TCP	45h
ocnrf-nrfconfiguration	LoadBalancer	10.233.27.134	<pending>	8080:31941/TCP	90d
ocnrf-ocnrf-app-info	ClusterIP	10.233.25.191	<none>	5906/TCP	45h
ocnrf-ocnrf-perf-info	ClusterIP	10.233.5.8	<none>	5905/TCP	45h
ocnrf-stub-service01	LoadBalancer	10.233.62.84	<pending>	80:30178/TCP, 8081:30891/TCP, 8443:30223/TCP	3d2h

3.3.7 Post-Installation Steps (if Persistent Volume is Used)

If persistent volume is used, follow the post-installation steps mentioned in the [Persistent Volume for 5G ATS](#) section.

3.4 Installing ATS for NSSF

The procedure to install ATS for NSSF covers three steps:

1. [Resource Requirements](#)
2. [Locating and downloading ATS Images](#)
3. [Deploying ATS and Stub Pod in Kubernetes Cluster](#)

3.4.1 Resource Requirements

Total Number of Resources

The total number of resource requirements are as follows:

Table 3-12 Total Number of Resources

Resource	CPUs	Memory(GB)	Storage(GB)
NSSF SUT Total	30.2	22	4
cnDBTier Total	40	40	20
ATS Total	5	5	0
Grand Total NSSF ATS	75.2	67	24

Resource Details

The details of resources required to install NSSF-ATS are as follows:

Table 3-13 Resource Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	Replicas (regular deployment)	Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
NSSF Pods								
ingressgateway	4	4	0	2	1	4	4	0
egressgateway	4	4	0	2	1	4	4	0
nsselection	4	2	0	2	1	4	2	0
nsavailability	4	2	0	2	1	4	2	0
nsconfig	2	2	0	1	1	2	2	0
nssubscription	2	2	0	1	1	2	2	0
nrf-client-discovery	1	1	0	2	1	1	1	0
nrf-client-management	1	1	0	1	1	1	1	0
appinfo	0.2	1	0	2	1	0.2	1	0
perfinfo	0.2	0.5	0	1	1	0.2	0.5	0
config-server	0.2	0.5	0	1	1	0.2	0.5	0
NSSF SUT Totals						22.6 CPU	20 GB	0
ATS								
ATS Behave	2	2	0	0	1	2	2	0
ATS AMF Stub (Python)	2	2	0	0	1	2	2	0

Table 3-13 (Cont.) Resource Details

Microservice	CPUs Required per Pod	Memory Required per Pod (GB)	Storage PVC Required per Pod (GB)	Replicas (regular deployment)	Replicas (ATS deployment)	CPUs Required - Total	Memory Required - Total (GB)	Storage PVC Required - Total (GB)
ATS NRF Stub (Python)	1	1	0	0	1	2	2	0
ATS Totals						4	3	0
cnDBTier Pods (minimum of 4 worker nodes required)								
virt-launcher-dt-1.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-dt-2.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-dt-3.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-dt-4.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-mt-1.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-mt-2.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-mt-3.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-sq-1.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-sq-2.cluster.local	4	4	2	2	1	4	4	2
virt-launcher-db-installer.cluster.local	4	4	2	2	1	4	4	2
cnDBTier Totals						40	40	20

3.4.2 Locating and Downloading ATS and Simulator Images

To locate and download the ATS Image from MOS:

1. Log in to [My Oracle Support](#) using the appropriate credentials.
2. Select the **Patches and Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Core Network Slice Selection Function <release_number>* from **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required ATS patch from the search results. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the **<p*****_<release_number>_Tekelec>.zip** file to download the NSSF ATS package file.
10. Untar the zip file to get *ocats-nssf* directory, which consists of all the ATS Images. The *ocats-nssf* directory has the following files:

Note

Prerequisites:

- NSSF ATS release should be "ocats".
- To run oauth test cases for NSSF, oauth secrets needs to be generated. For more information, see "Configuring Secrets to Enable Access Token " section in *Oracle communications Cloud Native Core, Network Slice Selection Function Installation, Upgrade, Fault Recovery Guide*.
- NSSF needs to point to NRF and Stub Servers.

The required changes in NSSF custom-values.yaml file are mentioned below

```
ocats-nssf
├── ocats-nssf-24.1.1.0.0-mcafee.log
├── ocats-nssf-custom-configtemplates-24.1.1.0.0-README.txt  file
contains all the information required for the package.
├── ocats-nssf-custom-configtemplates-24.1.1.0.0.zip  contains
serviceaccount,PVC File and Custom values file
├── ocats-nssf-tools-pkg-24.1.1.0.0-README.txt          file contains all the
information required for the package.
└── ocats-nssf-tools-pkg-24.1.1.0.0.tgz                file has the following
images and charts packaged as tar files
```

11. Untar the *ocats-nssf-tools-pkg-24.1.1.0.0.tgz* tar file

The structure of the file looks as given below:

```

ocats-nssf-tools-pkg-24.1.1.0.0
├── amfstub-24.1.1.tar          -AMF Stub Server Docker image
├── amfstub-24.1.1.tar.sha256
├── ats_data-24.1.1.tar        - ATS data, After untar "ocnssf_tests"
folder will gets created in which ATS feature files present
├── ats_data-24.1.1.tar.sha256
├── ocats-nssf-24.1.1.tar      - NSSF ATS Docker Image
├── ocats-nssf-24.1.1.tar.sha256
├── ocats-nssf-24.1.1.tgz      - ATS Helm Charts, after untar "ocats-
nssf" ats charts folder gets created.
├── ocats-nssf-24.1.1.tgz.sha256
└── README.md

```

12. Copy the ocats-nssf-tools-pkg-24.1.1.0.0.tgz tar file to the CNE or Kubernetes cluster where you want to deploy ATS.
13. Along with the above packages, there is ocats-nssf-custom-configtemplates-24.1.1.0.0.zip at the same location. The readme file ocats-nssf-custom-configtemplates-24.1.1.0.0-README.txt contains information about the content of this zip file.

Content of ocats-nssf-custom-configtemplates-24.1.1.0.0.zip is as below:

```

nssf_ats_pvc_24.1.1.yaml          - NSSF ATS PVC file
ocats_nssf_custom_values_24.1.1.yaml - NSSF ATS custom values
file which needs to be used while installing ATS
ocats_ocnssf_custom_serviceaccount_24.1.1.yaml- Template to Create ATS
service account

```

Copy these files to OCCNE or Kubernetes cluster where you want to deploy ATS.

```

ocats-nssf-custom-configtemplates-24.1.1.0.0.zip
-----nssf_ats_pvc_24.1.1.yaml          - NSSF ATS PVC file
-----ocats_nssf_custom_values_24.1.1.yaml - NSSF ATS custom
values file which needs to be used while installing ATS
-----ocats_ocnssf_custom_serviceaccount_24.1.1.yaml- Template to Create
ATS service account

```

```

ocats-nssf-tools-pkg-24.1.1.0.0
├── amfstub-24.1.1.tar.          -AMF Stub Server Docker image
├── ats_data-24.1.1.tar.        - ATS data, After untar "ocnssf_tests"
folder will gets created in which ATS feature files present
├── ocats-nssf-24.1.1.tar.      - NSSF ATS Docker Image
├── ocats-nssf-24.1.1.tgz.      - ATS Helm Charts, after untar "ocats-
nssf" ats charts folder gets created.

```

3.4.3 Deploying ATS in Kubernetes Cluster

To deploy ATS in Kubernetes Cluster:

1. Verify checksums of the tarballs mentioned in the file `Readme.txt`.
2. Run the following commands to extract tar file content, Helm charts, and Docker images of ATS:

```
tar -xvzf ocats-nssf-tools-pkg-24.1.1.0.0.tgz
```

The output of this command will return the following files:

```
amfstub-24.1.1.tar
amfstub-24.1.1.tar.sha256
ats_data-24.1.1.tar
ats_data-24.1.1.tar.sha256
ocats-nssf-24.1.1.tar
ocats-nssf-24.1.1.tar.sha256
ocats-nssf-24.1.1.tgz
ocats-nssf-24.1.1.tgz.sha256
Readme.txt
```

3. **NSSF-ATS and Stub Images Load and Push:** Run the following commands in your cluster to load the ocats image and amf stubserver image:

Docker Commands:

```
docker load -i ocats-nssf-<version>.tar
```

```
docker load -i amfstub-<version>.tar
```

Examples:

```
docker load -i ocats-nssf-24.1.1.tar
```

```
docker load -i amfstub-24.1.1.tar
```

Podman Commands:

```
podman load -i ocats-nssf-<version>.tar
```

```
podman load -i amfstub-<version>.tar
```

Examples:

```
podman load -i ocats-nssf-24.1.1.tar
```

```
podman load -i amfstub-24.1.1.tar
```

4. Run the following commands to tag and push the ATS image registry.

- a. Run the following commands to grep the image:

```
docker images | grep ocats-nssf
```

```
docker images | grep amfstub
```

- b. Copy the Image ID from the output of the grep command and change the tag (version number) to your registry.

Docker Commands:

```
docker tag <Image_ID> <your-registry-name/ocats-nssf:<tag>>
```

```
docker push <your-registry-name/ocats-nssf:<tag>>
```

```
docker tag <Image_ID> <your-registry-name/amfstub:<tag>>
```

```
docker push <your-registry-name/amfstub:<tag>>
```

Podman Commands:

```
podman tag <Image_ID> <your-registry-name/ats/ocats-nssf:<tag>>
```

```
podman push <your-registry-name/ocats-nssf:<tag>>
```

```
podman tag <Image_ID> <your-registry-name/amfstub:<tag>>
```

```
podman push <your-registry-name/amfstub:<tag>>
```

5. **ATS Helm Charts** : Run the following command to get "ATS" Helm charts as shown below:

```
tar -xvzf ocats-nssf-24.1.1.tgz
```

The above command creates "ocats-nssf" helm charts of ATS.

6. **ATS Data**: Run the following command to get ATS data, which contains feature files and data:

```
tar -xvf ats_data-24.1.1.tar
```

The above command creates "ocnssf_tests" ATS feature files data, which needs to be copied inside after the ATS installation is complete.

7. <Optional> Go to certificate folder inside ocats-nssf and run the following command:

```
kubectrl create secret generic ocnssf-secret --from-file=certificates/rsa_private_key_pkcs1.pem --from-file=certificates/trust.txt --from-file=certificates/key.txt --from-file=certificates/ocnssf.cer --from-file=certificates/caroot.cer -n ocnssf
```

8. **ATS Custom Values File Changes**: Update the image name and tag in the ocats_nssf_custom_values_24.1.1.yaml file as required.

- a. For this, open the ocats_nssf_custom_values_24.1.1.yaml file

- b. Update the `image.repository` and `image.tag` parameters for `ocats-nssf`, `ocats-amf`, and `ocats-nrf`.
 - c. Save and close the file after making the updates.
9. <Optional>**To enable static port:**

Note

ATS supports static port. By default, this feature is not available. In the `ocats-nssf/values.yaml` file under service section, set the value of `staticNodePortEnabled` parameter as 'true' and provide a valid `nodePort` value for `staticNodePort`.

10. **ATS Service Account Creation:** In `ocats-nssf-custom-serviceaccount_24.1.1.yaml`, change namespace as below:

```
sed -i "s/changeme-ocats/${namespace}/g"
ocats_ocnssf_custom_serviceaccount_24.1.1.yaml
```

11. Run the following command to apply `ocats-nssf-custom-serviceaccount_24.1.1.yaml` file:
- ```
kubectl apply -f <serviceaccount.yaml file> -n <namespace_name>
```

**For example:**

```
kubectl apply -f ocats-nssf-custom-serviceaccount_24.1.1.yaml -n ocnssf
```

12. **ATS Helm Release Name Update:** If NSSF ATS helm release name is changed from "ocats" to any other value while installing ATS, that needs to be updated in NSSF custom-values.yaml file. Here is a NSSF custom-values.yaml file snippet where ATS helm release name is "ocats":

```
#Static virtual FQDN Config
staticVirtualFqdns:
 - name: https://abc.test.com
 alternateFqdns:
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20
 - name: http://xyz.test.com
 alternateFqdns:
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20

nrf-client:
 # This config map is for providing inputs to NRF-Client
 configmapApplicationConfig:
 &configRef
```

```

Config-map to provide inputs to Nrf-Client
primaryNrfApiRoot - Primary NRF Hostname and Port
SecondaryNrfApiRoot - Secondary NRF Hostname and Port
retryAfterTime - Default downtime(in Duration) of an NRF detected
to be unavailable.
nrfClientType - The NfType of the NF registering
nrfClientSubscribeTypes - the NfType for which the NF wants to
subscribe to the NRF.
appProfiles - The NfProfile of the NF to be registered with NRF.
enableF3 - Support for 29.510 Release 15.3
enableF5 - Support for 29.510 Release 15.5
renewalTimeBeforeExpiry - Time Period(seconds) before the
Subscription Validity time expires.
validityTime - The default validity time(days) for subscriptions.
enableSubscriptionAutoRenewal - Enable Renewal of Subscriptions
automatically.
acceptAdditionalAttributes - Enable additionalAttributes as part
of 29.510 Release 15.5
retryForCongestion - The duration(seconds) after which nrf-client
should retry to a NRF server found to be congested.
profile: |-
 [appcfg]
 primaryNrfApiRoot=ocats-nrf-stubserver.changeme-ocats:8080
 secondaryNrfApiRoot=ocats-nrf-stubserver.changeme-ocats:8080
 nrfScheme=http

```

sbiRouting:

```

sbiRoutingDefaultScheme: http
peerConfiguration:
- id: peer1
 host: ocats-amf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v1"
- id: peer2
 host: ocats-amf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v2"
- id: peer3
 host: ocats-nrf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v3"

```

In above snippet, ATS Helm release name is "ocats". For example, if ATS helm release name is changed from "ocats" to "ocatsnssf" while installing, NSSF custom-values.yaml file should be updated as below snippet:

```

#Static virtual FQDN Config
staticVirtualFqdns:
- name: https://abc.test.com
 alternateFqdns:

```

```

- target:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
- target:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20
- name: http://xyz.test.com
 alternateFqdns:
 - target:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
 - target:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20

nrf-client:
 # This config map is for providing inputs to NRF-Client
 configmapApplicationConfig:
 &configRef
 # Config-map to provide inputs to Nrf-Client
 # primaryNrfApiRoot - Primary NRF Hostname and Port
 # SecondaryNrfApiRoot - Secondary NRF Hostname and Port
 # retryAfterTime - Default downtime(in Duration) of an NRF detected
to be unavailable.
 # nrfClientType - The NfType of the NF registering
 # nrfClientSubscribeTypes - the NfType for which the NF wants to
subscribe to the NRF.
 # appProfiles - The NfProfile of the NF to be registered with NRF.
 # enableF3 - Support for 29.510 Release 15.3
 # enableF5 - Support for 29.510 Release 15.5
 # renewalTimeBeforeExpiry - Time Period(seconds) before the
Subscription Validity time expires.
 # validityTime - The default validity time(days) for subscriptions.
 # enableSubscriptionAutoRenewal - Enable Renewal of Subscriptions
automatically.
 # acceptAdditionalAttributes - Enable additionalAttributes as part
of 29.510 Release 15.5
 # retryForCongestion - The duration(seconds) after which nrf-client
should retry to a NRF server found to be congested.
 profile: |-
 [appcfg]
 primaryNrfApiRoot=ocatsnssf-nrf-stubserver.changeme-ocats:8080
 secondaryNrfApiRoot=ocatsnssf-nrf-stubserver.changeme-ocats:8080
 nrfScheme=http

sbiRouting:

sbiRoutingDefaultScheme: http
peerConfiguration:
 - id: peer1
 host:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v1"
 - id: peer2

```

```

 host:ocatsnssf-amf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v2"
- id: peer3
 host:ocatsnssf-nrf-stubserver.changeme-ocats
 port: 8080
 apiPrefix: "/"
 healthApiPath: "/health/v3"

```

- 13. Pointing NSSF to Stub Servers:** Follow this step to point out NSSF to NRF-Stubserver and AMF-Stubserver in NSSF custom values file:

```
sed -i "s/changeme-ocats/${namespace}/g" $NSSF_CUSTOM_DEPLOY_FILE
```

For example:

```
sed -i "s/changeme-ocats/${namespace}/g" ocnssf_custom_values_24.1.1.yaml
```

The NSSF custom values snippet is as follows:

```

nrf-client:
 # This config map is for providing inputs to NRF-Client
 configmapApplicationConfig:
 &configRef
 # primaryNrfApiRoot - Primary NRF Hostname and Port
 # SecondaryNrfApiRoot - Secondary NRF Hostname and Port
 profile: |-
 [appcfg]
 primaryNrfApiRoot=ocats-nrf-stubserver.changeme-ocats:8080
 secondaryNrfApiRoot=ocats-nrf-stubserver.changeme-ocats:8080

staticVirtualFqdns:
- name: https://abc.test.com
 alternateFqdns:
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20
- name: http://xyz.test.com
 alternateFqdns:
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 10
 - target: ocats-amf-stubserver.changeme-ocats
 port: 8080
 priority: 20

```

- 14. Deploy ATS as shown below.** NSSF ATS Helm release name should be "ocats".  
`helm install <release_name> <charts> -n <namespace_name> -f <custom_values file> --version <helm-chart-version>`

**For example:**



```
helm install ocats ocats-nssf -n ocnssf -f
ocats_nssf_custom_values_24.1.1.yaml --version 24.1.1
```

Running the above command creates the following three pods:

```
ocats-amf-stubserver
ocats-nrf-stubserver
ocats-nssf
```

15. Run the following command to verify the ATS deployment:  
`helm status <release_name>`

The following screenshot is an example of a successful ATS deployment, where STATUS:DEPLOYED is an indicator of the same.

```
[opc@ocnssf-oci-phx-einstein-bastion-01]$ helm status ocats
NAME: ocats
LAST DEPLOYED: Tue Mar 5 02:22:08 2024
NAMESPACE: nssf-release
STATUS: deployed
REVISION: 1
TEST SUITE: None
```

## 3.5 Installing ATS for Policy

Installing ATS for Policy procedure consists of the following two steps:

1. Locating and downloading the ATS package
2. Deploying ATS and stub pods in Kubernetes cluster

This includes installation of nine stubs (nf1stub, nf11stub, nf12stub, nf2stub, nf21stub, nf22stub, nf3stub, nf31stub, nf32stub), ocamf stub, ocdns-bind stub, ocldap-stub, and Policy ATS in the namespace where CNC Policy is deployed.

### 3.5.1 Resource Requirements

This section describes the ATS resource requirements for CNC Policy.

#### Overview - Total Number of Resources

The following table describes the overall resource usage in terms of CPUs, and memory for the following:

- PCF SUT
- cnDBTier
- ATS

**Table 3-14 PCF - Total Number of Resources**

| Resource Name  | Non-ASM CPU | Non-ASM Memory (GB) | ASM CPU | ASM Memory (GB) |
|----------------|-------------|---------------------|---------|-----------------|
| PCF SUT Total  | 219         | 197                 | 293     | 244             |
| ATS Total      | 26          | 28                  | 54      | 42              |
| CnDBTier Total | 107.1       | 175.2               | 137.1   | 190.2           |

**Table 3-14 (Cont.) PCF - Total Number of Resources**

| Resource Name       | Non-ASM CPU | Non-ASM Memory (GB) | ASM CPU | ASM Memory (GB) |
|---------------------|-------------|---------------------|---------|-----------------|
| Grand Total PCF ATS | 352.1       | 400.2               | 484.1   | 476.2           |

**PCF Pods Resource Requirements Details**

This section describes the resource requirements, which are needed to deploy Policy ATS successfully.

**Table 3-15 PCF Pods Resource Requirements Details**

| Policy Microservices | Max CPU | Memory (GB) | Max Replica | Non-ASM Total CPU | Non-ASM Memory (GB) | ASM Total CPU | ASM Total Memory (GB) | Isito ASM CPU | Isito ASM Memory (GB) |
|----------------------|---------|-------------|-------------|-------------------|---------------------|---------------|-----------------------|---------------|-----------------------|
| oc-app-info          | 2       | 2           | 1           | 2                 | 2                   | 4             | 3                     | 2             | 1                     |
| oc-bulwark           | 8       | 6           | 2           | 16                | 12                  | 20            | 14                    | 2             | 1                     |
| oc-diam-connect or   | 4       | 2           | 2           | 8                 | 4                   | 12            | 6                     | 2             | 1                     |
| oc-diam-gateway      | 4       | 2           | 1           | 4                 | 2                   | 6             | 3                     | 2             | 1                     |
| alternate-route      | 2       | 4           | 1           | 2                 | 4                   | 4             | 5                     | 2             | 1                     |
| oc-config-server     | 4       | 2           | 1           | 4                 | 2                   | 6             | 3                     | 2             | 1                     |
| ocegress_gateway     | 4       | 6           | 1           | 4                 | 6                   | 6             | 7                     | 2             | 1                     |
| ocingress_gateway    | 5       | 6           | 1           | 5                 | 6                   | 7             | 7                     | 2             | 1                     |
| nrf-client-disc      | 4       | 2           | 2           | 8                 | 4                   | 12            | 6                     | 2             | 1                     |
| nrf-client-mngt      | 1       | 1           | 2           | 2                 | 2                   | 6             | 4                     | 2             | 1                     |
| oc-audit             | 2       | 4           | 1           | 2                 | 4                   | 4             | 5                     | 2             | 1                     |
| oc-config-mgmt       | 4       | 2           | 2           | 8                 | 4                   | 12            | 6                     | 2             | 1                     |
| oc-ldap-gateway      | 4       | 2           | 2           | 8                 | 8                   | 12            | 10                    | 2             | 1                     |
| oc-policy-ds         | 7       | 8           | 2           | 14                | 16                  | 18            | 18                    | 2             | 1                     |
| oc-pre               | 4       | 4           | 2           | 8                 | 8                   | 12            | 10                    | 2             | 1                     |

**Table 3-15 (Cont.) PCF Pods Resource Requirements Details**

| Policy Microservices | Max CPU | Memory (GB) | Max Replica | Non-ASM Total CPU | Non-ASM Memory (GB) | ASM Total CPU | ASM Total Memory (GB) | Isito ASM CPU | Isito ASM Memory (GB) |
|----------------------|---------|-------------|-------------|-------------------|---------------------|---------------|-----------------------|---------------|-----------------------|
| oc-query             | 2       | 1           | 2           | 4                 | 2                   | 8             | 4                     | 2             | 1                     |
| oc-soap-connect or   | 4       | 4           | 2           | 8                 | 8                   | 12            | 10                    | 2             | 1                     |
| oc-pcf-am            | 8       | 8           | 2           | 16                | 16                  | 20            | 18                    | 2             | 1                     |
| oc-pcf-sm            | 7       | 10          | 2           | 14                | 20                  | 18            | 22                    | 2             | 1                     |
| oc-pcf-ue            | 8       | 6           | 2           | 16                | 12                  | 20            | 24                    | 2             | 1                     |
| oc-pcrf-core         | 8       | 8           | 2           | 16                | 16                  | 0             | 18                    | 2             | 1                     |
| oc-perf-info         | 2       | 2           | 2           | 4                 | 4                   | 8             | 6                     | 2             | 1                     |
| oc-binding           | 6       | 8           | 1           | 6                 | 8                   | 8             | 9                     | 2             | 1                     |
| oc-udr-connect or    | 6       | 4           | 2           | 12                | 8                   | 16            | 10                    | 2             | 1                     |
| oc-chf-connect or    | 6       | 4           | 2           | 12                | 8                   | 16            | 10                    | 2             | 1                     |
| usage-mon            | 5       | 4           | 2           | 10                | 8                   | 14            | 10                    | 2             | 1                     |
| nwdaf-agent          | 2       | 1           | 1           | 2                 | 1                   | 4             | 2                     | 2             | 1                     |
| notifier             | 2       | 1           | 2           | 4                 | 2                   | 8             | 4                     | 2             | 1                     |
| Policy Totals        |         |             |             | 219               | 197                 | 293           | 244                   |               |                       |

**ATS Resource Requirements details for Policy**

This section describes the ATS resource requirements, which are needed to deploy Policy ATS successfully.

**Table 3-16 ATS Resource Requirements Details**

| ATS Microservices | Max CPU | Max Memory (GB) | Max Replica | Non-ASM Total CPU | Non-ASM Total Memory (GB) | ASM Total CPU | ASM Total Memory (GB) | Isito ASM CPU | Isito ASM Memory (GB) |
|-------------------|---------|-----------------|-------------|-------------------|---------------------------|---------------|-----------------------|---------------|-----------------------|
| ocstub1-py        | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub2-py        | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub3-py        | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |

Table 3-16 (Cont.) ATS Resource Requirements Details

| ATS Microservices | Max CPU | Max Memory (GB) | Max Replica | Non-ASM Total CPU | Non-ASM Total Memory (GB) | ASM Total CPU | ASM Total Memory (GB) | Isito ASM CPU | Isito ASM Memory (GB) |
|-------------------|---------|-----------------|-------------|-------------------|---------------------------|---------------|-----------------------|---------------|-----------------------|
| ocstub1 1-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub1 2-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub2 1-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub2 2-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub3 1-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocstub3 2-py      | 2       | 2               | 1           | 2                 | 2                         | 4             | 3                     | 2             | 1                     |
| ocamf-stub        | 1       | 1               | 1           | 1                 | 1                         | 3             | 2                     | 2             | 1                     |
| ocats-policy      | 4       | 6               | 1           | 4                 | 6                         | 6             | 7                     | 2             | 1                     |
| ocdns-bind        | 1       | 1               | 1           | 1                 | 1                         | 3             | 2                     | 2             | 1                     |
| ocldap-org1       | 1       | 1               | 1           | 1                 | 1                         | 3             | 2                     | 2             | 1                     |
| ocdiam-sim        | 1       | 1               | 1           | 1                 | 1                         | 3             | 2                     | 2             | 1                     |
| ATS Totals        |         |                 |             | 26                | 28                        | 54            | 42                    |               |                       |

**cnDBTier Resource Requirements Details for Policy ATS**

This section describes the cnDBTier resource requirements, which are needed to deploy Policy ATS successfully.

**Note**

For cnDBTier pods, a minimum of 4 worker nodes are required.

Table 3-17 cnDBTier Resource Requirements Details

| cnDBTier Microservices | Min CPU | Min Memory (GB) | Min Replica | Total CPU | Total Memory (GB) | ASM Total CPU | ASM Total Memory (GB) | Isito ASM CPU | Isito ASM Memory (GB) |
|------------------------|---------|-----------------|-------------|-----------|-------------------|---------------|-----------------------|---------------|-----------------------|
| db_monitor_svc         | 1       | 1               | 1           | 1         | 1                 | 3             | 2                     | 2             | 1                     |
| db_replication_svc     | 2       | 12              | 1           | 2         | 12                | 4             | 13                    | 2             | 1                     |

Table 3-17 (Cont.) cnDBTier Resource Requirements Details

| cnDBTier<br>Microservices    | Min<br>CPU | Min<br>Memory<br>(GB) | Min<br>Replica | Total<br>CPU | Total<br>Memory<br>(GB) | ASM<br>Total<br>CPU | ASM<br>Total<br>Memory<br>(GB) | Isito<br>ASM<br>CPU | Isito<br>ASM<br>Memory<br>(GB) |
|------------------------------|------------|-----------------------|----------------|--------------|-------------------------|---------------------|--------------------------------|---------------------|--------------------------------|
| db_backup_manager_svc        | 0.1        | 0.2                   | 1              | 0.1          | 0.2                     | 2.1                 | 1.2                            | 2                   | 1                              |
| ndbapp<br>mysqld             | 8          | 10                    | 4              | 32           | 40                      | 40                  | 44                             | 2                   | 1                              |
| ndbmgm<br>d                  | 4          | 10                    | 2              | 8            | 20                      | 12                  | 22                             | 2                   | 1                              |
| ndbmtd                       | 10         | 18                    | 4              | 40           | 72                      | 48                  | 76                             | 2                   | 1                              |
| ndbmys<br>qld                | 8          | 10                    | 2              | 16           | 20                      | 20                  | 22                             | 2                   | 1                              |
| db_infra<br>_moditor<br>_svc | 8          | 10                    | 1              | 8            | 10                      | 8                   | 10                             |                     |                                |
| DB Tier Total                |            |                       |                | 107.1        | 175.2                   | 137.1               | 190.2                          |                     |                                |

**Note**

The requirements shown in the above table for CnDBTier are the default numbers and must be changed as per the deployment requirements.

## 3.5.2 Downloading the ATS Package

This section provides information on how to locate and download the Policy ATS package file from My Oracle Support (MOS).

### Locating and Downloading Policy ATS Package

To locate and download the ATS package from MOS, perform the following steps:

1. Log in to [My Oracle Support](#) using the valid credentials.
2. Select the **Patches & Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter Oracle Communications Cloud Native Core - 5G in the **Product** field.
5. Select Oracle Communications Cloud Native Network Repository Function <release\_number> using the drop-down menu of the **Release** field.
6. Click **Search**. The list of **Patch Advanced Search Results** appears.
7. Select the required ATS patch from the list. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the <p>\*\*\*\*\*</p>\_<release\_number>\_Tekelec>.zip file to download the CNC Policy ATS package file.

10. Untar the gzip file `ocats-policy-tools-24.1.0.0.0.tgz` to access the following files:

```
ocats-policy-pkg-24.1.0.0.0.tgz
ocdns-pkg-24.1.0.0.0.tgz
ocamf-pkg-24.1.0.0.0.tgz
oc-ldap-org1-pkg-24.1.0.0.0.tgz
ocstub-pkg-24.1.0.0.0.tgz
ocdiam-pkg-24.1.0.0.0.tgz
```

The contents included in each of these files are as follow:

```
|
|
|_ _ _ocats-policy-pkg-24.1.0.0.0.tgz
|
| |_ _ _ _ _ocats-policy-24.1.0.tgz (Helm Charts)
|
| |_ _ _ _ _ocats-policy-image-24.1.0.tar (Docker Images)
|
| |_ _ _ _ _ocats-policy-data-24.1.0.tgz (Policy ATS and Jenkins
job Data)
|
|
|_ _ _ocstub-pkg-24.1.0.0.0.tgz
|
| |_ _ _ _ _ocstub-py-24.1.0.tgz(Helm Charts)
|
| |_ _ _ _ _ocstub-py-image-24.1.0.tar (Docker Images)
|
|
|_ _ _ocdns-pkg-24.1.0.0.0.tgz
|
| |_ _ _ _ _ocdns-bind-24.1.0.tgz(Helm Charts)
|
| |_ _ _ _ _ocdns-bind-image-24.1.0.tar (Docker Images)
|
|
|_ _ _ocamf-pkg-24.1.0.0.0.tgz
|
| |_ _ _ _ _ocamf-stub-24.1.0.tgz(Helm Charts)
|
| |_ _ _ _ _ocamf-stub-image-24.1.0.tar (Docker Images)
|
|
|_ _ _oc-ldap-org1-pkg-24.1.0.0.0.tgz
|
| |_ _ _ _ _oc-ldap-org1-24.1.0.tgz(Helm Charts)
|
| |_ _ _ _ _oc-ldap-org1-image-24.1.0.tar (Docker Images)
```

```
|
|_ _ _ _ _ ocdiam-pkg-24.1.0.0.0.tgz
| |_ _ _ _ _ ocdiam-sim-24.1.0.tgz(Helm Charts)
| |_ _ _ _ _ ocdiam-sim-image-24.1.0.tar (Docker Images)
```

11. Copy the tar file from the downloaded package to OCCNE, OCI, or Kubernetes cluster where you want to deploy ATS.

### 3.5.3 Pushing the Images to Customer Docker Registry

#### Preparing to deploy ATS and Stub Pods in Kubernetes Cluster

To deploy ATS and stub pods in Kubernetes Cluster, perform the following steps:

1. Run the following command to extract tar file content.  
tar -zxvf ocats-policy-tools-24.1.0.0.0.tgz

The following is the output of this command:

```
ocats-policy-pkg-24.1.0.0.0.tgz
ocstub-pkg-24.1.0.0.0.tgz
ocdns-pkg-24.1.0.0.0.tgz
ocamf-stub-24.1.0.0.0.tgz
oc-ldap-org1-24.1.0.tgz
ocdiam-pkg-24.1.0.0.0.tgz
```

2. Run the following command to extract the helm charts and docker images of ATS:  
tar -zxvf ocats-policy-pkg-24.1.0.0.0.tgz

The following is the output:

```
ocats-policy-24.1.0.tgz
ocats-policy-images-24.1.0.tar
ocats-policy-data-24.1.0.tgz
```

3. Run the following command to load the ATS docker image:  
docker load --input ocats-policy-images-24.1.0.tar
4. Run the following commands to tag and push the ATS images:

```
docker tag ocats-policy:24.1.0 <registry>/ocats-policy:24.1.0
docker push <registry>/ocats-policy:24.1.0
```

#### Example:

```
docker tag ocats-policy:24.1.0 localhost:5000/ocats-policy:24.1.0
docker push localhost:5000/ocats-policy:24.1.0
```

**Note**

If you are using Podman instead of Docker, replace docker with podman in all the docker commands given in this document.

5. Run the following command to untar the helm charts – 24.1.0.  
tar -zxvf ocats-policy-24.1.0.tgz

**Note**

atsFeatures section is newly introduced in values.yaml which helps Engineering team to control feature deliveries over the releases. It is not advisable to update any of the following flags without Engineering team's permission.

```
atsFeatures: ## DO NOT UPDATE this section without Engineering
team's permission
 testCaseMapping: true # To display Test cases on
GUI along with Features
 logging: true # To enable feature to
collect applogs in case of failure
 lightWeightPerformance: false # The Feature is not
implemented yet
 executionWithTagging: true # To enable Feature/Scenario
execution with Tag
 scenarioSelection: false # The Feature is not
implemented yet
 parallelTestCaseExecution: true # To run ATS features parallel
parallelFrameworkChangesIntegrated: true # To run ATS features
parallel
 mergedExecution: false # To execute ATS Regression
and NewFeatures pipelines together in merged manner
 individualStageGroupSelection: false # The Feature is not
implemented yet
 parameterization: true # When set to false, the
Configuration_Type parameter on the GUI will not be available.
 atsApi: true # To trigger ATS using ATS API
 healthcheck: true # TO enable/disable ATS
Health Check.
 atsGuiTLSEnabled: false # To run ATS GUI in https
mode.
 atsCommunicationTLSEnabled: false #If set to true, ATS will get
necessary variables to communicate with SUT, Stub or other NFs with
TLS enabled. It is not required in ASM environment.
```

6. Update the registry name, image name and tag in the ocats-policy/values.yaml file as required. For this, you need to update the image.repository and image.tag parameters in the ocats-policy/values.yaml file.



## 3.5.4 Configuring ATS

### 3.5.4.1 Enabling Static Port

To enable static port, in the `ocats-policy/values.yaml` file under the service section, set the value of `staticNodePortEnabled` parameter to `true` and enter a valid `nodePort` value for `staticNodePort` parameter.

```
service:
 customExtension:
 labels: {}
 annotations: {}
 type: LoadBalancer
 ports:
 http:
 port: "8080"
 staticNodePortEnabled: false
 staticNodePort: ""
```

### 3.5.4.2 Enabling Static API Node Port

To enable static API node port, in the `ocats-policy/values.yaml` file under the service section, set the value of `staticAPINodePortEnabled` parameter to `true` and enter a valid `nodePort` value for `staticAPINodePortEnabled` parameter.

```
service:
 customExtension:
 labels: {}
 annotations: {}
 type: LoadBalancer
 ports:
 api:
 port: "5001"
 staticNodePortEnabled: false
 staticNodePort: ""
```

### 3.5.4.3 Service Account Requirements

To run Policy-ATS, use the following rules to create a service account:

```
rules:
- apiGroups: ["extensions"]
 resources: ["deployments", "replicasets", "statefulsets"]
 verbs: ["watch", "get", "list", "update"]
- apiGroups: ["apps"]
 resources: ["deployments", "replicasets", "statefulsets"]
 verbs: ["watch", "get", "list", "update"]
- apiGroups: [""]
 resources: ["pods", "services", "secrets", "configmaps"]
 verbs: ["watch", "get", "list", "delete", "update", "create"]
- apiGroups: [""]
```

```
resources: ["pods/log"]
verbs: ["get", "list"]
```

To run oc-ldap ATS, use the following rules to create a service account:

| PolicyRule:                            |                   |                    |       |
|----------------------------------------|-------------------|--------------------|-------|
| Resources                              | Non-Resource URLs | Resource Names     | Verbs |
| -----                                  | -----             | -----              | ----- |
| deployments.apps                       | []                | [rc1-oc-ldap-org1] | [get  |
| list watch create update patch delete] |                   |                    |       |
| deployments.extensions                 | []                | [rc1-oc-ldap-org1] | [get  |
| list watch create update patch delete] |                   |                    |       |
| podsecuritypolicies.policy             | []                | [1org1-rc1]        | [use] |

#### Note

For information about creating service account, see the *Oracle Communications Cloud Native Core, Converged Policy Installation, Upgrade, and Fault Recovery Guide* available on MOS.

### 3.5.4.4 Enabling Aspen Service Mesh

This section provides information on how to enable Aspen service mesh while deploying ATS for CNC Policy. The configurations mentioned in this section are optional and should be performed only if ASM is required.

To enable service mesh for CNC Policy ATS, perform the following steps:

1. In the **service** section of the `values.yaml` file, the **serviceMeshCheck** parameter is set to `false` (default configuration). To enable service mesh, set the value for **serviceMeshCheck** to `true`. The following is a snippet of the service section in the `yaml` file:

```
service:
 customExtension:
 labels: {}
 annotations: {}
 type: LoadBalancer
 ports:
 https:
 port: "8443"
 staticNodePortEnabled: false
 staticNodePort: ""
 http:
 port: "8080"
 staticNodePortEnabled: false
 staticNodePort: ""
 api:
 port: "5001"
 staticNodePortEnabled: false
 staticNodePort: ""
 serviceMeshCheck: true
```

2. If the ASM is not enabled on the global level for the namespace, run the following command to enable it before deploying the ATS:

```
kubectl label --overwrite namespace <namespace_name> istio-
injection=enabled
```

For example:

```
kubectl label --overwrite namespace ocpf istio-injection=enabled
```

3. Uncomment and add the following annotation under the **customExtension** section of the global section in values.yaml file and deploy the ATS Pods:

```
customExtension:
 allResources:
 labels: {}
 annotations: {
 #Enable this section for service-mesh based installation
 traffic.sidecar.istio.io/excludeInboundPorts: "9000",
 traffic.sidecar.istio.io/excludeOutboundPorts: "9000"
 }
```

After making this update in the values.yaml file, make sure that all the ATS and stub pods come up with istio container.

4. For ServerHeader feature, the user needs to perform the following configurations under the envoyFilters for nf1stub and nf2stub in the occnp-servicemesh-config-custom-values-24.1.0.yaml:

```
envoyFilters:
 - name: serverheaderfilter-nf1stub
 labelselector: "app: nf1stub"
 applyTo: NETWORK_FILTER
 filename: envoy.filters.network.http_connection_manager
 operation: MERGE
 typeconfig: type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionM
anager
 configkey: server_header_transformation
 configvalue: PASS_THROUGH
 - name: serverheaderfilter-nf2stub
 labelselector: "app: nf2stub"
 applyTo: NETWORK_FILTER
 filename: envoy.filters.network.http_connection_manager
 operation: MERGE
 typeconfig: type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionM
anager
 configkey: server_header_transformation
 configvalue: PASS_THROUGH

 - name: serverheaderfilter-nf3stub
 labelselector: "app: nf3stub"
 applyTo: NETWORK_FILTER
```

```

 filtername: envoy.filters.network.http_connection_manager
 operation: MERGE
 typeconfig: type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionM
anager
 configkey: server_header_transformation
 configvalue: PASS_THROUGH
 - name: serverheaderfilter-nf32stub
 labelselector: "app: nf32stub"
 applyTo: NETWORK_FILTER
 filtername: envoy.filters.network.http_connection_manager
 operation: MERGE
 typeconfig: type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionM
anager
 configkey: server_header_transformation
 configvalue: PASS_THROUGH

```

5. Perform helm upgrade on the occnp-servicemesh-config release using the modified occnp-servicemesh-config-custom-values-24.1.0.yaml.

```

helm upgrade <helm_release_name_for_servicemesh> -n <namespace>
<servicemesh_charts> -f
 <servicemesh-custom.yaml>

```

#### Example

```

helm upgrade occnp-servicemesh-config occnp-servicemesh-config-24.1.0.tgz -
n <namespace> -f occnp-servicemesh-config-custom-values-24.1.0.yaml

```

6. Configure DNS for Alternate Route service. For more information, see [Post-Installation Steps](#).

### 3.5.4.5 Enabling Persistent Volume

#### ① Note

The steps provided in this section are optional and required only if Persistent Volume needs be to enabled.

ATS supports Persistent storage to retain ATS historical build execution data, test cases and one-time environment variable configurations. With this enhancement, the user can decide whether to use persistent volume based on their resource requirements. By default, the persistent volume feature is not enabled.

To enable persistent storage, perform the following steps:

1. Create a PVC and associate the same to the ATS pod.
  - a. Edit pvc.yaml file.
  - b. Set **PVClaimName** to the PVC created in Step 1.

deployment:

```

customExtension:
 labels: {}
 annotations: {}
PVEEnabled: true
PVClaimName: "ocpcf-pvc-24.1.0"

```

- c. Enter the **storageClassName** to the Storage Class Name.
- d. Set storage to and size of the persistent volume.  
Sample PVC:

```

apiVersion: v1
kind: PersistentVolumeClaim
metadata:
 name: policy-pvc-24.1.0 annotations:
spec:
 storageClassName: standard
 accessModes:
 - ReadWriteOnce
 resources:
 requests:
 storage: 1Gi

```

**Note**

It is recommended to suffix the pvc name with the release version to avoid confusion during the subsequent releases. For example, policy-pvc-24.1.0.

- e. Run the following command to create the PVC.

```
kubectl apply -f <filename> -n <namespace>
```

For example:

```
kubectl apply -f PersistentVolumeClaim.yaml -n ocpcf
```

- f. Once the PVC is created , run the following command to verify that it is bound to the persistent volume and is available.

```
kubectl get pvc -n <namespace used for pvc creation>
```

Sample output:

```

[cloud-user@platform-bastion-1 ocats-policy]$ kubectl get pvc -n ocpcf
NAME STATUS VOLUME CAPACITY ACCESS MODES STORAGECLASS AGE
policy-pvc-24.1.0 Bound pvc-65484045-3805-4064-9fc3-f9eeeaccc8b8 1Gi RWO standard 11s

```

**Note**

Do not proceed further with the next step if there is an issue with the PV creation and contact your administrator to get the PV Created

2. Enable PVC.
  - a. Set the **PVEnabled** flag to **true**.
  - b. Set **PVClaimName** to the PVC created in Step 1.

```
PVEnabled: false
PVClaimName: "policy-pvc-24.1.0"
```

**Note**

Make sure that ATS is deployed before proceeding to the further steps.

3. Copy the `<nf_main_folder>` and `<jenkins jobs>` folders from the tar file to their ATS pod and restart the pod.
  - a. Extract the tar file.

```
tar -xvf ocats-policy-data-24.1.0.tgz
```

- b. Run the following commands to copy the desired folder.

```
kubectl cp ocats-policy-data-24.1.0/ocpcf_tests <namespace>/<pod-name>:/var/lib/jenkins/
```

```
kubectl cp ocats-policy-data-24.1.0/jobs <namespace>/<pod-name>:/var/lib/jenkins/.jenkins/
```

- c. Restart the pod.

```
kubectl delete po <pod-name> -n <namespace>
```

4. Once the Pod is up and running, log in to the Jenkins console and configure the **Discard old Builds** option to configure the number of Jenkins builds, which must be retained in the persistent volume.

**Figure 3-9 Discarding Old Builds**

☒ Discard old builds

Strategy: Log Rotation

Days to keep builds:

if not empty, build records are only kept up to this number of days

Max # of builds to keep: 4

if not empty, only up to this number of build records are kept

Advanced...

**Note**

If **Discard old Builds** is not configured, Persistent Volume can get filled when there are huge number of builds.

For more details on Persistent Volume Storage, see [Persistent Volume for 5G ATS](#).

### 3.5.4.6 Enabling Health Check

This section describes how to enable Health Check for ATS.

To enable Health Check, in the `ocats-policy/values.yaml` file, set the value of `healthcheck` parameter to `true` and enter a valid value to select the environment.

```
webscale: false
healthchecksecretname: "healthchecksecret"
occnehostip: ""
occnehostusername: ""
occnehostpassword: ""
envtype: ""
webscalejumpip: ""
webscalejumpusername: ""
webscalejumpassword: ""
webscaleprojectname: ""
webscalelabserverFQDN: ""
webscalelabserverport: ""
webscalelabserverusername: ""
webscalelabserverpassword: ""
```

To select OCCNE environment, update the values of the following two parameters:

- webscale - Update the value as false
- envtype - T0NDTkU= (i.e envtype=\$(echo -n '**OCCNE**' | base64))
- occnehostip - OCCNE Host IP address
- occnehostusername - OCCNE Host Username
- occnehostpassword - OCCNE Host Password

The following is the sample configuration for OCCNE environment:

```
atsFeatures: ## DO NOT UPDATE this section without Engineering team's
permission
 healthcheck: true # TO enable/disable ATS Health Check.

webscale: false
healthchecksecretname: "healthchecksecret"
occnehostip: "MTAuMTcuMjE5LjY1"
occnehostusername: "dXNlcm5hbWU="
occnehostpassword: "KioqKg=="
envtype: "T0NDTKU="
```

To select WEBSCALE environment, update the values of the following two parameters:

- webscale - Update the value as true
- envtype - V0VCU0NBTEU= (i.e envtype=\$(echo -n 'WEBSCALE' | base64))

After the configurations are done, encrypt below parameters and provide the values as shown in the following snippet:

The following is the sample configuration for WEBSCALE environment:

```
atsFeatures: ## DO NOT UPDATE this section without Engineering team's
permission
 healthcheck: true # TO enable/disable ATS Health Check.

webscale: true
healthchecksecretname: "healthchecksecret"
occnehostip: ""
occnehostusername: ""
occnehostpassword: ""
envtype: "V0VCU0NBTEU="
webscalejumpip: "MTAuNzAuMTE3LjQy"
webscalejumpusername: "dXNlcm5hbWU="
webscalejumppassword: "KioqKg=="
webscaleprojectname: "KioqKg=="
webscalelabserverFQDN: "KioqKg=="
webscalelabserverport: "KioqKg=="
webscalelabserverusername: "KioqKg=="
webscalelabserverpassword: "KioqKg=="
```

#### Note

Once the ATS is deployed with HealthCheck feature enabled or disabled, then it cannot be changed. To change the configuration, you are required to re-install.

## 3.5.5 Deploying ATS and Pods

### 3.5.5.1 Deploying ATS in Kubernetes Cluster

To deploy ATS, perform the following steps:



1. Run the following command using the updated helm charts.

**Note**

Ensure that all the components, ATS, go-Stub, dns-bind, ocampf, and CNC Policy are deployed in the same namespace.

**Using Helm 3** `helm install -name <release_name> ocats-policy-24.1.0.tgz --namespace <namespace_name> -f <values-yaml-file>`

**Example:** `helm install -name ocats ocats-policy-24.1.0.tgz --namespace ocpcf -f ocats-policy/values.yaml`

2. Run the following command to verify ATS deployment:  
`helm ls -n ocpcf`

The sample output is as follows:

| NAME     | REVISION            | UPDATED                      | STATUS |
|----------|---------------------|------------------------------|--------|
| CHART    | APP VERSION         | NAMESPACE                    |        |
| ocats    | 1                   | Mon November 6 14:56:11 2023 |        |
| DEPLOYED | ocats-policy-24.1.0 | 1.0                          | ocpcf  |

The status appears as **DEPLOYED** after the deployment is successful.

### 3.5.5.2 Deploying Stub Pod in Kubernetes Cluster

To deploy Stub Pod in Kubernetes cluster, perform the following steps:

1. Go to the `ocats-policy-tools-24.1.0.0.0` folder and run the following command to extract the `ocstub` tar file content.  
`tar -zxvf ocstub-pkg-24.1.0.0.0.tgz`

The output of this command is:

- `ocstub-py-24.1.0.tgz`
- `ocstub-py-image-24.1.0.tar`

**Note**

To deploy additional stubs required for session retry feature validation:

- `nf11stub, nf12stub` → Alternate FQDN for `nf1stub`
- `nf21stub, nf22stub` → Alternate FQDN for `nf2stub`
- `nf31stub, nf32stub` → Alternate FQDN for `nf3stub`

2. Run the following command in your cluster to load the STUB image.  
`docker load --input ocstub-py-image-24.1.0.tar`
3. Run the following commands to tag and push the STUB image.

```
docker tag ocstub-py:24.1.0 <registry>/ocstub-py:24.1.0
docker push <registry>/ocstub-py:24.1.0
```

4. Run the following command to untar the helm charts, ocstub-py-24.1.0.tgz.  
tar -zxvf ocstub-py-24.1.0.tgz
5. Update the registry name, image name and tag (if required) in the ocstub-py/values.yaml file as required. For this, open the **values.yaml** file and update the image.repository and image.tag parameters.
6. If required, change the apiVersion to apps/v1 in the **ocstub-py/templates/deployment.yaml** file as follows:  
apiVersion: apps/v1

7. Deploy Stub.  
**Using Helm3:**

```
helm install -name <release_name> ocstub-py --set env.NF=<NF> --set
env.LOG_LEVEL=<DEBUG/INFO> --set service.name=<service_name>--set
service.appendReleaseName=false --namespace=<namespace_name> -f <valuesyaml-
file>
```

**Example:**

```
helm install -name nf1stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf1stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf2stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf2stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf3stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf3stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf11stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf11stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf12stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf12stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf21stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf21stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf22stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf22stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf31stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf31stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
helm install -name nf32stub ocstub-py --set env.NF=PCF --set
env.LOG_LEVEL=DEBUG --set service.name=nf32stub --set
service.appendReleaseName=false --namespace=ocpcf -f ocstub-py/values.yaml
```

8. Run the following command to verify stub deployment.  
helm ls -n ocpcf

The sample output is as follows:

| NAME     | REVISION | UPDATED                    | APP VERSION | NAMESPACE |
|----------|----------|----------------------------|-------------|-----------|
| STATUS   | CHART    |                            |             |           |
| nf11stub | 1        | Tue March 14 10:05:59 2024 |             |           |

```

DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf12stub 1 Tue March 14 10:06:00 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf1stub 1 Tue March 14 10:05:57 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf21stub 1 Tue March 14 10:06:01 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf22stub 1 Tue March 14 10:06:02 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf2stub 1 Tue March 14 10:05:58 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf31stub 1 Tue March 14 10:06:03 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf32stub 1 Tue March 14 10:06:11 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf
nf3stub 1 Tue March 14 10:05:59 2024
DEPLOYED ocstub-py-24.1.0 1.0 ocpcf

```

The status changes to **DEPLOYED** after the deployment is successful.  
Similarly, install all other stubs.

9. Run the following command to check the status of Stub deployment.  
helm status <release\_name> -n ocpcf

The sample output is as follows:

| NAME                                 | READY | STATUS | RESTARTS |
|--------------------------------------|-------|--------|----------|
| AGE                                  |       |        |          |
| nf11stub-ocstub-py-66449ddb94-qg2j90 |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf12stub-ocstub-py-6b8575487-l8pxv0  |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf1stub-ocstub-py-5ff485954c-prc2x0  |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf21stub-ocstub-py-56cf5b77fc-x8wkr0 |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf22stub-ocstub-py-547dfdf476-4j2sn0 |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf2stub-ocstub-py-6fb6f786d6-bc9fr0  |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf31stub-ocstub-py-c6c6d5584-5m48z0  |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf32stub-ocstub-py-848dfc7757-q797z0 |       | 1/1    | Running  |
| 19h                                  |       |        |          |
| nf3stub-ocstub-py-6cb769ccd9-4fv9b0  |       | 1/1    | Running  |
| 19h                                  |       |        |          |

A sample output of Policy namespace with Policy and ATS after installation is as follows:

| NAME                                      | READY | STATUS | RESTARTS | AGE  |
|-------------------------------------------|-------|--------|----------|------|
| ocpcf-appinfo-6c74cccd47-zsbb20           |       | 1/1    | Running  | 155m |
| ocpcf-oc-binding-77fbb9b79c-jv7kd0        |       | 1/1    | Running  | 155m |
| ocpcf-oc-diam-connector-6c6fd868bd-4zfrn0 |       | 1/1    | Running  | 155m |

|                                                      |     |         |
|------------------------------------------------------|-----|---------|
| ocpcf-oc-diam-gateway-0                              | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-oc-oc-stub-595bb858d4-smzj8                    | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-ocats-ocats-policy-667d8cf78-b8bc8             | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-occnp-alternate-route-75455c858d-f6qs8         | 1/1 | Running |
| 0 146m                                               |     |         |
| ocpcf-occnp-alternate-route-75455c858d-sqvlq         | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-occnp-chf-connector-6b8b8bfcd6-jjch6           | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-occnp-config-server-77bd99f96-mpscn            | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-occnp-egress-gateway-59c4b784cc-6dx4w          | 1/1 | Running |
| 0 16m                                                |     |         |
| ocpcf-occnp-ingress-gateway-75c47c57bc-pljtc         | 1/1 | Running |
| 0 39m                                                |     |         |
| ocpcf-occnp-nrf-client-nfdiscovery-74b854956b-s6blq  | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-occnp-nrf-client-nfmanagement-76cb55b8b8-tdjkj | 1/1 | Running |
| 0 49m                                                |     |         |
| ocpcf-occnp-udr-connector-75ffb9db9b-7xz9v           | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocdns-ocdns-bind-57fbcd95dc-h4dtl              | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-ocpm-audit-service-5cc46665c4-j6vhh            | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocpm-cm-service-7795bb4c6c-446rb               | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocpm-policyds-75cbc9fc9d-7lbl5                 | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocpm-pre-59b94d979-jzkv4                       | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocpm-pre-test-84d9c89dd8-fqlpg                 | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-ocpm-queryservice-94895bf88-bhwcc              | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-pcf-amservice-56cddb75c9-ph7tt                 | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-pcf-smsservice-64b899d766-jfhjm                | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-pcf-ueservice-7c6bd7ccc9-mrnxn                 | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-pcrf-core-7594dbb7f8-z95vt                     | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-performance-689dd556b-7vblc                    | 1/1 | Running |
| 0 155m                                               |     |         |
| ocpcf-fnf11stub-5bb6b4f95d-v6fbb                     | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-fnf12stub-59fb974f5d-2qr42                     | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-fnf1stub-5bdf545fcb-zgbjb                      | 1/1 | Running |
| 0 147m                                               |     |         |
| ocpcf-fnf21stub-ff6db9d86-5hvjb                      | 1/1 | Running |
| 0 147m                                               |     |         |

|                                          |     |         |
|------------------------------------------|-----|---------|
| ocpcfntf22stub-794456fd66-sxq8q          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcfntf2stub-656755dc46-hnr8m           | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcfntf31stub-68c6596b6-jdsgj           | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcfntf32stub-f49b57d86-rklc8           | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcfntf3stub-6c4c697648-lj6q7           | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-ocpm-ldap-gateway-5fd489b8fd-52dqn | 1/1 | Running |
| 0 147m                                   |     |         |

### 3.5.5.3 Deploying DNS Stub in Kubernetes Cluster

#### Note

Ensure there is sufficient resource requests and limit is configured for DNS Stub. Set the resource request and limit values in the **resources** section in the **values.yaml** file as follows:

```
resources: {}
We usually recommend not to specify default resources and to leave
this as a conscious
choice for the user. This also increases chances charts run on
environments with little
resources, such as Minikube. If you do want to specify resources,
uncomment the following
lines, adjust them as necessary, and remove the curly braces after
'resources:'. # limits:
cpu: 1000m
memory: 1024Mi
requests:
cpu: 500m
memory: 500Mi
```

To deploy DNS stub in Kubernetes cluster, perform the following steps:

1. Go to the **ocats-policy-tools-24.1.0.0.0** folder and run the following command:  
tar -zxvf ocdns-pkg-24.1.0.0.0.tgz

The output of this command is:

```
[cloud-user@platform-bastion-1 ocdns-pkg-24.1.0.0.0]$ ls -ltrh
total 211M
-rw-----. 1 cloud-user cloud-user 211M Mar 14 14:49 ocdns-bind-
image-24.1.0.tar
-rw-r--r--. 1 cloud-user cloud-user 2.9K Mar 14 14:49 ocdns-bind-24.1.0.tgz
```

2. Run the following command in your cluster to load the DNS Stub image:  
docker load --input ocdns-bind-image-24.1.0.tar

3. Run the following commands to tag and push the DNS stub to the registry:

```
docker tag ocdns-bind:24.1.0 localhost:5000/ocdns-bind:24.1.0
docker push localhost:5000/ocdns-bind:24.1.0
```

4. Run the following command to untar the helm charts (ocdns-bind-24.1.0.tgz):  
tar -zxvf ocdns-bind-24.1.0.tgz
5. Update the registry name, image name and tag (if required) in the ocdns-bind/values.yaml file as required. Open the **values.yaml** file and update the image.repository and image.tag parameters.
6. Run the following command to install DNS Stub:

```
helm :
[cloud-user@platform-bastion-1 ocdns-bind]$ helm install -name ocdns
ocdns-bind-24.1.0.tgz --namespace ocpf -f ocdns-bind/values.yaml
```

7. Run the following command to capture the cluster name of the pcf deployment, namespace where nfstubs are deployed and cluster IP of DNS Stub.  
kubectl get svc -n ocpf | grep dns

| NAME          | TYPE      | CLUSTER-IP   | EXTERNAL-IP |
|---------------|-----------|--------------|-------------|
| ocdns         | ClusterIP | 10.233.11.45 | <none>      |
| UDP, 6236/TCP |           | 19h          | 53/         |

#### Note

This information is required to configure DNS stub.

**Figure 3-10 Cluster Name**

```
kubectl -n kube-system get configmap kubeadm-config -o yaml | grep
clusterName
clusterName: platform
```

### 3.5.5.4 Deploying AMF Stub in Kubernetes Cluster

To deploy OCAMF stub in Kubernetes cluster:

1. Go to the **ocats-policy-tools-24.1.0.0.0** folder and run the following command:  
tar -zxvf ocamf-pkg-24.1.0.0.0.tgz

The output of this command is:

```
[cloud-user@platform-bastion-1 ocamf-pkg-24.1.0.0.0]$ ls -ltrh
total 211M
-rw-----. 1 cloud-user cloud-user 211M Mar 14 14:49 ocamf-stub-
image-24.1.0.tar
-rw-r--r--. 1 cloud-user cloud-user 2.9K Mar 14 14:49 ocamf-stub-24.1.0.tgz
```

2. Run the following command in your cluster to load the AMF Stub image:  
docker load --input ocamf-stub-image-24.1.0.tar

3. Run the following command to tag and push the DNS stub to the registry:

```
docker tag ocamf-stub:24.1.0 localhost:5000/ocamf-stub:24.1.0
docker push localhost:5000/ocamf-stub:24.1.0
```

4. Run the following command to untar the helm charts (ocamf-stub-24.1.0.tgz):  
tar -zxvf ocamf-stub-24.1.0.tgz
5. Update the registry name, image name and tag (if required) in the ocamf-stub/values.yaml file as required. Open the **values.yaml** file and update the registry name, image name, and tag (if required) in the file.
6. Run the following command to install AMF Stub:  
**Using Helm3:**

```
[cloud-user@platform-bastion-1 ocamf-stub]$ helm3 install-name ocamf2
ocamf-stub-24.1.0.tgz --setservice.name=ocamf2 --namespace ocpf -f ocamf-
stub/values.yaml
```

The status changes to **RUNNING** after the deployment is successful.

The following is a sample output for a successful deployment:

```
ocamf2-ocamf-ocamf-stub-79c8fbd6f7-qp5cl 1/1 Running
0 5h47m
```

### 3.5.5.5 Deploying LDAP Stub in Kubernetes Cluster

To deploy oc-ldap stub in the Kubernetes cluster, perform the following steps:

1. Go to the ocats-policy-tools-24.1.0.0.0 folder and run the following command:  
tar -zxvf oc-ldap-org1-pkg-24.1.0.0.0

The following is the output:

```
[cloud-user@platform-bastion-1 oc-ldap-org1-pkg-24.1.0.0.0]$ ls -ltrh
total 211M
-rw-----. 1 cloud-user cloud-user 211M Mar 14 14:49 oc-ldap-org1-
image-24.1.0.tar
-rw-r--r--. 1 cloud-user cloud-user 2.9K Mar 14 14:49 oc-ldap-
org1-24.1.0.tgz
```

2. Run the following command in your cluster to load the LDAP Stub image:  
docker load --input oc-ldap-org1-image-24.1.0.tar
3. Run the following command to tag and push the DNS stub to the registry:  
  
docker tag oc-ldap-org1:24.1.0 localhost:5000/oc-ldap-org1:24.1.0  
docker push localhost:5000/oc-ldap-org1:24.1.0
4. Run the following command to untar the helm charts (oc-ldap-org1-24.1.0.tgz):  
tar -zxvf oc-ldap-org1-24.1.0.tgz
5. Update the registry name, image name and tag (if required) in the oc-ldap-org1/values.yaml file as required. Open the **values.yaml** file and update the registry name, image name, and tag (if required) in the file.

- Run the following command to install AMF Stub:  
**Using Helm3:**

```
[cloud-user@platform-bastion-1 oc-ldap-org1]$ helm upgrade --install --
namespace ocpcf --set image.repository=localhost:5000/ocnp/oc-ldap-org1
oc-ldap-org1 oc-ldap-org1-24.1.0.tgz
```

The status changes to **RUNNING** after the deployment is successful.

The following is a sample output for a successful deployment:

```
ocpcf-oc-ldap-org1-7b9d957bc6-ngtrl 1/1 Running
0 5h47m
```

**Note**

oc-ldap-org1-secret of the OC-LDAP Stub is being created by Helm chart that comes with ATS package.

### 3.5.5.6 Deploying ocdiam Simulator in Kubernetes Cluster

To deploy ocdiam Simulator in the Kubernetes cluster, perform the following steps:

- Go to the ocats-policy-tools-24.1.0.0.0 folder and run the following command:

```
tar -zxvf ocdiam-pkg-24.1.0.0.0
```

The following is the output:

```
[cloud-user@platform-bastion-1 ocdiam-pkg-24.1.0.0.0]$ ls -ltrh
total 908M
-rw-----. 1 cloud-user cloud-user 908M Mar 14 14:49 ocdiam-sim-
image-24.1.0.tar
-rw-r--r--. 1 cloud-user cloud-user 3.8K Mar 14 14:49 ocdiam-sim-24.1.0.tgz
```

- Run the following command in your cluster to load the Diameter Simulator Image using the command :

```
docker load --input ocdiam-sim-image-24.1.0.tar
```

- Run the following command to tag and push the DNS stub to the registry:

```
docker tag ocdiam-sim:24.1.0 localhost:5000/ocdiam-sim:24.1.0
docker push localhost:5000/ocdiam-sim:24.1.0
```

- Run the following command to untar the helm charts (ocdiam-sim-24.1.0.tgz):

```
tar -zxvf ocdiam-sim-24.1.0.tgz
```

- Update the registry name, image name and tag (if required) in the ocdiam-sim/values.yaml file as required. Open the **values.yaml** file and update the registry name, image name, and tag (if required) in the file.



6. Run the following command to install Diameter Simulator:  
**Using Helm3:**

```
[cloud-user@platform-bastion-1 ocdiam-sim]$ helm3 install-name ocdiam-sim
ocdiam-sim --namespace ocpcf -f ocdiam-sim/values.yaml
```

The status changes to **RUNNING** after the deployment is successful.

The following is a sample output for a successful deployment:

```
ocdiam-sim-69968444b6-fg6ks 1/1 Running 0 5h47m
```

Sample of Policy namespace with Policy and ATS after installation:

```
[cloud-user@platform-bastion-1 ocstub-pkg-24.1.0.0.0]$ kubectl get po -n ocpcf
NAME READY STATUS
RESTARTS AGE
ocpcf-appinfo-6c74cccd47-zsbb2 1/1 Running
0 155m
ocpcf-oc-binding-77fbb9b79c-jv7kd 1/1 Running
0 155m
ocpcf-oc-diam-connector-6c6fd868bd-4zfrn 1/1 Running
0 155m
ocpcf-oc-diam-gateway-0 1/1 Running
0 147m
ocamf2-ocamf-stub-595bb858d4-smzj8 1/1 Running
0 147m
ocpcf-ocats-ocats-policy-667d8cf78-b8bc8 1/1 Running
0 147m
ocpcf-occnp-alternate-route-75455c858d-f6qs8 1/1 Running
0 146m
ocpcf-occnp-chf-connector-6b8b8bfcd6-jjch6 1/1 Running
0 155m
ocpcf-occnp-config-server-77bd99f96-mpscn 1/1 Running
0 155m
ocpcf-occnp-egress-gateway-59c4b784cc-6dx4w 1/1 Running
0 16m
ocpcf-occnp-ingress-gateway-75c47c57bc-pljtc 1/1 Running
0 39m
ocpcf-occnp-nrf-client-nfdiscovery-74b854956b-s6blq 1/1 Running
0 155m
ocpcf-occnp-nrf-client-nfmanagement-76cb55b8b8-tdjkj 1/1 Running
0 49m
ocpcf-occnp-udr-connector-75ffb9db9b-7xz9v 1/1 Running
0 155m
ocpcf-ocdns-ocdns-bind-57fbcd95dc-h4dtl 1/1 Running
0 147m
ocpcf-ocpm-audit-service-5cc46665c4-j6vhh 1/1 Running
0 155m
ocpcf-ocpm-cm-service-7795bb4c6c-446rb 1/1 Running
0 155m
ocpcf-ocpm-policyds-75cbc9fc9d-7lbl5 1/1 Running
0 155m
ocpcf-ocpm-pre-59b94d979-jzkv4 1/1 Running
0 155m
```

|                                          |     |         |
|------------------------------------------|-----|---------|
| ocpcf-ocpm-pre-test-84d9c89dd8-fqlpg     | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-ocpm-queryservice-94895bf88-bhwcc  | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-pcf-amservice-56cddb75c9-ph7tt     |     |         |
| ocpcf-pcf-smsservice-64b899d766-jfhjm    | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-pcf-ueservice-7c6bd7ccc9-mrnxn     | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-pcrf-core-7594dbb7f8-z95vt         | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-performance-689dd556b-7vblc        | 1/1 | Running |
| 0 155m                                   |     |         |
| ocpcf-fnf11stub-5bb6b4f95d-v6fbb         | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf12stub-59fb974f5d-2qr42         | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf1stub-5bdf545fcb-zgbjb          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf21stub-ff6db9d86-5hvj6          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf22stub-794456fd66-sxq8q         | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf2stub-656755dc46-hnr8m          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf31stub-68c6596b6-jdsgj          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf32stub-f49b57d86-rklc8          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-fnf3stub-6c4c697648-lj6q7          | 1/1 | Running |
| 0 147m                                   |     |         |
| ocpcf-ocpm-ldap-gateway-5fd489b8fd-52dqn | 1/1 | Running |
| 0 147m                                   |     |         |
| ocdiam-sim-69968444b6                    | 1/1 | Running |
| 0 147m                                   |     |         |

## 3.5.6 Post-Installation Steps

This section describes the post-installation steps for Policy.

### Alternate Route Service Configurations

To edit the Alternate Route Service deployment file (ocpcf-ocnp-alternate-route) that points to DNS Stub, perform the following steps:

1. Run the following command to get searches information from dns-bind pod to enable communication between Alternate Route and dns-bind service:

```
kubectl exec -it <dns-bind pod> -n <NAMESPACE> -- /bin/bash -c 'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
```

The following output is displayed after running the command:

**Figure 3-11 Sample Output**

```
[flanders@bastion-1 ~]$ kubectl exec -it ocdns-stub-ocdns-bind-7467cf56c8-5d9g2 -n flanders-ns -- /bin/bash -c 'cat /etc/resolv.conf | grep search | tr
'\n' | grep -v 'search'
flanders-ns.svc.inferno.lab.us.oracle.com
svc.inferno.lab.us.oracle.com
inferno.lab.us.oracle.com
```

By default alternate service will point to CoreDNS and you will see following settings in deployment file:

**Figure 3-12 Alternate Route Service Deployment File**

```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

2. Run the following command to edit the deployment file and add the following content in alternate service to query DNS stub:

```
$kubectl edit deployment ocpcf-occpn-alternate-route -n ocpcf
```

- a. Add the IP Address of the nameserver that you have recorded after installing the DNS stub (cluster IP Address of DNS Stub).
- b. Add the search information one by one which you recorded earlier.
- c. Set dnsPolicy to "None".

```
dnsConfig:
 nameservers:
 - 10.233.33.169 // cluster IP of DNS Stub
 searches:
 - ocpcf.svc.occne15-ocpcf-ats
 - svc.occne15-ocpcf-ats
 - occne15-ocpcf-ats
dnsPolicy: None
```

For example:

**Figure 3-13 Example**

```
terminationMessagePath: /dev/termination-log
terminationMessagePolicy: File
dnsConfig:
 nameservers:
 - 10.233.31.245
 searches:
 - flanders-ns.svc.inferno.lab.us.oracle.com
 - svc.inferno.lab.us.oracle.com
 - inferno.lab.us.oracle.com
dnsPolicy: None
restartPolicy: Always
schedulerName: default-scheduler
```

### NRF client configmap

In the `-application-config` configmap, configure the following parameters with the respective values:

- `primaryNrfApiRoot=nf1stub.<namespace_gostubs_are_deployed_in>.svc:8080`  
**Example:** `primaryNrfApiRoot=nf1stub.ocats.svc:8080`
- `secondaryNrfApiRoot=nf1stub.ocats.svc:8080` (Remove the `secondaryNrfApiRoot`)
- `nrfClientSubscribeTypes=UDR, CHF, NWDAF`
- `supportedDataSetId=POLICY` (Remove the `supportedDataSetId`)

#### Note

Configure these values at the time of Policy deployment.

#### Note

To get all configmaps in your namespace, execute the following command:  
`kubectl get configmaps -n <Policy_namespace>`

### Persistent Volume (Optional)

If persistent volume is used, follow the post-installation steps provided in the [Persistent Volume for 5G ATS](#) section.

## 3.6 Installing ATS for SCP

This section describes Automated Testing Suite (ATS) installation procedures for Service Communication Proxy (SCP) in a cloud native environment using Continuous Delivery Control Server (CDCS) or Command Line Interface (CLI) procedures. For more information about CDCS, see the following documents:

- *Oracle Communications CD Control Server Installation and Upgrade Guide*
- *Oracle Communications CD Control Server User Guide*

You must perform ATS installation procedures for SCP in the same sequence as outlined in the following sections.

### 3.6.1 Prerequisites

To run SCP test cases, the following prerequisites are required.

#### 3.6.1.1 Software Requirements

This section lists the software that must be installed before installing ATS.

**Table 3-18 Preinstalled Software**

| Software   | Version                |
|------------|------------------------|
| Kubernetes | 1.28.x, 1.27.x, 1.26.x |
| Helm       | 3.13.2                 |
| Podman     | 4.4.1                  |

To check the versions of the preinstalled software in the cloud native environment, run the following commands:

```
kubectl version
```

```
helm version
```

```
podman version
```

### 3.6.1.2 Environment Setup Requirements

This section describes the requirements for the client machine, that is, the machine used by the user to run deployment commands.

The client machine should have:

- Helm repository configured.
- Network access to the Helm repository and Docker image repository.
- Network access to the Kubernetes cluster.
- Required environment settings to run `kubectl`, `docker`, and `podman` commands. The environment should have privileges to create a namespace in the Kubernetes cluster.
- Helm client is installed with the push plugin. Configure the environment in such a manner that the `helm install` command deploys the software in the Kubernetes cluster.

### 3.6.1.3 Resource Requirements

This section describes ATS resource requirements for SCP.

#### Overview - Total Number of Resources

The following table describes the total resource usage by different resource types:

- SCP SUT
- cnDB Tier
- ATS

**Table 3-19 SCP - Total Number of Resources**

| Resource Name   | CPU | Memory (GB) | Storage (GB) |
|-----------------|-----|-------------|--------------|
| SCP SUT Totals  | 56  | 61          | 0            |
| cnDBTier Totals | 29  | 65          | 235          |

**Table 3-19 (Cont.) SCP - Total Number of Resources**

| Resource Name              | CPU        | Memory (GB) | Storage (GB) |
|----------------------------|------------|-------------|--------------|
| ATS Totals                 | 100        | 106         | 4            |
| <b>Grand Total SCP ATS</b> | <b>185</b> | <b>232</b>  | <b>263</b>   |

**SCP Pods Resource Requirements**

This section describes the resources required to deploy SCP ATS.

**Table 3-20 SCP Pods Resource Requirements**

| Microservice              | CPUs<br>Require<br>d per<br>Pod | Memory<br>Require<br>d per<br>Pod<br>(GB) | Storage<br>PVC<br>Require<br>d per<br>Pod<br>(GB) | # Replicas<br>(regular<br>deploy<br>ment) | # Replicas<br>(ATS<br>deploy<br>ment) | CPUs<br>Require<br>d (ATS)<br>- Total | Memory<br>Require<br>d (ATS)<br>- Total<br>(GB) | Storage<br>PVC<br>Require<br>d - Total<br>(GB) |
|---------------------------|---------------------------------|-------------------------------------------|---------------------------------------------------|-------------------------------------------|---------------------------------------|---------------------------------------|-------------------------------------------------|------------------------------------------------|
| SCP Pods                  |                                 |                                           |                                                   |                                           |                                       |                                       |                                                 |                                                |
| scpc-subscription         | 1                               | 1                                         | 0                                                 | 1                                         | 1                                     | 1                                     | 1                                               | 0                                              |
| scpc-notification         | 4                               | 4                                         | 0                                                 | 1                                         | 1                                     | 4                                     | 4                                               | 0                                              |
| scpc-audit                | 3                               | 4                                         | 0                                                 | 1                                         | 1                                     | 3                                     | 4                                               | 0                                              |
| scpc-configuration        | 2                               | 2                                         | 0                                                 | 1                                         | 1                                     | 2                                     | 2                                               | 0                                              |
| scp-worker                | 4                               | 8                                         | 0                                                 | 2                                         | 1                                     | 4                                     | 8                                               | 0                                              |
| scpc-alternate-resolution | 2                               | 2                                         | 0                                                 | 1                                         | 1                                     | 2                                     | 2                                               | 0                                              |
| scp-nrfproxy              | 8                               | 8                                         | 0                                                 | 2                                         | 1                                     | 8                                     | 8                                               | 0                                              |
| scp-cache                 | 8                               | 8                                         | 0                                                 | 3                                         | 1                                     | 8                                     | 8                                               | 0                                              |
| scp-mediation             | 8                               | 8                                         | 0                                                 | 2                                         | 1                                     | 8                                     | 8                                               | 0                                              |
| scp-load-manager          | 8                               | 8                                         | 0                                                 | 2                                         | 1                                     | 8                                     | 8                                               | 0                                              |
| scp-oauth-nrfproxy        | 8                               | 8                                         | 0                                                 | 2                                         | 1                                     | 8                                     | 8                                               | 0                                              |
| <b>SCP SUT Totals</b>     |                                 |                                           |                                                   |                                           |                                       | <b>56</b>                             | <b>61</b>                                       | <b>0</b>                                       |

**ATS Resource Requirements for SCP**

This section describes the ATS resources required to deploy SCP-ATS.

**Table 3-21 ATS Resource Requirements**

| Microservice                     | CPUs<br>Require<br>d per<br>Pod | Memory<br>Require<br>d per<br>Pod<br>(GB) | Storage<br>PVC<br>Require<br>d per<br>Pod<br>(GB) | # Replicas<br>(regular<br>deploy<br>ment) | # Replicas<br>(ATS<br>deploy<br>ment) | CPUs<br>Require<br>d - Total | Memory<br>Require<br>d - Total<br>(GB) | Storage<br>PVC<br>Require<br>d - Total<br>(GB) |
|----------------------------------|---------------------------------|-------------------------------------------|---------------------------------------------------|-------------------------------------------|---------------------------------------|------------------------------|----------------------------------------|------------------------------------------------|
| ATS Behave                       | 6                               | 12                                        | 4                                                 | -                                         | 1                                     | 6                            | 12                                     | 4                                              |
| ATS pystub                       | 1                               | 1                                         | -                                                 | -                                         | 90                                    | 91                           | 91                                     | 0                                              |
| DNS Stub                         | 1                               | 1                                         | -                                                 | -                                         | 1                                     | 1                            | 1                                      | 0                                              |
| Global Egress Rate Limiting Stub | 1                               | 1                                         | -                                                 | -                                         | 1                                     | 1                            | 1                                      | 0                                              |

Table 3-21 (Cont.) ATS Resource Requirements

| Microservice       | CPUs<br>Require<br>d per<br>Pod | Memory<br>Require<br>d per<br>Pod<br>(GB) | Storage<br>PVC<br>Require<br>d per<br>Pod<br>(GB) | #<br>Replica<br>s<br>(regular<br>deploy<br>ment) | #<br>Replica<br>s (ATS<br>deploy<br>ment) | CPUs<br>Require<br>d - Total | Memory<br>Require<br>d - Total<br>(GB) | Storage<br>PVC<br>Require<br>d - Total<br>(GB) |
|--------------------|---------------------------------|-------------------------------------------|---------------------------------------------------|--------------------------------------------------|-------------------------------------------|------------------------------|----------------------------------------|------------------------------------------------|
| ATS DD Client stub | 1                               | 1                                         | -                                                 | -                                                | 1                                         | 1                            | 1                                      | 0                                              |
| <b>ATS Totals</b>  |                                 |                                           |                                                   |                                                  |                                           | <b>100</b>                   | <b>106</b>                             | <b>0</b>                                       |

### 3.6.1.4 Downloading the ATS Package

This section provides information about how to download the ATS package using Command Line Interface (CLI). To download ATS package using CDCS, see *Oracle Communications CD Control Server User Guide*.

To locate and download the ATS Image from MOS:

1. Log in to [My Oracle Support](#) using the appropriate login credentials.
2. Click the **Patches & Updates** tab.
3. In the Patch Search section, click **Product or Family (Advanced)**.
4. Enter Oracle Communications Cloud Native Core - 5G in the **Product** field.
5. Select Oracle Communications Cloud Native Core Service Communication Proxy <release\_number> from the **Release** drop-down list.
6. Click **Search**.  
The Patch Advanced Search Results list appears.
7. Select the required ATS patch from the list.  
The Patch Details window appears.
8. Click **Download**.  
The File Download window appears.
9. Click the `ocats_ocscp_csar_24_1_1_0_0.pkg` file to download the CNC SCP ATS package file.  
The `ocats_ocscp_csar_24_1_1_0_0.pkg` package contains the following files:

```
ocats_ocscp_csar_24_1_1_0_0.zip
mcafee-gen-ats-csar-24.1.1.log
```

#### Note

The above zip file contains all the images and custom values required for 24.1.1 release of OCATS-OCSCP.

Unzip the `ocats_ocscp_csar_24_1_1_0_0.zip` file to get the following files and folders:

```
.
|-- Definitions
| |-- ocats_ocscp_ats_tests.yaml
```

```

| |-- ocats_ocscp_cne_compatibility.yaml
| |-- ocats_ocscp.yaml
|-- Files
| |-- ChangeLog.txt
| |-- Helm
| | |-- ocats-ocscp-24.1.1.tgz
| |-- Licenses
| |-- ocats-ddclientstub-24.1.1.tar
| |-- ocats-dnsstub-24.1.1.tar (Docker Image)
| |-- ocats-pystub-24.1.1.tar (Docker Image)
| |-- ocats-scp-24.1.1.tar (Docker Image)
| |-- ocats-scpglbratelimitstub-24.1.1.tar (Docker Image)
| |-- Oracle.cert
| |-- Tests
|-- ocats_ocscp_csar_24_1_1_0_0.zip
|-- ocats_ocscp.mf
|-- Scripts
| |-- ocats_ocscp_custom_serviceaccount_24.1.1.yaml (Template to create
custom service account)
| |-- ocats_ocscp_tests_jenkinsjobs_24.1.1.tgz. (ocscp_tests and jenkins
jobs folder to be copied if persistent volume is deployed)
| |-- ocats_ocscp_values_24.1.1.yaml (Custom values file for installation)
|-- TOSCA-Metadata
|-- TOSCA.meta

```

10. Copy the umbrella Helm chart `ocats-ocscp-24.1.1.tgz` file from the **Files** folder to Kubernetes cluster where you want to deploy ATS.  
The following table describes ATS parameters in the `ocats_ocscp_values_24.1.1.yaml` file:

**Table 3-22 ATS Parameters of the YAML File**

| Parameter                                    | Default Value | Possible Values | Description                                                                                                                                                                                                                                                                                                             |
|----------------------------------------------|---------------|-----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>ocatsdnsstubService</code>             | true          | true, false     | Set it to true or false depending on the need of user.<br>Setting these values to true to deploy <code>ocats-dnsstub</code> , <code>ocats-scp stub</code> , <code>ocats-scpglbratelimitstub</code> , <code>ocats-ddclientStub</code> , <code>ocats-pystubs</code> ( <code>ausf1</code> , <code>udm3</code> etc.)        |
| <code>ocatsscpService</code>                 | true          | true, false     | Set it to true or false depending on the requirement.<br>Setting these values to true to deploy <code>ocats-dnsstub</code> , <code>ocats-scp stub</code> , <code>ocats-scpglbratelimitstub</code> , <code>ocats-ddclientStub</code> , <code>ocats-pystubs</code> ( <code>ausf1</code> , <code>udm3</code> , and so on)  |
| <code>ocatsscpglbratelimitstubService</code> | true          | true, false     | Set it to true or false depending on the requirement.<br>Setting these values to true to deploy <code>ocats-dnsstub</code> , <code>ocats-scp stub</code> , <code>ocats-scpglbratelimitstub</code> , <code>ocats-ddclientStub</code> , <code>ocats-pystubs</code> ( <code>ausf1</code> , <code>udm3</code> , and so on.) |



Table 3-22 (Cont.) ATS Parameters of the YAML File

| Parameter                                                 | Default Value                           | Possible Values                                                                                  | Description                                                                                                                                                                                                                                                                  |
|-----------------------------------------------------------|-----------------------------------------|--------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocatsddclientstubService                                  | true                                    | true, false                                                                                      | Set it to true or false depending on the requirement.<br>Setting these values to true to deploy ocats-dnsstub, ocats-scp stub, ocats-scpglbratelimitstub, ocats-ddclientStub, ocats-pystubs (ausf1, udm3, and so on.)                                                        |
| ausf1Stubs                                                | true                                    | true, false                                                                                      | Set it to true or false depending on the requirement.<br>Setting these values to true to deploy all the ocats-pystubs (ausf1, udm3, chf1, scp3, and so on.)                                                                                                                  |
| sutScpIPFamiliesforATSRun                                 | [IPv4]                                  | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                                         | The parameter sutScpIPFamiliesforATSRun is used to specify the IP families that ATS will consider when running test cases.<br><b>Note:</b> If any value other than those specified is provided, ATS will proceed with the assumption that the deployment supports IPv4 only. |
| traffic.sidecar.istio.io/excludeOutboundPorts             | 8091                                    | -                                                                                                | This annotation under lbDeployment is required for fetching metrics from soothsayer pods, which are required for a few FT's of ATS. When ATS runs in an ASPEN MESH environment, do not change this port.                                                                     |
| traffic.sidecar.istio.io/excludeInboundPorts              | 8080                                    | -                                                                                                | This annotation under lbDeployment is required for fetching metrics from soothsayer pods, which are required for a few FT's of ATS. When ATS runs in an ASPEN MESH environment, do not change this port.                                                                     |
| tokenConsumptionIntervalInMillis                          | 50                                      | Count in milliseconds                                                                            | Token Consumption Simulation Parameters.                                                                                                                                                                                                                                     |
| scpStubNflInstanceid                                      | 2faf1bbc-6e4a-4454-a507-a14ef8e1bc22    | nflInstanceID of SCP                                                                             | NF Instance ID of SCP.                                                                                                                                                                                                                                                       |
| rateDataReporterStartingOffset                            | 35                                      | Value in milliseconds                                                                            | -                                                                                                                                                                                                                                                                            |
| coherence.clusterName                                     | scpstub-coherence-cluster               | Local Coherence Cluster name, size not more than 66 characters                                   | Local Coherence Cluster name size must not be more than 66 characters.                                                                                                                                                                                                       |
| coherence.clusterName.federation.remoteScpOne.fqdnOrIp    | ocscp-scp-cache.scpvc.svc.cluster.local | -                                                                                                | FQDN or IP of federation configuration.                                                                                                                                                                                                                                      |
| coherence.clusterName.federation.remoteScpOne.port        | 30001                                   | -                                                                                                | Port number of federation configuration.                                                                                                                                                                                                                                     |
| coherence.clusterName.federation.remoteScpOne.clusterName | scp-coherence-cluster                   | Ensure that the cluster name is unique among all participants. Size not more than 66 characters. | remoteScpOne Coherence Cluster name.                                                                                                                                                                                                                                         |

Table 3-22 (Cont.) ATS Parameters of the YAML File

| Parameter                                                                                                                                                                                                                                                           | Default Value                                           | Possible Values                                                                     | Description                                                                                                                                                                                                                                                                                                                                                 |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| servicIpFamilyPolicy.ocatsdnsstubService<br>servicIpFamilyPolicy.ocatsscpService<br>servicIpFamilyPolicy.ocatsscpglbratelimitstubService<br>servicIpFamilyPolicy.ocatsddclientstubService<br>servicIpFamilyPolicy.ausf1Service<br>servicIpFamilyPolicy.ausf2Service | SingleStack                                             | SingleStack, PreferDualStack, RequireDualStack                                      | IpFamilyPolicy of ocatsdnsstubService, ocatsscpService, ocatsscpglbratelimitstubService, ocatsddclientstubService and pyStubs.<br><b>Note:</b> PreferDualStack and RequireDualStack values can only be used if the setup is dual stack.                                                                                                                     |
| servicIpFamilies.ocatsdnsstubService<br>servicIpFamilies.ocatsscpService<br>servicIpFamilies.ocatsscpglbratelimitstubService<br>servicIpFamilies.ocatsddclientstubService<br>servicIpFamilies.ausf1Service<br>servicIpFamilies.ausf2Service                         | [IPv4]                                                  | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                            | IpFamilies of ocatsdnsstubService, ocatsscpService, ocatsscpglbratelimitstubService, ocatsddclientstubService and pyStubs.<br><b>Note:</b> If servicIpFamilyPolicy is SingleStack, then servicIpFamilies can be [IPv4] or [IPv6]. If servicIpFamilyPolicy is PreferDualStack or RequireDualStack, then servicIpFamilies can be [IPv4,IPv6] or [IPv6, IPv4]. |
| PVEnabled                                                                                                                                                                                                                                                           | false                                                   | true, false                                                                         | Enabling persistent volume.                                                                                                                                                                                                                                                                                                                                 |
| PVClaimName                                                                                                                                                                                                                                                         | false                                                   | Name                                                                                | Persistent volume claim name.                                                                                                                                                                                                                                                                                                                               |
| atsGuiTLSEnabled                                                                                                                                                                                                                                                    | false                                                   | true, false                                                                         | Enabling Https in Jenkins GUI.                                                                                                                                                                                                                                                                                                                              |
| atsCommunicationTLSEnabled                                                                                                                                                                                                                                          | false                                                   | true, false                                                                         | Enabling Https communication.                                                                                                                                                                                                                                                                                                                               |
| ocats-scp.image.repository                                                                                                                                                                                                                                          | <docker-registryIP:docker-registryport>/ocats/ocats-scp | <Image repository name:port>/ocats/ocats-scp                                        | Image repository and port of ocats scp image.                                                                                                                                                                                                                                                                                                               |
| ocats-scp.image.tag                                                                                                                                                                                                                                                 | helm-tag                                                | Value of tag to be deployed                                                         | Tag of ocats-scp image.                                                                                                                                                                                                                                                                                                                                     |
| ocats-scp.image.pullPolicy                                                                                                                                                                                                                                          | Always                                                  | Always, IfNotPresent, Never                                                         | Tag of ocats-scp image.                                                                                                                                                                                                                                                                                                                                     |
| ocats-scp.replicaCount                                                                                                                                                                                                                                              | 1                                                       | Positive integers                                                                   | Replica count of ocats-scp stub.                                                                                                                                                                                                                                                                                                                            |
| ocats-scp.resources.limits.cpu                                                                                                                                                                                                                                      | 6                                                       | CPU value that is allocated                                                         | Limit to CPU allocated to ocats-scp pod.                                                                                                                                                                                                                                                                                                                    |
| ocats-scp.resources.limits.memory                                                                                                                                                                                                                                   | 12Gi                                                    | Memory values that is allocated (in Gi or mi)                                       | Limit to memory allocated to ocats-scp pod .                                                                                                                                                                                                                                                                                                                |
| ocats-scp.resources.requests.cpu                                                                                                                                                                                                                                    | 6                                                       | CPU value that is allocated (must be less than or equal to limits)                  | Request of CPU allocation for ocats-scp.                                                                                                                                                                                                                                                                                                                    |
| ocats-scp.resources.requests.memory                                                                                                                                                                                                                                 | 12Gi                                                    | Memory value that is allocated (in Gi or mi) (must be less than or equal to limits) | Request of memory allocation for ocats-scp.                                                                                                                                                                                                                                                                                                                 |
| ocats-scp.service.customExtension.labels                                                                                                                                                                                                                            | {}                                                      | Label of node                                                                       | Node labels for node allocation during deployment.                                                                                                                                                                                                                                                                                                          |
| ocats-scp.service.customExtension.type                                                                                                                                                                                                                              | LoadBalancer                                            | ClusterIP, NodePort, LoadBalancer                                                   | Service type of ocats-scp pod                                                                                                                                                                                                                                                                                                                               |

**Table 3-22 (Cont.) ATS Parameters of the YAML File**

| Parameter                                               | Default Value                                               | Possible Values                                  | Description                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------------------------------------------------|-------------------------------------------------------------|--------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocats-scp.service.customExtension.port                  | 8080                                                        | Port number                                      | Port number of ocats-scp service.                                                                                                                                                                                                                                                                                                                                                                                 |
| ocats-scp.service.customExtension.staticNodePortEnabled | false                                                       | true, false                                      | Enabling of static node port.                                                                                                                                                                                                                                                                                                                                                                                     |
| ocats-scp.service.customExtension.staticNodePort        | false                                                       | Port number                                      | Port number of static node port.                                                                                                                                                                                                                                                                                                                                                                                  |
| ocats-scp.service.ports.http.port                       | 8080                                                        | Port number                                      | Port number of the ocats-scp service if https is not enabled.                                                                                                                                                                                                                                                                                                                                                     |
| ocats-scp.service.ports.http.staticNodePortEnabled      | false                                                       | true, false                                      | Enabling of static node port.                                                                                                                                                                                                                                                                                                                                                                                     |
| ocats-scp.service.ports.http.staticNodePort             | false                                                       | Port number                                      | Port number of static node port.                                                                                                                                                                                                                                                                                                                                                                                  |
| ocats-scp.service.ipFamilyPolicy                        | SingleStack                                                 | SingleStack, PreferDualStack, RequireDualStack   | ipFamilyPolicy to be allocated to the ocats-scp service.<br>This value will be referred to as whatever is defined at serviceIpFamilyPolicy.ocatsscpService under global parameters.                                                                                                                                                                                                                               |
| ocats-scp.service.ipFamilies                            | [IPv4]                                                      | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]         | ipFamilies to be allocated to the ocats-scp service.<br>This value will be referred to as the value defined at serviceIpFamilies.ocatsscpService under global parameters.                                                                                                                                                                                                                                         |
| SELECTED_NF                                             | SCP                                                         | NF name                                          | ATS parameters are set with default values in the ocats_ocscp_values_23.3.0.yaml file. Update ATS parameters with the actual value based on the environment, and then deploy the OCATS_OCSCP chart using this ocats_ocscp_values_23.3.0.yaml file. The updated ATS parameters are automatically applied during the deployment process, and the ATS will come up with the configuration as mentioned in this file. |
| NFNAMESPACE                                             | scpsvc                                                      | Update the SCP namespace                         | -                                                                                                                                                                                                                                                                                                                                                                                                                 |
| CLUSTERDOMAIN                                           | cluster.local                                               | Cluster Domain where SCP is Deployed             | -                                                                                                                                                                                                                                                                                                                                                                                                                 |
| DESTNAMESPACE                                           | scpsvc                                                      | Test Stubs NameSpace same as SCP Namespace       | -                                                                                                                                                                                                                                                                                                                                                                                                                 |
| ocats-dnsstub.image.repository                          | <docker-registryIP:docker-registryport>/ocats/ocats-dnsstub | <Image repository name:port>/ocats/ocats-dnsstub | Image repository and port of ocats-dnsstub.                                                                                                                                                                                                                                                                                                                                                                       |

**Table 3-22 (Cont.) ATS Parameters of the YAML File**

| Parameter                                  | Default Value                                                    | Possible Values                                                                      | Description                                                                                                                                                                                  |
|--------------------------------------------|------------------------------------------------------------------|--------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocats-dnsstub.image.tag                    | helm-tag                                                         | Value of tag to be deployed                                                          | Tag of ocats-dnsstub image.                                                                                                                                                                  |
| ocats-dnsstub.image.pullPolicy             | Always                                                           | Always, IfNotPresent, Never                                                          | Image pull policy of ocats-dnsstub image.                                                                                                                                                    |
| ocats-dnsstub.replicaCount                 | 1                                                                | Positive integers                                                                    | Replica count of ocats-dnsstub.                                                                                                                                                              |
| ocats-dnsstub.service.customExtension.type | ClusterIP                                                        | ClusterIP, NodePort, LoadBalancer                                                    | Service type of ocats-dnsstub pod.                                                                                                                                                           |
| ocats-dnsstub.service.customExtension.port | 53                                                               | Port                                                                                 | Port of ocats-dnsstub.                                                                                                                                                                       |
| ocats-dnsstub.service.ipFamilyPolicy       | SingleStack                                                      | SingleStack, PreferDualStack, RequireDualStack                                       | ipFamilyPolicy to be allocated to the ocats-dnsstub service.<br>This value will be referred to as whatever is defined at serviceIpFamilyPolicy.ocatsdnsstub Service under global parameters. |
| ocats-dnsstub.service.ipFamilies           | [IPv4]                                                           | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                             | ipFamilies to be allocated to the ocats-dnsstub service.<br>This value will be referred to as whatever is defined at serviceIpFamilies.ocatsdnsstubService under global parameters.          |
| ocats-dnsstub.resources.limits.cpu         | 1                                                                | CPU value that is allocated                                                          | Limit to CPU allocated to ocats-dnsstub pod.                                                                                                                                                 |
| ocats-dnsstub.resources.limits.memory      | 1Gi                                                              | Memory value that is allocated (in Gi or mi)                                         | Limit to memory allocated to ocats-dnsstub pod .                                                                                                                                             |
| ocats-dnsstub.resources.requests.cpu       | 1                                                                | CPU value that is allocated (must be less than or equal to limits)                   | Request of CPU allocation for ocats-dnsstub.                                                                                                                                                 |
| ocats-dnsstub.resources.requests.memory    | 1Gi                                                              | Memory values that is allocated (in Gi or mi) (must be less than or equal to limits) | Request of memory allocation for ocats-dnsstub.                                                                                                                                              |
| ocats-ddclientstub.image.repository        | <docker-registryIP:docker-registryport>/ocats/ocats-ddclientstub | <Image repository name:port>/ocats/ocats-ddclientstub                                | Image repository and port of ocats-ddclientstub.                                                                                                                                             |
| ocats-ddclientstub.image.tag               | helm-tag                                                         | Value of tag to be deployed                                                          | Tag of ocats-ddclientstub image.                                                                                                                                                             |
| ocats-ddclientstub.image.pullPolicy        | Always                                                           | Always, IfNotPresent, Never                                                          | Image pull policy of ocats-ddclientstub image.                                                                                                                                               |
| ocats-ddclientstub.replicaCount            | 1                                                                | Positive integers                                                                    | Replica count of ocats-ddclientstub.                                                                                                                                                         |
| ocats-ddclientstub.service.type            | LoadBalancer                                                     | ClusterIP, NodePort, LoadBalancer                                                    | Service type of ocats-ddclientstub.                                                                                                                                                          |

**Table 3-22 (Cont.) ATS Parameters of the YAML File**

| Parameter                                    | Default Value                                                             | Possible Values                                                                       | Description                                                                                                                                                                                              |
|----------------------------------------------|---------------------------------------------------------------------------|---------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocats-ddclientstub.ipFamilyPolicy            | SingleStack                                                               | SingleStack, PreferDualStack, RequireDualStack                                        | ipFamilyPolicy to be allocated to the ocatsddclientstubService service. This value will be referred to as whatever is defined at serviceIpFamilyPolicy.ocatsddclientstubService under global parameters. |
| ocats-ddclientstub.ipFamilies                | [IPv4]                                                                    | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                              | ipFamilies to be allocated to the ocatsddclientstubService service. This value will be referred to as whatever is defined at serviceIpFamilies.ocatsddclientstubService under global parameters.         |
| ocats-ddclientstub.resources.limits.cpu      | 1                                                                         | CPU value that is allocated                                                           | Limit to CPU allocated to ocats-ddclientstub pod.                                                                                                                                                        |
| ocats-ddclientstub.resources.limits.memory   | 1Gi                                                                       | Memory value that is allocated (in Gi or mi)                                          | Limit to memory allocated to ocats-ddclientstub pod.                                                                                                                                                     |
| ocats-ddclientstub.resources.requests.cpu    | 1                                                                         | CPU value that is allocated (should be less than or equal to limits)                  | Request of CPU allocation for ocats-ddclientstub.                                                                                                                                                        |
| ocats-ddclientstub.resources.requests.memory | 1Gi                                                                       | Memory value that is allocated (in Gi or mi) (should be less than or equal to limits) | Request of memory allocation for ocats-ddclientstub.                                                                                                                                                     |
| ocats-ddclientstub.log.level                 | INFO                                                                      | INFO, WARN, DEBUG                                                                     | Log level of ddclientstub pod.                                                                                                                                                                           |
| ocats-ddclientstub.kafka_broker              | "kafka-broker-0.kafka-broker.ddkafkanamespace.svc.cluster.local:9092"     | Kafka broker fqdn and port                                                            | Kafka broker fqdn or port for ddClientStub.                                                                                                                                                              |
| ocats-ddclientstub.string_topic_name         | "string_topic"                                                            | String topic                                                                          | ddClientStub string topic name.                                                                                                                                                                          |
| ocats-ddclientstub.json_topic_name           | "json_topic"                                                              | Json topic                                                                            | ddClientStub json topic name.                                                                                                                                                                            |
| ocats-scpglbratelimitstub.image.repository   | <docker-registry IP:docker-registry port>/ocats/ocats-scpglbratelimitstub | <Image repository name:port>/ocats/ocats-scpglbratelimitstub                          | Image repository and port of ocats-scpglbratelimitstub.                                                                                                                                                  |
| ocats-scpglbratelimitstub.image.tag          | helm-tag                                                                  | Value of tag to be deployed                                                           | Tag of ocats-scpglbratelimitstub.                                                                                                                                                                        |
| ocats-scpglbratelimitstub.image.pullPolicy   | Always                                                                    | Always, IfNotPresent, Never                                                           | Image pull policy of ocats-scpglbratelimitstub.                                                                                                                                                          |
| ocats-scpglbratelimitstub.replicaCount       | 1                                                                         | Positive integers                                                                     | Replica count of ocats-scpglbratelimitstub.                                                                                                                                                              |
| ocats-scpglbratelimitstub.service.type       | ClusterIP                                                                 | ClusterIP, NodePort, LoadBalancer                                                     | Service type of ocats-scpglbratelimitstub.                                                                                                                                                               |

**Table 3-22 (Cont.) ATS Parameters of the YAML File**

| Parameter                                                   | Default Value                                                | Possible Values                                                                       | Description                                                                                                                                                                                                                   |
|-------------------------------------------------------------|--------------------------------------------------------------|---------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocats-scpglbratelimitstub.service.ipFamilyPolicy            | SingleStack                                                  | SingleStack, PreferDualStack, RequireDualStack                                        | ipFamilyPolicy to be allocated to the ocatsscpglbratelimitstubService service.<br><br>This value will be referred to as whatever is defined at serviceIpFamilyPolicy.ocatsscpglbratelimitstubService under global parameters. |
| ocats-scpglbratelimitstub.service.ipFamilies                | [IPv4]                                                       | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                              | ipFamilies to be allocated to the ocatsscpglbratelimitstubService service.<br><br>This value will be referred to as whatever is defined at serviceIpFamilies.ocatsscpglbratelimitstubService under global parameters.         |
| ocats-scpglbratelimitstub.deployment.customExtension.labels | {}                                                           | Label of node                                                                         | Node labels for node allocation during deployment.                                                                                                                                                                            |
| ocats-scpglbratelimitstub.resources.limits.cpu              | 1                                                            | CPU value that is allocated                                                           | Limit to CPU allocated to ocats-scpglbratelimitstub.                                                                                                                                                                          |
| ocats-scpglbratelimitstub.resources.limits.memory           | 1Gi                                                          | Memory values that is allocated (in Gi or mi)                                         | Limit to memory allocated to ocats-scpglbratelimitstub.                                                                                                                                                                       |
| ocats-scpglbratelimitstub.resources.requests.cpu            | 1                                                            | CPU value that is allocated (should be less than or equal to limits)                  | Request of CPU allocation for ocats-scpglbratelimitstub.                                                                                                                                                                      |
| ocats-scpglbratelimitstub.resources.requests.memory:        | 1Gi                                                          | Memory value that is allocated (in Gi or mi) (should be less than or equal to limits) | Request of memory allocation for ocats-scpglbratelimitstub.                                                                                                                                                                   |
| ocats-scpglbratelimitstub.minreplicas                       | 1                                                            | Positive integer                                                                      | Minimum replicas of ocats-scpglbratelimitstub.                                                                                                                                                                                |
| ocats-scpglbratelimitstub.maxreplicas                       | 1                                                            | Positive integer                                                                      | Maximum replicas of ocats-scpglbratelimitstub.                                                                                                                                                                                |
| ocats-scpglbratelimitstub.maxPdbUnavailable                 | 1                                                            | Positive integer                                                                      | -                                                                                                                                                                                                                             |
| ocats-scpglbratelimitstub.log.level                         | INFO                                                         | INFO, WARN, DEBUG                                                                     | Log level of ocats-scpglbratelimitstub.                                                                                                                                                                                       |
| ocats-pystub.image.repository                               | <docker-registry IP:docker-registry port>/ocats/ocats-pystub | <Image repository name:port>/ocats/ocats-pystub                                       | Image repository and port of ocats-pystub.                                                                                                                                                                                    |
| ocats-pystub.image.tag                                      | helm-tag                                                     | Value of tag to be deployed                                                           | Tag of ocats-pystub.                                                                                                                                                                                                          |
| ocats-pystub.image.pullPolicy                               | Always                                                       | Always, IfNotPresent, Never                                                           | Image pull policy of ocats-pystub.                                                                                                                                                                                            |
| ocats-pystub.replicaCount                                   | 1                                                            | Positive integers                                                                     | Replica count of ocats-pystub.                                                                                                                                                                                                |
| RESPONSE_FROM_HEADER                                        | true                                                         | true, false                                                                           | When true pystub returns podname.                                                                                                                                                                                             |

**Table 3-22 (Cont.) ATS Parameters of the YAML File**

| Parameter                            | Default Value | Possible Values                                                                       | Description                                                                                                                                                                         |
|--------------------------------------|---------------|---------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ocats-pystub.resources.limits.cpu    | 1             | CPU value that is allocated                                                           | Limit to CPU allocated to ocats-pystubocats-pystub.resources.limits.memory.                                                                                                         |
| ocats-pystub.resources.limits.memory | 1Gi           | Memory values that is allocated (in Gi or mi)                                         | Limit to memory allocated to ocats-pystub.                                                                                                                                          |
| ocats-pystub.resources.requests.cpu  | 1             | CPU value that is allocated (should be less than or equal to limits)                  | Request of CPU allocation for ocats-pystub.                                                                                                                                         |
| ocats-pystub.resources.limits.memory | 1Gi           | Memory value that is allocated (in Gi or mi) (should be less than or equal to limits) | Request of memory allocation for ocats-pystub.                                                                                                                                      |
| ausf1.service.name:*                 | ausf1svc      | Service name of ausf1                                                                 | This is applicable for all the ocats-pystubs.                                                                                                                                       |
| ausf1.service.type:*                 | ClusterIP     | ClusterIP, NodePort, LoadBalancer                                                     | This is applicable for all the ocats-pystubs.                                                                                                                                       |
| ausf1.service.ports.port:*           | 8080          | Port number                                                                           | This is applicable for all the ocats-pystubs.                                                                                                                                       |
| ausf1.service.ipFamilyPolicy         | SingleStack   | SingleStack, PreferDualStack, RequireDualStack                                        | ipFamilyPolicy to be allocated to the ausf1Service service.<br>This value will be referred to as whatever is defined at serviceIpFamilyPolicy.ausf1Service under global parameters. |
| ausf1.service.ipFamilies             | [IPv4]        | [IPv4], [IPv6], [IPv4,IPv6], [IPv6,IPv4]                                              | ipFamilies to be allocated to the ausf1Service service.<br>This value will be referred to as whatever is defined at serviceIpFamilies.ausf1Service under global parameters.         |
| ausf1.deploymentName:*               | ausf1         | Deployment name of ausf1                                                              | This is applicable for all the ocats-pystubs.                                                                                                                                       |

### 3.6.1.5 Pushing the Images to Customer Docker Registry

#### Preparing to Deploy ATS and Stub Pod in Kubernetes Cluster

To deploy ATS and Stub Pods in the Kubernetes Cluster:

1. Click the file to download the CNC SCP ATS package file:

ocats\_ocscp\_csar\_24\_1\_1\_0\_0.pkg

The package contains the following files:

ocats\_ocscp\_csar\_24\_1\_1\_0\_0.zip  
mcafee-gen-ats-csar-24.1.1.log

Unzip the ocats\_ocscp\_csar\_24\_1\_1\_0\_0.zip file to get the following files and folders:

The output of this command is:

```
.
|-- Definitions
| |-- ocats_ocscp_ats_tests.yaml
| |-- ocats_ocscp_cne_compatibility.yaml
| `-- ocats_ocscp.yaml
|-- Files
| |-- ChangeLog.txt
| |-- Helm
| | `-- ocats-ocscp-24.1.1.tgz
| |-- Licenses
| |-- ocats-ddclientstub-24.1.1.tar
| |-- ocats-dnsstub-24.1.1.tar (Docker Image)
| |-- ocats-pystub-24.1.1.tar (Docker Image)
| |-- ocats-scp-24.1.1.tar (Docker Image)
| |-- ocats-scpglbratelimitstub-24.1.1.tar (Docker Image)
| |-- Oracle.cert
| `-- Tests
|-- ocats_ocscp_csar_24_1_1_0_0.zip
|-- ocats_ocscp.mf
|-- Scripts
| |-- ocats_ocscp_custom_serviceaccount_24.1.1.yaml (Template to create
custom service account)
| |-- ocats_ocscp_tests_jenkinsjobs_24.1.1.tgz. (ocscp_tests and jenkins
jobs folder to be copied if persistent volume is deployed)
| `-- ocats_ocscp_values_24.1.1.yaml (Custom values file for installation)
`-- TOSCA-Metadata
 `-- TOSCA.meta
```

2. Run the following commands in your cluster to load the ATS and stubs docker images, ocats-scp-image-24.1.3.tar, and push it to your registry:

```
docker load --input ocats-scp-24.1.1.tar
docker load --input oocats-dnsstub-24.1.1.tar
docker load --input ocats-pystub-24.1.1.tar
docker load --input ocats-scpglbratelimitstub.tar
docker load --input ocats-ddclientstub-24.1.1.tar
```

3. Run the following command in your cluster to load the ATS image:

```
docker load --input ocats-scp-images-24.1.1.tar
```

4. Run the following commands to push the ATS image to the registry:

```
docker tag ocats/ocats-scp:24.1.1 <local_registry>/ocats/ocatsscp:24.1.1
docker push <local_registry>/ocats/ocats-scp:24.1.1
```

Where, <local\_registry> indicates the registry where you can push the downloaded images.

5. Run the following commands to push the Stub image to the registry:

```
docker tag ocats/ocats-pystub:24.1.1 <local_registry>/ocats/ocats-
```



```
pystub:24.1.1
docker push <local_registry>/ocats/ocats-pystub:24.1.1
```

6. Run the following command to push the DNS Stub Image to the registry:

```
docker tag ocats/ocats-dnsstub:24.1.1 <local_registry>/ocats/ocats-
dnsstub:24.1.1
docker push <local_registry>/ocats/ocats-dnsstub:24.1.1
```

7. Run the following command to push the Global Rate Limiting Stub Image to the registry:

```
docker tag ocats/ocats-scpglbratelimitstub:24.1.1 <local_registry>/ocats/
ocats-scpglbratelimitstub:24.1.1
docker push <local_registry>/ocats/ocats-scpglbratelimitstub:24.1.1
```

8. Run the following command to push the Data Director Stub Image to the registry:

```
docker tag ocats/ocats-ddclientstub:24.1.1 <local_registry>/ocats/ocats-
ddclientstub:24.1.1
docker push <local_registry>/ocats/ocats-ddclientstub:24.1.1
```

9. In the **Scripts** folder, extract the following content:

```
ocats-ocscp-values-24.1.1.yaml
ocats-ocscp-custom-serviceaccount-24.1.1.yaml
ocats-ocscp-tests-jenkinsjobs-24.1.1.tgz
```

10. Update the image name and tag in the ocats-ocscp-values-24.1.1.yaml file as required.

### 3.6.1.6 Preinstall Preparation of SCP for SCP-ATS

Complete the following steps before performing an installation:

- When deploying default ATS with role binding, deploy ATS and test stubs in the same namespace as SCP.
- The SCP stub for the Global Egress Rate Limiting feature must be deployed by setting the required Helm parameters as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade and Fault Recovery Guide* to support the Global Egress Rate Limiting test cases.
- In the ocats\_ocscp\_values\_24.1.1.yaml, add the following for Prometheus that is required for alert test case:

```
traffic.sidecar.istio.io/excludeInboundPorts: "9090"
```

- If ASM is adding additional XFCC headers, the certExtractIndex and extractIndex of the xfccHeaderDecode value must be -1, otherwise certExtractIndex and extractIndex of xfccHeaderDecode value must be 0.
- If ASM is enabled, for fetching the metrics from Prometheus, a destination rule must be created. In most deployments, Prometheus is kept outside of the service mesh, so a

destination rule is required to communicate between a TLS enabled entity (ATS) and a non-TLS entity (Prometheus). The rule can be created as follows:

```
kubectl apply -f - <<EOF

apiVersion:networking.istio.io/v1alpha3
kind:DestinationRule
metadata:
 name:prometheus-dr
 namespace:ocscp
spec:
 host:oso-prometheus-server.ocscp.svc.cluster.local
 trafficPolicy:
 tls:
 mode:DISABLE
EOF
```

Where,

- name indicates the name of the destination rule.
  - namespace indicates where ATS is deployed.
  - host indicates the hostname of the Prometheus server.
  - FQDN and interPlmnFqdn must be the same for both nfServices for NRF profile NRF1.
- Sample**

```
NRF profiles for primary(Priority=0) and secondary(Priority=1) NRF. Note
that these NRFs needs to be backend DB Synced.
For Secondary NRF profile always make it priority lesser than First
priority NRF, currently we set secondary NRF priority to 1.
In case of no secondary NRF user can comment the secondary NRF Profile.
Service level FQDN's of NRF are from the same namespace as that of SCP,
this is put for SCP ATS cases. Otherwise, NRF's can be part of other
namespaces or even other k8s clusters.
nrfProfiles:
- capacity: 10000
 locality: USEast
 nfInstanceId: 6faf1bbc-6e4a-2828-a507-a14ef8e1bc5a
 nfStatus: REGISTERED
 nfType: NRF
 priority: '0'
 # with rel15 flag is enabled, below mentioned NRF region to be
specified.
nfSetIdList: ["Reg1"]
 # with rel16 flag is enabled, below mentioned NRF set Id to be
specified.
 nfSetIdList: ["setnrfl1.nrfset.5gc.mnc012.mcc345"]
 #Uncomment below section to configure interPlmnFqdn, plmnList or
snpnList
 #NRF-Change-3
 interPlmnFqdn: nrf1.5gc.mnc213.mcc410.3gppnetwork.org
 plmnList:
 - mcc: 410
 mnc: 213
```

```

- mcc: 410
 mnc: 214
#snpnList:
#- mcc: 345
mnc: 445
nid: 000007ed9d5
customInfo:
 preferredNrfForOnDemandDiscovery: true

nfServices:
- capacity: 5000
 #apiPrefix: USEast
 #NRF-Change-4
 fqdn: nrf1svc.scpsvc.svc.cluster.local
 interPlmnFqdn: nrf1svc.scpsvc.svc.cluster.local
 # Sample IPEndpoint with all the fields and it is commented out.
 #NRF-Change-5
 #ipEndpoints: [{"ipv4Address": "NRF-IP", "port": "NRF-PORT"}]
 #ipEndpoints: [{"ipv4Address": "10.75.213.56", "port": "31014"}]
 # ATS test cases need 8080 port with FQDN. Hence, in order to run
ATS cases, below "ipEndpoints" field is left uncommented.
 #NRF-Change-6
 ipEndpoints: [{"port": "8080"}]
 load: 0
 nfServiceStatus: REGISTERED
 scheme: http
 serviceInstanceId: fe137ab7-740a-46ee-aa5c-951806d77b01
 serviceName: nnrf-nfm
 priority: 0
 versions:
 - apiFullVersion: 1.0.0
 apiVersionInUri: v1

- capacity: 5000
 #apiPrefix: USEast
 #NRF-Change-4
 fqdn: nrf1svc.scpsvc.svc.cluster.local
 interPlmnFqdn: nrf1svc.scpsvc.svc.cluster.local
 # Sample IPEndpoint with all the fields and it is commented out.
 #NRF-Change-5
 #ipEndpoints: [{"ipv4Address": "NRF-IP", "port": "NRF-PORT"}]
 #ipEndpoints: [{"ipv4Address": "10.75.213.56", "port": "31014"}]
 # ATS test cases need 8080 port with FQDN. Hence, in order to run
ATS cases, below "ipEndpoints" field is left uncommented.
 #NRF-Change-6
 ipEndpoints: [{"port": "8080"}]
 load: 0
 nfServiceStatus: REGISTERED
 scheme: http
 serviceInstanceId: fe137ab7-740a-46ee-aa5c-951806d77b02
 serviceName: nnrf-disc
 priority: 0
 versions:
 - apiFullVersion: 1.0.0
 apiVersionInUri: v1

```

```

- capacity: 5000
 #apiPrefix: USEast
 #NRF-Change-4
 fqdn: nrf1svc.scpsvc.svc.cluster.local
 interPlmnFqdn: nrf1.5gc.mnc213.mcc410.3gppnetwork.org
 # Sample IPEndpoint with all the fields and it is commented out.
 #NRF-Change-5
 #ipEndpoints: [{"ipv4Address": "NRF-IP", "port": "NRF-PORT"}]
 #ipEndpoints: [{"ipv4Address": "10.75.213.56", "port": "31014"}]
 # ATS test cases need 8080 port with FQDN. Hence, in order to run
 #NRF-Change-6
 ipEndpoints: [{"port": "8080"}]
 load: 0
 nfServiceStatus: REGISTERED
 scheme: http
 serviceInstanceId: fe137ab7-740a-46ee-aa5c-951806d77b03
 serviceName: nnrf-oauth2
 priority: 0
 versions:
 - apiFullVersion: 1.0.0
 apiVersionInUri: v1

```

- Ensure SCP is deployed with the following parameters:
  - Make sure that while providing NRF information at the time of SCP deployment, stub NRF details like **nrf1svc** and **nrf2svc** should also be provided at the time of ATS deployment before running these test cases. For example, if the teststub namespace is scpsvc, then SCP should have been deployed with primary NRF as nrf1svc.scpsvc.svc.<clusterDomain> and secondary NRF as nrf2svc.scpsvc.svc.<clusterDomain> for NRF test cases to work.
  - Ensure the defaultTopologySource parameter is set to NRF in the ocscp\_values.yaml file.
  - Ensure the preventiveAuditOnLastNFInstanceDeletion parameter is set to false in the ocscp\_values.yaml file.
  - The number of replicas of all SCP microservices pods must be set to 1 during SCP deployment as ATS is enabled to perform metric validations for metrics obtained from a single pod.
  - When you deploy, make sure to define the additional NRF stubs needed for InterSCP cases as nrfr2l1svc (preferred NRF of Reg2), nrfr2l2svc (non-preferred NRF of Reg2), nrfr3l1svc (non-preferred NRF of Reg3), and nrfr3l2svc (preferred NRF of Reg3), which are provided in the default custom value file. Also, in the SCP deployment file, ensure that the namespace of all these NRFs is the same as the deployed SCP namespace. Reg1, Reg2, and Reg3 are replaced with setnrfl1.nrfset.5gc.mnc012.mcc345, setnrfr1.nrfset.5gc.mnc012.mcc345, and setnrfr2.nrfset.5gc.mnc012.mcc345 for Release 16 SCP deployment.
  - Ensure the supportedNRFRegionOrSetIdList must have Reg1, Reg2, and Reg3 for Release 15 SCP deployment or setnrfl1.nrfset.5gc.mnc012.mcc345, setnrfr1.nrfset.5gc.mnc012.mcc345, and setnrfr2.nrfset.5gc.mnc012.mcc345 for Release 16 SCP deployment.

- Ensure only Loc7, Loc8, Loc9, and USEast should be part of the servingLocalities for Release 15 SCP deployment and the servingScope for Release 16 SCP deployment.
- Recommended auditInterval is 60 seconds and guardTime is 10 seconds in the SCP deployment file.
- The regions such as Reg2 and Reg3 are the corresponding values for Release 15 SCP deployment, while NRF setIDS such as servingLocalities and setnrfr2.nrfset.5gc.mnc012.mcc345 are the corresponding values for Release 16 SCP deployment. The NRF belonging to either regions or the NRF set IDS's localities must not match with the SCP servingLocalities or SCP serving scope.
- SCP deployment file should have the attribute scpToRegisterWithNrfRegions set to Reg1 for Release 15 SCP deployment and setnrfl1.nrfset.5gc.mnc012.mcc345 for Release 16 SCP deployment. For information about Release 15 and Release 16, see 3GPP TS 23.501.
- To run CCA Validation feature tests, refer to [Configuring ATS for CCA Test Cases](#) section.
- To enable OAuth support while deploying SCP, refer [Configuring SCP to Run OAuth Test Cases in ATS](#) section.
- To enable alternate resolution service support while deploying SCP, refer to [Configuring SCP to Run DNS SRV Test Cases in ATS](#) section.
- To enable mediation support while deploying SCP, refer to [Configuring SCP to Run Mediation Test Cases in ATS](#) section.
- To enable nrproxy support, refer to [Configuring SCP to Run Model D Test Cases in ATS](#) section.
- To enable load manager support, refer to [Configuring SCP to Run LCI Test Cases in ATS](#) section.
- To enable the Global Egress Rate Limiting feature for ATS environment, refer to [Updating the Global Egress Rate Limiting Changes in the SCP Deployment File for ATS](#) section.
- By default, the ATS suite runs HTTPS test cases if the "ALL" option is selected, and SCP must be deployed with HTTPS support enabled to support the same. To enable HTTPs for ATS, refer to [Enabling HTTPs for ATS, pystubs and Jenkins](#)

```
If ingress gateway is available then set ingressGWAvalable flag to true
```

```
and provide ingress gateway IP and Port in publicSignalingIP and publicSignalingPort respectively.
```

```
 publicSignalingPort: &publicSignalingPort 8000 #Signaling PORT
 publicSignalingPortHttps: &publicSignalingPortHttps 9443
#Signaling PORT for HTTPS
 # uncomment below lines when deployed with release 16. Also, take a
note that http port for SCP should be same as "publicSignalingPort" of
SCP mentioned above.
 scpInfo:
 scpPrefix: scpPrefix
 scpPorts:
 http: *publicSignalingPort
 #Uncomment below key-value(https) to enable https for ingress
connections for rel16 deployment. this port should be same as
```

```
"publicSignalingPortHttps" of SCP mentioned above.
https: *publicSignalingPortHttps
Note: If this flag is false, then by default all connections to
PNF will be made using http protocol.
nativeEgressHttpsSupport: false
```

If SCP is deployed with HTTP support, either the single or multiple feature execution option must be selected, excluding all https test cases. Otherwise, make sure in an ASM environment where HTTPS is not enabled; in that case, manually remove the HTTPS-related test cases from the features directory on the ATS pod.

### 3.6.1.7 Preinstallation Preparation for SCP-ATS

Complete the following steps before performing an installation of SCP-ATS.

#### 3.6.1.7.1 Enabling Aspen Service Mesh

To enable Aspen Service Mesh (ASM) for ATS, complete the following procedure:

**Note**

By default, this feature is disabled.

1. If ASM is not enabled on the global level for the namespace, run the following command before deploying ATS:

```
kubectl label --overwrite namespace <namespace_name> istio-
injection=enabled
```

Example:

```
kubectl label --overwrite namespace scpsvc istio-injection=enabled
```

2. Add the following annotations in the lbDeployments section of the global section in the ocats\_scp\_values\_24.1.1.yaml file:

```
traffic.sidecar.istio.io/excludeOutboundPorts: "8091"
traffic.sidecar.istio.io/excludeInboundPorts: "8080"
```

Sample file with annotations:

```
lbDeployments:
 labels: {}
 annotations:
 traffic.sidecar.istio.io/excludeOutboundPorts: "8091"
 traffic.sidecar.istio.io/excludeInboundPorts: "8080"
```

3. Add Envoy Filter to enable the XFCC header forwarding by ASM sidecar.

## Envoy Filter for ATS:

```

apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
 name: <name>
 namespace: <namespace>
spec:
 workloadSelector:
 labels:
 app: ocats-scp
 configPatches:
 - applyTo: NETWORK_FILTER
 match:
 listener:
 filterChain:
 filter:
 name: "envoy.http_connection_manager"
 patch:
 operation: MERGE
 value:
 typed_config:
 '@type': type.googleapis.com/
envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManage
r
 forward_client_cert_details: ALWAYS_FORWARD_ONLY
 use_remote_address: true
 xff_num_trusted_hops: 1

```

## Envoy filter to enable the XFCC header forwarding on the application sidecar:

```

apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
 name: <name>
 namespace: <namespace>
spec:
 workloadSelector:
 labels:
 app.kubernetes.io/instance: ocscp
 configPatches:
 - applyTo: NETWORK_FILTER
 match:
 listener:
 filterChain:
 filter:
 name: "envoy.http_connection_manager"
 patch:
 operation: MERGE
 value:
 typed_config:
 '@type': type.googleapis.com/
envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManage
r
 forward_client_cert_details: ALWAYS_FORWARD_ONLY

```

```
use_remote_address: true
xff_num_trusted_hops: 1
```

Envoy filter to enable server header pass through on sidecar:

```
apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
 name: <name>
 namespace: <namespace>
spec:
 configPatches:
 - applyTo: NETWORK_FILTER
 match:
 listener:
 filterChain:
 filter:
 name: envoy.filters.network.http_connection_manager
 patch:
 operation: MERGE
 value:
 typed_config:
 '@type': type.googleapis.com/
 envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManager
 server_header_transformation: PASS_THROUGH
```

#### Note

- The sidecar configuration for response timeout or stream timeout should not be applied for any of the SCP microservices.
- For virtual service CRD, when the destinationhost is any SCP microservice, do not configure the timeout value.

### Updating Virtual Services

Disabling retry attempts in virtual services:

For all SCP and ATS pods in the virtual services, the retry attempt should be set to 0.

```
retries:
 attempts: 0
```

#### 3.6.1.7.2 Enabling Persistent Volume

ATS supports persistent storage to retain ATS historical build execution data, test cases, and one-time environment variable configurations.

To enable persistent storage:

1. Create a PVC and associate it with the ATS pod.
2. Set the **PVEnabled** flag to **true** in the `ocats_ocscp_values_24.1.1.yaml` file.



3. Set **PVClaimName** to PVC that is created for ATS.

```
ocats-scp:
 PVEnabled: false
 PVClaimName: "ocats-scp-24.1.1-pvc"
```

#### ① Note

In the event that Persistent Volume (PV) is enabled, ATS starts up with the parameter values specified in the `ocats_ocscp_values_24.1.1.yaml` file. If the ATS pod is restarted, the PV restores the configuration, ensuring that the new ATS pod will have the same configuration settings as the previous pod.

For more details on Persistent Volume Storage, you can refer to [Persistent Volume for 5G ATS](#).

## 3.6.2 Configuring SCP-ATS and SCP

This section provides information about updating ATS deployment configuration, enabling and disabling stubs, configuring SCP to run test cases, and so on.

#### ① Note

Make sure to follow the steps mentioned in the [Preinstall Preparation of SCP for SCP-ATS](#) to deploy SCP. For more information on SCP deployment, refer to the *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

### 3.6.2.1 Updating ATS Configuration

The following section covers updating the ATS deployment configuration and ATS input parameter configuration.

#### 3.6.2.1.1 Updating ATS Deployment Configuration

Previously, manual modification of the ATS configuration parameters for New feature and Regression jobs was required in the ATS graphical user interface (GUI). Now, with the introduction of a section "ATS\_Config" in the `ocats_ocscp_values_24.1.1.yaml` file, you can update the values for ATS parameters and then deploy the OCATS\_OCSCP charts using this modified `ocats_ocscp_values_24.1.1.yaml` file. The updated ATS parameters are automatically applied during the deployment process, and the ATS will come up with the configuration as mentioned in the `ocats_ocscp_values_24.1.1.yaml` file.

The `ocats_ocscp_values_24.1.1.yaml` file can be modified to update the ATS parameters according to their environment. To illustrate this, here's an example of how you can update the ATS parameters in the `ocats_ocscp_values_24.1.1.yaml` file:

**Figure 3-14** ATS Deployment Configuration

```

450 extraContainers: USE_GLOBAL_VALUE
451
452
453 ATS_Config:
454 SELECTED_NF: SCP
455 NF_NAMESPACE: scpsvc
456 CLUSTERDOMAIN: cluster.local
457 DEST_NAMESPACE: scpsvc
458 #Do not add a trailing / at the end of ATS_REGISTRY
459 ATS_REGISTRY: ocsfpf-registry.us.oracle.com:5000/ocats
460 AUDITINTERVAL: 60
461 GUARDTIME: 10
462 SCPSVCNAME: ocsdp-scp-worker
463 SCPCONFIGSVCNAME: ocsdp-scp-config
464 SCPCONFIGSVCNAME: ocsdp-scp-config
465 SCPCONFIGSVCNAME: ocsdp-scp-config
466 DBSECRETNAME: appuser-secret
467 MYSQLHOST: mysql.default.svc.cluster.local
468 ATSSUBIMAGE: ocats-pystub:tag
469 ATSSUBCPU: 1
470 ATSSUBMEMORY: 1G
471 SCPAUDITSVCNAME: ocsdp-scp-audit
472 RERUN_COUNT: 2
473 SCPSVCPORT: 8000
474 PROMSVNAME: ocne-prometheus-server.ocne-infra
475 PROMSVCPORT: 80
476 STUBWAITTIME: 300
477 PROMSVC_URI: /api/
478 SUBVALIDITYTIME: 120
479 OPENSEARCH_WAIT_TIME: 0
480 RERUNLOGS: false
481 OPENSEARCH_HOST: ocne-opensearch-cluster-master.ocne-infra
482 OPENSEARCH_PORT: 9200
483 #Please add STUB_LOG_COLLECTION value in quotes
484 STUB_LOG_COLLECTION: "no"
485 OPENSEARCH_LOGS: opensearch
486 SCPSVCHTTPSPORT: 9443
487 SCPNRFPROXYSVCNAME: ocsdp-scp-nrfproxy
488 KUBERNETES_HOST: kubernetes.default
489 KAFKANAMESPACE: scpsvc
490 SCPDDCLIENTSVCNAME: ocats-scpddclientstub
491 SCPLBRATELIMITSTUBSVCNAME: ocats-scpplbratelimitstub
492 SCPCONFIGSVCPORT: 8081
493

```

**Note**

Initially, at the time of deployment, you can configure or modify the parameter in the `ocats_ocsdp_values_24.1.1.yaml` file or you can also update or modify the parameters post deployment by following the process described in the [Configuring New Feature Pipelines](#) section.

### 3.6.2.1.2 Configuring SCP-ATS for OCI Setup

To leverage the existing infrastructure of Oracle Cloud, SCP which was only deployed on CNE, can now be integrated on the Oracle Cloud Infrastructure (OCI).

#### Creating Secret for Alarm(Alerts)

When deploying ATS within the OCI environment, users are required to provide the following inputs in the form of a Kubernetes secret named `ocats-oci-secret`:

```

user="<your user ocid>"
tenancy="<your tenancy ocid>"
region="<your oci region>"
fingerprint="<fingerprint of your public key>"
key_file="<full path to your private key>"
metric_namespace="<metric_namespace under which all the metrics and alarms of

```

```
SUT NF will be captured>"
nf_compartment_id="<Compartment Id of SUT NF>"
```

Run the command to create the secret for alarm or alerts related to test cases:

```
kubectl
 create secret generic ocats-oci-secret --from-
literal=user='<your_user_ocid>'--from-literal=tenancy='<your_tenancy_ocid>'--
from-literal=region='<your_oci_region>'--from-
literal=fingerprint='<fingerprint_of_your_oci_api_public_key>'--from-
literal=metric_namespace='<metric_namespace under which all the metrics and
alarms of SUT NF will be
captured>'--from-literal=nf_compartment_id='Compartment Id of SUT
NF'--from-file=key_file='<full_path_to_your_oci_api_private_key
on_the_host_machine_where_you_are_running_this_command>' -n
<namespace>
```

For example,

```
kubectl
 create secret generic ocats-oci-secret --from-
literal=user='ocid1.user.oc1..aaaaaaaajjxlzewn3e76aufhdjfhdkfjkl6ea3aaazgzx7cx
g74ljs5an3a'--from-
literal=tenancy='ocid1.tenancy.oc1..aaaaaaa5oqwziy4bngiebry6letze4hdjskjksdkd
lksurhc6pojwe4wxe34a'--from-literal=region='us-ashburn-1'--from-
literal=fingerprint='79:17:f2:89:76:d6:82:b2:13:b9:1d:9f:ff:92:28:3b'--from-
literal=metric_namespace=scpdemons
--from-
literal=nf_compartment_id='ocid1.compartment.oc1..aaaaaaa6crdjhjkkdkldlxbi7e
rwtmo3wa7jy6q6ldjjskddnmitot4smcczqg'--from-file=key_file='/tmp/
oci_api_key.pem' -n scpssc
```

### ATS Deployment Configuration in OCI Setup

Update the values for ATS parameters and then deploy the OCATS\_OCSCP charts using this modified `ocats_ocscp_values_24.1.1.yaml` file. To update the ATS parameters in the `ocats_ocscp_values_24.1.1.yaml` file, see the [Updating ATS Deployment Configuration](#) section.

Modify the `scpMetricVersion` parameter to "v2" in the `ocats_ocscp_values_24.1.1.yaml` file. For more information, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

### 3.6.2.1.3 Updating ATS Input Parameter Configuration

This section provides information about how to modify different services in SCP and configure SCP test cases in ATS.

#### 3.6.2.1.3.1 Enabling or Disabling Stubs

By default, all the stubs are enabled.

**Note**

Deploy NRF stubs with port 8080. NRF details of SCP should specify the **ipEndpoints** port as 8080 without any **ipv4Address** field. For example, ipEndpoints: [{"port": "8080"}]].

To enable or disable the stubs or pods, set the variable to `true` or `false`, respectively. You can install the required stubs or pods during `ocats-ocscp` deployments.

The following sample parameters show how the stub is enabled by setting different variables to `true` in the `ocats_ocscp_values_24.1.1.yaml` file:

```
global:
***** Sub-Section Start: Custom Extension Global Parameters *****
#*****
ocatsdnsstubService: true
ocatsscpService: true
ocatsscpglbratelimitstubService: true
ocatsddclientstubService: true
ausf1Stubs: true
ausf2Stubs: true
ausf3Stubs: true
ausf4Stubs: true
ausf5Stubs: true
ausf6Stubs: true
ausf7Stubs: true
ausf11Stubs: true
ausf12Stubs: true
ausf13Stubs: true
ausf14Stubs: true
ausf15Stubs: true
ausf16Stubs: true
ausf21Stubs: true
ausf22Stubs: true
ausf23Stubs: true
chf1Stubs: true
chf2Stubs: true
nrf1Stubs: true
nrf2Stubs: true
nrfr2l1Stubs: true
nrfr2l2Stubs: true
nrfr3l1Stubs: true
nrfr3l2Stubs: true
pcf1Stubs: true
pcf1cStubs: true
pcf2Stubs: true
pcf3Stubs: true
pcf4Stubs: true
pcf5Stubs: true
pcf6Stubs: true
pcf7Stubs: true
pcf8Stubs: true
pcf10Stubs: true
pcf11Stubs: true
pcf12Stubs: true
```

```
pcf13Stubs: true
pcf14Stubs: true
pcf15Stubs: true
pcf16Stubs: true
pcf21Stubs: true
pcf22Stubs: true
pcf23Stubs: true
pcf24Stubs: true
pcf25Stubs: true
pcf26Stubs: true
pcf27Stubs: true
pcf28Stubs: true
scp1Stubs: true
scp2Stubs: true
scp3Stubs: true
scp11Stubs : true
scp12Stubs: true
scp51Stubs: true
scp52Stubs: true
scp61Stubs: true
smf1Stubs: true
smf2Stubs: true
smf3Stubs: true
smf4Stubs: true
smf5Stubs: true
smf11Stubs: true
udm1Stubs: true
udm2Stubs: true
udm3Stubs: true
udm4Stubs: true
udm5Stubs: true
udm22Stubs: true
udm23Stubs: true
udm33Stubs: true
udm21Stubs: true
udm31Stubs: true
udm32Stubs: true
udr1Stubs: true
udr2Stubs: true
scp51svcxxxxStubs: true
scp52svcxxxxStubs: true
scp61svcxxxxStubs: true
sepp1Stubs: true
sepp2Stubs: true
sepp3Stubs: true
nef1Stubs: true
nef2Stubs: true
nef3Stubs: true
nef4Stubs: true
nef5Stubs: true
nef6Stubs: true
nef7Stubs: true
nef8Stubs: true
gen1Stubs: true
gen2Stubs: true
```

**Note**

Replica count of 'scp51svcxxxx' 'scp52svcxxxx' 'scp61svcxxxx' stubs must be set to Zero.

## 3.6.2.1.3.1.1 Modifying IpFamilyPolicy or IpFamilies of Stubs

The deployment of all stubs can adhere to the following:

- If IpFamilyPolicy is set to "SingleStack," then the value of IpFamilies can either be [IPv4] or [IPv6] only.
- If IpFamilyPolicy is set as "PreferDualStack" or "RequireDualStack", then the values of IpFamilies can either be [IPv4,IPv6] or [IPv6,IPv4] only.

**Note**

All the pyStubs should be deployed with the same combination of IpFamilyPolicy and IpFamilies.

## 3.6.2.1.3.2 Configuring ATS YAML File for Deployment

**Umbrella Chart**

Helm charts contain different charts that are referred to as subcharts through their dependencies section in the `requirements.yaml` file. When a chart is created for the purpose of grouping related subcharts or services, such as composing a whole application or deployment, it is known as umbrella chart.

Helm umbrella charts are created to deploy generic ATS packages in CDCS through a single `values.yaml` file.

Perform the following procedure to create umbrella charts and add stubs to the umbrella charts:

1. To add stubs to the umbrella chart, do the following:
  - a. Below parameters can be updated to `ocats_ocscp_values_24.1.1.yaml` file:

```
ocats-scp:
 image:
 repository: cgbu-ocscp-dev-docker.dockerhub-phx.oci.oraclecorp.com/ocats-ocats-scp
 tag: 24.1.1
 pullPolicy: Always
 replicaCount: 1
 resources:
 limits:
 cpu: 3
 memory: 23Gi
 #ephemeral-storage: 4Gi
 requests:
 cpu: 3
 memory: 3Gi
```

**ocats-dnsstub**

```
ocats-dnsstub:
 image:
 repository: cgbu-ocscp-dev-docker.dockerhub-phx.oci.oraclecorp.com/
ocats/ocats-dnsstub
 tag: helm-tag
 pullPolicy: Always
 replicaCount: 1
 service:
 customExtension:
 labels: {}
 annotations: {}
 type: ClusterIP
 port: 53

 deployment:
 customExtension:
 labels: {}
 annotations: {}

 resources:
 limits:
 cpu: 1
 memory: 1Gi
 requests:
 cpu: 1
 memory: 1Gi
```

**ocats-pystub**

```
ocats-pystub:
 image:
 repository: cgbu-ocscp-dev-docker.dockerhub-phx.oci.oraclecorp.com/
ocats/ocats-pystub
 tag: helm-tag
 pullPolicy: Always
 replicaCount: 1
 resources:
 limits:
 cpu: 1
 memory: 1Gi
 requests:
 cpu: 1
 memory: 1Gi
 ausf1:
 service:
 name: ausf1svc
 type: ClusterIP
 ports:
 port: 8080
 deploymentName: ausf1
 ausf2:
 service:
 name: ausf2svc
```

```

 type: ClusterIP
 ports:
 port: 8080
 deploymentName: ausf2

```

### ocats-scpglbratelimitstub

```

ocats-scpglbratelimitstub:
 image:
 repository: <docker-registry IP:docker-registry port>/ocats/ocats-
 scpglbratelimitstub
 tag: helm-tag
 pullPolicy: Always
 replicaCount: 1
 service:
 type: ClusterIP
 deployment:
 customExtension:
 labels: {}
 annotations: {}
 resources:
 limits:
 cpu: 1
 memory: 1Gi
 requests:
 cpu: 1
 memory: 1Gi
 minreplicas: 1
 maxreplicas: 1
 maxPdbUnavailable: 11:27
ocats-ddclientstub:
 image:
 repository: <docker-registry IP:docker-registry port>/ocats/ocats-
 ddclientstub
 tag: helm-tag
 pullPolicy: Always
 replicaCount: 1
 service:
 type: LoadBalancer
 resources:
 limits:
 cpu: 1
 memory: 1Gi
 #ephemeral-storage: 55Mi
 requests:
 cpu: 1
 memory: 1Gi
 #ephemeral-storage: 55Mi
 log:
 level: INFO
 extraContainers: USE_GLOBAL_VALUE

 kafka_broker: "kafka-broker1-0.kafka-
 broker1.ddkafkanamespace.svc.cluster.local:9092"

```



```
string_topic_name: "string_topic"
json_topic_name: "json_topic"
```

### ocats-dnsstub

```
ocats-dnsstub:
 image:
 repository: <docker-registry IP:docker-registry port>/ocats/ocats-
dnsstub
 tag: helm-tag
 pullPolicy: Always
 replicaCount: 1
 service:
 customExtension:
 labels: {}
 annotations: {}
 type: ClusterIP
 port: 53

 deployment:
 customExtension:
 labels: {}
 annotations: {}

 resources:
 limits:
 cpu: 1
 memory: 1Gi
 requests:
 cpu: 1
 memory: 1Gi
 extraContainers: USE_GLOBAL_VALUE
```

### 3.6.2.1.3.3 Dual Stack Support

Using the dual stack mechanism, applications or NFs can establish connections with pods and services in a Kubernetes cluster using IPv4 or IPv6 or both simultaneously.

With the introduction of the Dual Stack IPv6 support feature, there will be two categories of features performing the same tests. However, they are categorized differently based on the type of endpoint the stack supports (single endpoint or multiple endpoints). It is important to consider the IP endpoint support of the stack while running these tests.

For example:

- **SCP\_22.3.0\_BugFixes\_MultipleIpEndpoint\_P0.feature:** Shall run on stacks supporting multiple endpoints (IPv4 and IPv6)
- **SCP\_22.3.0\_BugFixes\_SingleIpEndpoint\_P0.feature:** Shall run on stacks supporting single endpoints (IPv4 or IPv6)

Feature files with "SingleIpEndpoint" in the name:

These test cases run on a single IP stack setup or a dual IP stack setup, where ipFamilies should be [IPv4] or [IPv6] in the ATS deployment file.

Feature files with "MultipleIpEndpoint" in name:

These test cases run only on a dual IP stack setup, where ipFamilies should be [IPv4, IPv6] or [IPv6, IPv4] in the ATS deployment file.

All other feature files can run on both setups.

#### 3.6.2.1.3.4 Enabling HTTPs for ATS, pystubs and Jenkins

Perform the following procedure to enable HTTPs for ATS. These steps can be avoided if your environment does not require HTTPS by adhering to the [Preinstall Preparation of SCP for SCP-ATS](#).

Ensure that you have the below file generated before you proceed with deployment:

- Private key
  - Client certificate
  - Certificate Authority Root certificate
1. To enable HTTPS for ATS, run the following command to create the Kubernetes secret:

```
kubectl create secret generic ocats-scp-secret
--from-file=rsa_private_key_pkcs1_client.pem --from-file=client.pem
--from-file=caroot.pem
--from-file=jenkinsserver.jks -n scpssc
```

#### Note

Name of the secret, private key, client certificate, and CA root certificate must be the same as mentioned in the above mentioned command.

- a. Uncomment the private key, client certificate, and CA root certificate in the `ocats_scp_values.yaml` as follows:

Existing value:

```
#certificates:
cert_secret_name: "ocats-scp-secret"
ca_cert: "caroot.pem"
client_cert: "client.pem"
private_key: "rsa_private_key_pkcs1_client.pem"
jks_file: "jenkinsserver.jks"
jks_password: "123456" #This is the password given to the jks file
while creation.
```

Updated values

```
certificates:
 cert_secret_name: "ocats-scp-secret"
 ca_cert: "caroot.pem"
 client_cert: "client.pem"
 private_key: "rsa_private_key_pkcs1_client.pem"
 jks_file: "jenkinsserver.jks"
 jks_password: "123456" #This is the password given to the jks file
while creation.
```

- b. Set the `atsGuiTLSEnabled` and `atsCommunicationTLSEnabled` parameters to true in `ocats_scp_values.yaml`, as shown below:

Generate Certificate

```
cat >>$SERVER_EXT<<EOF
authorityKeyIdentifier=keyid,issuer
basicConstraints=CA:FALSE
keyUsage = digitalSignature, nonRepudiation, keyEncipherment,
dataEncipherment
extendedKeyUsage = serverAuth, clientAuth
subjectAltName = @alt_names

[alt_names]
IP.1 = 127.0.0.1
replace IP.2 with local setup IP when enabling https Gui for jenkins
IP.2 = 10.75.226.134
DNS.1 = *.${NAMESPACE}.${COMMON_NAME}
DNS.2 = localhost
EOF
```

- c. Make changes to the `SERVER_EXT`. Set the `IP.2` value as the local setup IP as mentioned below:

Updated Value

```
atsGuiTLSEnabled: true
atsCommunicationTLSEnabled: true
```

- d. Deploy ATS pod with above mentioned changes.

2. To enable HTTPS for pystubs, run the following command to create the Kubernetes secret:

```
kubectl create secret generic ocats-pystub-secret --from-
file=rsa_private_key_pkcs1_server.pem --from-file=server.pem --from-
file=caroot.pem -n scpsvc
```

#### Note

Name of the secret, private key, client certificate, and CA root certificate must be the same as mentioned in the above mentioned command.

- a. Run the following command to create role and role binding for GET access to the pystub pod:

```
kubectl create role scpsvc-pystub-scp-role --verb=get --
resource=secrets -n scpsvc
kubectl create rolebinding --role=scpsvc-pystub-scp-role scpsvc-pystub-
scp-rolebinding --serviceaccount=scpsvc:default -n scpsvc
```

- b. Uncomment the private key, server certificate, and CA root certificate in the `ocats_pystub_values.yaml` file as follows:

Existing value:

```
#certificates:
cert_secret: "ocats-pystub-secret"
ca_cert: "caroot.pem"
server_cert: "server.pem"
private_key: "rsa_private_key_pkcs1_server.pem"
```

Updated values in ocats\_ocscp\_values\_24.1.1.yaml file.

```
certificates:
 cert_secret: "ocats-pystub-secret"
 ca_cert: "caroot.pem"
 server_cert: "server.pem"
 private_key: "rsa_private_key_pkcs1_server.pem"
```

- c. Deploy pystub pods with above mentioned changes.

### 3.6.2.1.3.5 Configuring SCP to Run OAuth Test Cases in ATS

By default, the ATS suite runs oauth test cases if the "ALL" option is selected, and SCP must be deployed with oauth support enabled to support the same.

```
Enable nrproxy-oauth service (only for Rel16)
nrProxyOAuthService: true
```

If SCP is deployed without oauth support, either the single or multiple feature execution option must be selected, excluding all oauth test cases, or these options must be manually removed from the feature directory on the ATS pod.

### 3.6.2.1.3.6 Configuring ATS for CCA Test Cases

Run CCA Validation feature tests with ATS.

- Enable HTTPs to run these feature tests.
- Generate 9 client certificates (using the gen\_certificates.sh script) before running the feature tests and write these certificates inside the ocats-scp-secret kubernetes secret. This script is to be executed when SCP-ATS deployments have IP families such as [IPv4] or [IPv4, IPv6]. The following lists the sample SANs that must be used while creating certificates:

#### — client2.pem

```
[alt_names]

IP = 10.75.213.1

DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}

URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177

URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
```

```
URI.3 = https://10.75.213.1:443
```

— **client3.pem**

```
[alt_names]
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
```

— **client4.pem**

```
[
alt_names]
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
```

— **client5.pem**

```
[alt_names]
IP = 10.75.213.1
```

— **client6.pem**

```
[alt_names]
URI.1 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
```

— **client7.pem**

```
[alt_names]
URI.1 = https://10.75.213.1:443
```

— **client8.pem**

```
[alt_names]
IP = 10.75.213.1
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
URI.2 = https://10.75.213.1:443
```

— **client9.pem**

```
[alt_names]
IP.1 = 10.75.213.2
IP.2 = 10.75.213.3
IP.3 = 10.75.213.4
IP.4 = 10.75.213.5
IP.5 = 10.75.213.6
IP.6 = 10.75.213.7
IP.7 = 10.75.213.8
IP.8 = 10.75.213.9
IP.9 = 10.75.213.10
```

```
IP.10 = 10.75.213.11
IP.11 = 10.75.213.1
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
URI.3 = https://10.75.213.10:443
```

#### — client10.pem

```
[alt_names]
IP = 10.75.213.1
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
URI.3 = https://10.75.213.10:443
URI.4 = https://10.75.213.2:443
URI.5 = https://10.75.213.3:443
URI.6 = https://10.75.213.4:443
URI.7 = https://10.75.213.5:443
URI.8 = https://10.75.213.6:443
URI.9 = https://10.75.213.7:443
URI.10 = https://10.75.213.8:443
URI.11 = https://10.75.213.1:443
```

- Generate 9 client certificates before running the feature tests, and write these certificates inside the ocats-scp-secret Kubernetes secret. Run the `gen_certificates_ipv6.sh` script when SCP-ATS deployment is either [IPv6] or [IPv6,IPv4]. The following lists the sample SANs that must be used while creating certificates:

#### — client2.pem

```
[alt_names]
IP = 2001:db8:85a3:0:0:8a2e:370:7334

DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}

URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177

URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443

URI.3 = https://[2001:db8:85a3:0:0:8a2e:370:7334]:443
```

#### — client3.pem

```
[alt_names]
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
```

#### — client4.pem

```
[
alt_names]
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
```

- **client5.pem**

```
[alt_names]
IP = 2001:db8:85a3:0:0:8a2e:370:7334
```

- **client6.pem**

```
[alt_names]
URI.1 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
```

- **client7.pem**

```
[alt_names]
URI.1 = https://[2001:db8:85a3:0:0:8a2e:370:7334]:443
```

- **client8.pem**

```
[alt_names]
IP = 2001:db8:85a3:0:0:8a2e:370:7334
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
URI.2 = https://[2001:db8:85a3:0:0:8a2e:370:7334]:443
```

- **client9.pem**

```
[alt_names]
IP.1 = 2002:db8:85a3:0:0:8a2e:370:7334
IP.2 = 2003:db8:85a3:0:0:8a2e:370:7334
IP.3 = 2004:db8:85a3:0:0:8a2e:370:7334
IP.4 = 2005:db8:85a3:0:0:8a2e:370:7334
IP.5 = 2006:db8:85a3:0:0:8a2e:370:7334
IP.6 = 2007:db8:85a3:0:0:8a2e:370:7334
IP.7 = 2008:db8:85a3:0:0:8a2e:370:7334
IP.8 = 2009:db8:85a3:0:0:8a2e:370:7334
IP.9 = 2010:db8:85a3:0:0:8a2e:370:7334
IP.10 = 2011:db8:85a3:0:0:8a2e:370:7334
IP.11 = 2001:db8:85a3:0:0:8a2e:370:7334
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
URI.3 = https://[2010:db8:85a3:0:0:8a2e:370:7334]:443
```

- **client10.pem**

```
[alt_names]
IP = 2001:db8:85a3:0:0:8a2e:370:7334
DNS.1 = chf1svc.${NAMESPACE}.${COMMON_NAME}
URI.1 = urn:uuid:11111111-aaaa-aaaa-aaaa-111111111177
URI.2 = https://chf1svc.${NAMESPACE}.${COMMON_NAME}:443
```

```

URI.3 = https://[2010:db8:85a3:0:0:8a2e:370:7334]:443
URI.4 = https://[2002:db8:85a3:0:0:8a2e:370:7334]:443
URI.5 = https://[2003:db8:85a3:0:0:8a2e:370:7334]:443
URI.6 = https://[2004:db8:85a3:0:0:8a2e:370:7334]:443
URI.7 = https://[2005:db8:85a3:0:0:8a2e:370:7334]:443
URI.8 = https://[2006:db8:85a3:0:0:8a2e:370:7334]:443
URI.9 = https://[2007:db8:85a3:0:0:8a2e:370:7334]:443
URI.10 = https://[2008:db8:85a3:0:0:8a2e:370:7334]:443
URI.11 = https://[2001:db8:85a3:0:0:8a2e:370:7334]:443

```

### 3.6.2.1.3.7 Updating the Global Egress Rate Limiting Changes in the SCP Deployment File for ATS

Perform the following procedure to enable the Global Egress Rate Limiting feature for ATS environment.

1. In the SCP `custom-values.yaml` file, update the following:
  - `federation.remoteScpOne.fqdnOrIp`: FQDN of the `scpglbratelimitstub` pod.
  - `federation.remoteScpOne.clusterName`: Coherence Cluster Name of global rate limit Stub [Example: `scpstub-coherence-cluster`].
  - `federation.remoteScpOne.nfInstanceId`: NFInstanceId of global rate limit Stub [Example: `2faf1bbc-6e4a-4454-a507-a14ef8e1bc22`].

### 3.6.2.1.3.8 Configuring SCP to Run DNS SRV Test Cases in ATS

By default, the ATS suite runs alternate resolution (DNS SRV) if the "ALL" option is selected, and SCP must be deployed with alternate resolution service support enabled in order to support the same.

```
Enable DNS SRV Alternate Routing Feature
dnsSRVAlternateRouting: false
```

If SCP is deployed without alternate resolution support, either single or multiple feature execution options must be selected, excluding all DNS SRV test cases, or these options must be manually removed from the feature directory on the ATS pod.

### 3.6.2.1.3.9 Configuring SCP to Run Mediation Test Cases in ATS

By default, the ATS suite runs Mediation test cases if the "ALL" option is selected, and SCP must be deployed with Mediation support enabled to support the same.

```
Enable mediation service
mediationService: false
```

If SCP is deployed without the Mediation support, then either a single or multiple feature execution option can be selected, excluding all Mediation test cases, or these options must be manually removed from the feature directory on the ATS pod.



### 3.6.2.1.3.10 Configuring SCP to Run Model D Test Cases in ATS

By default, the ATS suite runs delegated discovery ( Model D) if the "ALL" option is selected, and SCP must be deployed with nrproxy support enabled to support the same.

```
Enable Nrf Proxy service (only for Rel16)
nrproxyService: false
```

If SCP is deployed without nrproxy support, either single or multiple feature execution options must be selected, excluding all Model D test cases, or these options must be manually removed from the feature directory on the ATS pod.

### 3.6.2.1.3.11 Configuring ATS for Traffic Feed Test Cases

1. To run Traffic Feed ATS testcases, install the kafka broker.
2. In the ATS deployment file, update the following parameters:
  - kafka\_broker: <kafka broker host>:<kafka broker host> (Example, "kafka-broker-0.kafka-broker.scpsvc.svc.cluster.local:9092")
  - string\_topic\_name: <topic name for string serialization> (Example, "string\_topic")
  - json\_topic\_name: <topic name for json serialization> (Example, "json\_topic")
3. In the global.yaml file, update the following parameters:
  - global\_traffic\_feed\_key\_Serializer: <key serialization> (Example, string)
  - global\_traffic\_feed\_value\_Serializer: <value serialization> (Example, string)
  - global\_traffic\_feed\_topic\_name: <topic name for selected serialization> (Example, string\_topic)
  - global\_traffic\_feed\_bootstrap\_server\_host: <kafka broker host> (Example, kafka-broker1-0.kafka-broker1.scpsvc.svc.cluster.local)
  - global\_traffic\_feed\_bootstrap\_server\_port: <kafka broker port> (Example, 9092)For more information on global.yaml file, see the [ATS Testcase Parametrization on User Input](#).

For installation of the Data Director Kafka broker, perform the following:

- Update the ocnadd-custom-values.yaml as documented in the Data Director Installation and Upgrade Guide.
- Disable all services apart from ocnaddkafka by marking them as false.
- Keep ocnaddkafka as true.

For more information, see the *Oracle Communications Network Analytics Data Director Installation and Upgrade Guide*.

#### Note

Kafka broker should be deployed with 3 partitions under string\_topic.

### 3.6.2.1.3.12 Configuring SCP to Run LCI Test Cases in ATS

By default, the ATS suite runs load manager test cases if the "ALL" option is selected, and SCP must be deployed with load manager support enabled to support the same.

```
Enable load-manager service (only for Rel16)
loadManagerService: true
```

If SCP is deployed without load manager support, either the single or multiple feature execution option must be selected, excluding all load manager test cases, or these options must be manually removed from the feature directory on the ATS pod.

## 3.6.3 Deploying ATS and Stub in the Kubernetes Cluster

This section provides information about how to deploy ATS and stubs using Command Line Interface (CLI). To deploy ATS and stubs using CDCS, see *Oracle Communications CD Control Server User Guide*.

To deploy ATS in the Kubernetes Cluster:

#### Note

Deploy ATS, SCP, and stubs in the same namespace.

1. Ensure the `ocats_ocscp_values_24.1.1.yaml` file has been updated with correct repository, image tag, and parameters as per requirements.
2. In the Files or Helm folder, find the `ocats-ocscp` charts for version 23.3.0, which have to be used for installation.
3. Run the following command to deploy ATS:

```
helm install ocats-ocscp-24.1.1.tgz --name <release_name> --namespace
<namespace_name> -f ocats-ocscp-values-24.1.1.yaml
```

Example:

```
helm install ocats-ocscp-24.1.1.tgz --name ocats-ocscp --namespace scpsvc-f
ocats-ocscp-values-24.1.1.yaml
```

#### Note

Update image name, tag, service name, and deployment name in `ocats-pystub` of the `ocats_pystub_values_24.1.1.yaml` file before deploying.

4. Verify whether all stub pods are up and running in the deployed namespace as updated in the `ocats_ocscp_values_24.1.1.yaml`.

## 3.6.4 Post Installation and Deployment Steps

This section describes the post-installation steps for SCP.

### 3.6.4.1 Verifying ATS Deployment

Run the following command to verify the ATS deployment status:

```
helm status <release_name>
```

#### ① Note

If ATS is deployed in the service mesh environment, the **Ready** field for pods displays 2/2.

The following image displays that the deployment is complete because the **STATUS** field has changed to deployed.

**Figure 3-15** Checking ATS Helm Release and ATS related Pod Status

```
[opc@ocscp-ocncc-oci-phx-bell-bastion-01 ~]$ helm status ocats-ocscp -n scp-46754295-dev
NAME: ocats-ocscp
LAST DEPLOYED: Mon Oct 16 08:45:05 2023
NAMESPACE: scp-46754295-dev
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
Copyright 2020 (C), Oracle and/or its affiliates. All rights reserved.

Thank you for installing ocats-ocscp.

Your release is named ocats-ocscp , Release Revision: 1.
To learn more about the release, try:

 $ helm status ocats-ocscp
 $ helm get ocats-ocscp
[opc@ocscp-ocncc-oci-phx-bell-bastion-01 ~]$ kubectl get pods -n scp-46754295-dev | grep "ocats\|udr1"
ocats-ocscp-ocats-ddclientstub-7f894b65c9-gjqlm 2/2 Running 0 26h
ocats-ocscp-ocats-dnsstub-5d5fcfcf9f-bp8w2 2/2 Running 0 26h
ocats-ocscp-ocats-scp-6f44bbdfdc-wkbpq 2/2 Running 0 26h
ocats-ocscp-ocats-scpglbratelimitstub-55d7dc5b74-2njzn 2/2 Running 0 26h
udr1-59d4d694c4-gzqqc 2/2 Running 0 26h
[opc@ocscp-ocncc-oci-phx-bell-bastion-01 ~]$
```

### 3.6.4.2 Modifying the scpc-alternate-resolution Microservice

Perform the following procedure to modify the scpc-alternate-resolution microservice to point to the DNS Stub for the Alternate Routing based on the Domain Name System (DNS) Service Record (SRV) Records feature.

1. Capture the cluster IP of DNS Stub service.

By default, scpc-alternate-resolution points to CoreDNS and displays the following settings in the `ocscp_values.yaml` deployment file.

**Figure 3-16** CoreDNS

```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

2. Run the following command to change the deployment file to add content in `scpc-alternate-resolution` to query the DNS Stub.

Uncomment `ocscp-alternate-resolution`'s `dnsConfig` and `dnsPolicy` before deployment to allow editing.

```
$kubectl edit deployment ocscp-scpc-alternate-resolution -n
scpsvc
```

Sample deployment file:

```
dnsConfig:
 nameservers:
 - 10.96.77.54
 searches:
 - cidscpsvc-230228133808.svc.cluster.local
 - svc.cluster.local
 - cluster.local
dnsPolicy: None
```

Add the following content:

- `nameservers`: Add the IP address that you recorded after installing the DNS Stub (cluster IP of the DNS Stub).
- `searches`: Add all the search based on the cluster name.
  - `scpsvc`  
This is the namespace.
  - `cluster.local`  
This is the cluster name.
- `dnsPolicy`: Set it to "None" if it is not already set by default.

### 3.6.4.3 ATS Testcase Parametrization on User Input

Parameterization is an approach to decoupling the data from the feature files in ATS so that ATS test cases can be run against some predefined configuration that contains customer-specific data and service configurations. Parametrization allows you to provide or adjust values for the input and output parameters needed for the test cases to be compatible with the SUT configuration. You can update or adjust the key-value pair values in the `global.yaml` and `feature.yaml` files for each of the feature files so that they are compatible with SUT configuration. For more information, see the [Parameterization](#) section.

Three new folders "`cust_data`", "`product_config`" and "`custom_config`" are added to the base path inside ATS pod `/var/lib/jenkins/ocscp_tests`. Where the `cust_data` folder is a replica of the already existing data folder, the `product_config` folder contains configuration files that are compatible with the default product configurations, and the `cust_config` folder is a replica of the `product_config` folder. You can update the custom folders, such as `cust_data` and `custom_config`.

#### Product Config folder

The product config folder contains two types of YAML files: `global.yaml` and `feature_name>.yaml` (feature file-specific yaml).

#### Global File

The `global.yaml` contains global variable names and their corresponding default values that can be used across all the feature files. The variable name declared in `global.yaml` should start with `global_Var_` as prefix.

For example, `<key_name>: &<variable_name> <default_value_of_variable>`

```
#START_GLOBAL
global:
 File_Parameters:
 global_Var_SUT_apiPrefix1: &global_Var_SUT_apiPrefix1
USEast
 global_Var_SUT_ServingLocality1:
&global_Var_SUT_ServingLocality1 USEast
 global_Var_localNrfSetId1: &global_Var_localNrfSetId1
setnrfl1.nrfset.5gc.mnc012.mcc345
 global_Var_stubPort: &global_Var_stubPort 8080
#END_GLOBAL
```

Sample:

```
#START_GLOBAL
global:
 File_Parameters:
 global_Var_SUT_apiPrefix1: &global_Var_SUT_apiPrefix1 USEast
 # Following Serving Localities if needs to be changed are also to be
changed while deploying SCP before running ATS
 global_Var_SUT_ServingLocality1: &global_Var_SUT_ServingLocality1 USEast
 global_Var_SUT_ServingLocality2: &global_Var_SUT_ServingLocality2 Loc7
 global_Var_SUT_ServingLocality3: &global_Var_SUT_ServingLocality3 Loc8
 global_Var_SUT_ServingLocality4: &global_Var_SUT_ServingLocality4 Loc9

 # Following SetId's if needs to be changed are also to be changed while
deploying SCP before running ATS
 global_Var_localNrfSetId1: &global_Var_localNrfSetId1
setnrfl1.nrfset.5gc.mnc012.mcc345
 global_Var_remoteNrfSetId1: &global_Var_remoteNrfSetId1
setnrfr1.nrfset.5gc.mnc012.mcc345
 global_Var_remoteNrfSetId2: &global_Var_remoteNrfSetId2
setnrfr2.nrfset.5gc.mnc012.mcc345

 # If stubPort has to be changed then stub has to be deployed with the
same port number before running ATS
 global_Var_stubPort: &global_Var_stubPort 8080
 global_Var_stubErrorCode: &global_Var_stubErrorCode 404

 global_Var_udm_nfSetIdList1: &global_Var_udm_nfSetIdList1
set1.udmset.5gc.mnc012.mcc345
 global_Var_udm_nfSetIdList2: &global_Var_udm_nfSetIdList2
set2.udmset.5gc.mnc012.mcc345
 global_Var_udm_nfSetIdList3: &global_Var_udm_nfSetIdList3
set3.udmset.5gc.mnc012.mcc345

 global_Var_smf_nfSetIdList1: &global_Var_smf_nfSetIdList1
set1.smfset.5gc.mnc012.mcc345
 global_Var_smf_nfSetIdList2: &global_Var_smf_nfSetIdList2
```

```

set2.smfset.5gc.mnc012.mcc345
 global_Var_smf_nfSetIdList3: &global_Var_smf_nfSetIdList3
set3.smfset.5gc.mnc012.mcc345

 global_Var_pcf_nfSetIdList1: &global_Var_pcf_nfSetIdList1
set1.pcfset.5gc.mnc012.mcc345
 global_Var_pcf_nfSetIdList2: &global_Var_pcf_nfSetIdList2
set2.pcfset.5gc.mnc012.mcc345
 global_Var_pcf_nfSetIdList3: &global_Var_pcf_nfSetIdList3
set3.pcfset.5gc.mnc012.mcc345

 global_Var_udm1_nfInstanceId: &global_Var_udm1_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111111
 global_Var_udm2_nfInstanceId: &global_Var_udm2_nfInstanceId 21111111-aaaa-
aaaa-aaaa-111111111111
 global_Var_udm3_nfInstanceId: &global_Var_udm3_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111122
 global_Var_smf1_nfInstanceId: &global_Var_smf1_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111111
 global_Var_smf2_nfInstanceId: &global_Var_smf2_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111122
 global_Var_smf3_nfInstanceId: &global_Var_smf3_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111133
 global_Var_smf4_nfInstanceId: &global_Var_smf4_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111144
 global_Var_smf5_nfInstanceId: &global_Var_smf5_nfInstanceId 11111111-aaaa-
aaaa-aaaa-111111111155
 global_Var_pcf1_nfInstanceId: &global_Var_pcf1_nfInstanceId
1faf1bbc-6e4a-3994-a507-a14ef8e1bc5a
 global_Var_pcf2_nfInstanceId: &global_Var_pcf2_nfInstanceId
1faf1bbc-6e4a-3994-a507-a14ef8e1bc6b
 global_Var_scp51_nfInstanceId: &global_Var_scp51_nfInstanceId
2fbf1bbc-6e4b-3994-b507-b14ef8e1bc51
 global_Var_scp61_nfInstanceId: &global_Var_scp61_nfInstanceId
2fbf1bbc-6e4b-3994-b507-b14ef8e1bc61

 # If svc name has to be changed then stub has to be deployed with the
 same svc name before running ATS
 global_Var_udm1_svc_name: &global_Var_udm1_svc_name udm1svc
 global_Var_udm2_svc_name: &global_Var_udm2_svc_name udm2svc
 global_Var_udm3_svc_name: &global_Var_udm3_svc_name udm3svc
 global_Var_smf1_svc_name: &global_Var_smf1_svc_name smf1svc
 global_Var_smf2_svc_name: &global_Var_smf2_svc_name smf2svc
 global_Var_smf3_svc_name: &global_Var_smf3_svc_name smf3svc
 global_Var_smf4_svc_name: &global_Var_smf4_svc_name smf4svc
 global_Var_smf5_svc_name: &global_Var_smf5_svc_name smf5svc
 global_Var_pcf1_svc_name: &global_Var_pcf1_svc_name pcf1svc
 global_Var_pcf2_svc_name: &global_Var_pcf2_svc_name pcf2svc
 global_Var_scp51_svc_name: &global_Var_scp51_svc_name scp51svc
 global_Var_scp61_svc_name: &global_Var_scp61_svc_name scp61svc
 global_Var_nrf1_svc_name: &global_Var_nrf1_svc_name nrf1svc
#END_GLOBAL

```

## Feature File

The <Feature\_name>.yaml file contains feature-specific variables that can be parameterized. For example, if the name of the feature file that needs to be parameterized is "ModelC\_NF\_Set.feature", then the YAML file corresponding to it will be named ModelC\_NF\_Set.yaml. This yaml file contains the #START\_GLOBAL and #END\_GLOBAL tags without any data in between, as the data is copied over from the global.yaml file to this section during test execution. The variable name in feature.yaml should have feature\_Var\_ as prefix.

For example:- <key\_name>: &<variable\_name> <default\_value\_of\_variable>

```
#START_GLOBAL
#END_GLOBAL

ModelC_NF_Set.feature:

 File_Parameters:

 feature_Var_udm1_Priority:
&feature_Var_udm1_Priority 0
 feature_Var_smf1_Priority:
&feature_Var_smf1_Priority 0
 feature_Var_traffic_rate:
&feature_Var_traffic_rate 100
```

### Scenario

The variables are referenced under the scenario tag to use in the feature file; for this, the scenario tag has to be concatenated with Scenario\_ and enclosed within double commas.

The variables defined under the scenario tag should have sc\_ as its prefix.

For example:- "Scenario\_<Scenario\_tag>":

```
"Scenario_Scenario-1- <Scenario_tag>":
 Input:
 File_Parameters:
 sc_http_requests_total_udm1: 100
```

Sample:

```
#START_GLOBAL
#END_GLOBAL

ModelC_NFSet.feature:
 File_Parameters:
 # The priorities can be changed without having the impact on the order of
the priorities
 feature_Var_udm1_Priority: &feature_Var_udm1_Priority 0
 feature_Var_udm2_Priority: &feature_Var_udm2_Priority 1
 feature_Var_udm3_Priority: &feature_Var_udm3_Priority 1

 feature_Var_smf1_Priority: &feature_Var_smf1_Priority 0
 feature_Var_smf2_Priority: &feature_Var_smf2_Priority 0
 feature_Var_smf3_Priority: &feature_Var_smf3_Priority 1
 feature_Var_smf4_Priority: &feature_Var_smf4_Priority 3
 feature_Var_smf5_Priority: &feature_Var_smf5_Priority 4
```

```

feature_Var_pcf1_Priority: &feature_Var_pcf1_Priority 0
feature_Var_pcf2_Priority: &feature_Var_pcf2_Priority 0

feature_Var_supiOfPathURI: &feature_Var_supiOfPathURI imsi-1000000001

feature_Var_traffic_rate: &feature_Var_traffic_rate 100

The traffic for the below metrics to be configured in the same
proportionate as the traffice sent
feature_Var_scp_http_rx_req_total_cnt: *feature_Var_traffic_rate
feature_Var_scp_http_tx_req_total_cnt: *feature_Var_traffic_rate
feature_Var_scp_http_tx_req_total_cnt_alternate_route:
&feature_Var_scp_http_tx_req_total_cnt_alternate_route 200
feature_Var_http_requests_total_udm1:
&feature_Var_http_requests_total_udm1 0
feature_Var_http_requests_total_udm2:
&feature_Var_http_requests_total_udm2 0
feature_Var_http_requests_total_udm3:
&feature_Var_http_requests_total_udm3 0
feature_Var_http_requests_total_udm4:
&feature_Var_http_requests_total_udm4 0
feature_Var_http_requests_total_smf1:
&feature_Var_http_requests_total_smf1 0
feature_Var_http_requests_total_smf2:
&feature_Var_http_requests_total_smf2 0
feature_Var_http_requests_total_smf3:
&feature_Var_http_requests_total_smf3 0
feature_Var_http_requests_total_smf4:
&feature_Var_http_requests_total_smf4 0
feature_Var_http_requests_total_pcf1:
&feature_Var_http_requests_total_pcf1 0
feature_Var_scp_http_rx_res_total_cnt: *feature_Var_traffic_rate
feature_Var_scp_http_tx_res_total_cnt: *feature_Var_traffic_rate

"Scenario_Scenario-1- Forward route initial UECM AMF Registration messages
with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header":
 Input:
 File_Parameters:
 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-2- Alternate route initial UECM AMF Registration
messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid
Header":
 Input:
 File_Parameters:
 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent
 sc_http_requests_total_udm2: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-3- Load Balance initial UECM AMF Registration messages
with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header":
 Input:
 File_Parameters:
 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

```



```

 sc_http_requests_total_udm2: 50 #To be configured in the same
proportionate as the traffice sent
 sc_http_requests_total_udm3: 50 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-4- Alternate route initial UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot missing and 3GPP-Sbi-Disocvery-target-NfSetid Header present":

Input:

File\_Parameters:

```

 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-5- Forward route subsequent UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding Header with bl=nfset":

Input:

File\_Parameters:

```

 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-6- Alternate route subsequent UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding Header with bl=nfset":

Input:

File\_Parameters:

```

 sc_udm3_Priority: 30 #Priority can be changed without having
the impact on the order of the priorities
 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent
 sc_http_requests_total_udm2: 100 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-7- Load Balance subsequent UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding Header with bl=nfset":

Input:

File\_Parameters:

```

 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent
 sc_http_requests_total_udm2: 50 #To be configured in the same
proportionate as the traffice sent
 sc_http_requests_total_udm3: 50 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-8- Alternate route subsequent UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot missing and 3gpp-sbi-routing-binding Header with bl=nfset is present":

Input:

File\_Parameters:

```

 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

```

"Scenario\_Scenario-9- To test when Forward route for notification request UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-

routing-binding header bl=nfset":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-10- To test when Forward route fails for notification request UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding header,alternate route should happen on NfSet":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent

sc\_http\_requests\_total\_udm2: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-11- To test Forward route for notification request UECM AMF Registration messages with 3gpp-Sbi-Discovery-target-nf-set-id,3gpp-Sbi-Target-apiRoot":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-12- To test when Forward route fails for notification request UECM AMF Registration messages with 3gpp-Sbi-Discovery-target-nf-set-id and 3gpp-Sbi-Target-apiRoot,alternate route should happen on NfSet":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent

sc\_http\_requests\_total\_udm2: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-13- To test when Forward route fails for notification request UECM AMF Registration messages with 3gpp-Sbi-Discovery-target-nf-set-id and 3gpp-Sbi-Target-apiRoot,load balancing should happen on NfSet on NFs with similar priority":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent

sc\_http\_requests\_total\_udm2: 50 #To be configured in the same proportionate as the traffice sent

sc\_http\_requests\_total\_udm3: 50 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-14- Forward route initial SMF PduSession sm-contexts create messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Disocvery-target-NfSetid Header":

Input:

File\_Parameters:

sc\_http\_requests\_total\_smf1: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-15- Alternate route SMF PduSession sm-contexts create

messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header":

Input:

File\_Parameters:

sc\_http\_requests\_total\_smf1: 100 #To be configured in the same proportionate as the traffice sent  
sc\_http\_requests\_total\_smf2: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-16- Error response code received when SMF PduSession sm-contexts create message is sent with missing information in 3gpp-sbi-routing-binding header and 3gpp-Sbi-Target-apiRoot header is missing":

Input:

File\_Parameters:

sc\_scp\_http\_tx\_req\_total\_cnt: 0 #To be configured in the same proportionate as the traffice sent  
sc\_ocscp\_metric\_scp\_generated\_response\_total: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-17-Error response code received when SMF PduSession sm-contexts create message is sent with missing information in 3gpp-Sbi-Discovery-target-nf-set-id header and 3gpp-Sbi-Target-apiRoot header":

Input:

File\_Parameters:

sc\_targetNfSetId\_Send: 1  
sc\_http\_requests\_total\_smf1: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-18- Alternate route SMF PduSession sm-contexts create messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header":

Input:

File\_Parameters:

sc\_smf1\_Priority: 2 #Priority can be changed without having the impact on the order of the priorities  
sc\_scp\_http\_tx\_req\_total\_cnt\_alternate\_route: 400 #To be configured in the same proportionate as the traffice sent  
sc\_http\_requests\_total\_smf1: 100 #To be configured in the same proportionate as the traffice sent  
sc\_http\_requests\_total\_smf4: 100 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-19- No Alternate route for initial UECM AMF Registration messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header as reroute Policy is disabled":

Input:

File\_Parameters:

sc\_http\_requests\_total\_udm1: 100 #To be configured in the same proportionate as the traffice sent  
sc\_scp\_http\_rx\_res\_total\_cnt: 0 #To be configured in the same proportionate as the traffice sent

"Scenario\_Scenario-20- Forward route PCF SMPolicyControl Create SMPolicyAssociation with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Discovery-target-NfSetid Header and verify that 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-producer-id headers are not present in response":

```

Input:
 File_Parameters:
 sc_pcf1_Service_Priority: 80 #Priority can be changed
without having the impact on the order of the priorities
 sc_http_requests_total_pcf1: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-21- Alternate route PCF SMPolicyControl Create
SMPolicyAssociation messages with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-
Disocvery-target-NfSetid Header and verify that 3gpp-Sbi-Target-apiRoot and
3gpp-sbi-producer-id headers are present in response":
 Input:
 File_Parameters:
 sc_pcf1_Priority: 2 #Priority can be changed without having the
impact on the order of the priorities
 sc_http_requests_total_pcf2: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-22- Alternate route initial UECM AMF Registration
messages with 3gpp-Sbi-Target-apiRoot missing and 3GPP-Sbi-Disocvery-target-
NfSetid Header present and verify that only 3gpp-sbi-producer-id header is
present in response since location header is present":
 Input:
 File_Parameters:
 sc_http_requests_total_udm1: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-23- Alternate route PCF SMPolicyControl Create messages
with 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Disocvery-target-NfSetid Header and
verify that 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-producer-id headers are
present in response":
 Input:
 File_Parameters:
 sc_pcf1_Priority: 2 #Priority can be changed without having the
impact on the order of the priorities
 sc_http_requests_total_pcf2: 100 #To be configured in the same
proportionate as the traffice sent

"Scenario_Scenario-24- Alternate route PCF SMPolicyControl Create messages
with 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding Header and verify
that 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-producer-id headers are present in
response":
 Input:
 File_Parameters:
 sc_Var_pcf1_Priority: 2 #Priority can be changed without having
the impact on the order of the priorities
 sc_http_requests_total_pcf2: 100 #To be configured in the same
proportionate as the traffice sent

```

### Updates in Feature Files

The variables are defined in either global.yaml or <feature\_name>.yaml is used in the feature files by enclosing the variable name in curly brackets. The steps in a feature file without parameterization are as follows:

**Figure 3-17 Feature File without Parameterization**

```

Then Update File /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json For Key nfProfile.nfSetIdList.0 With Value set1.udmset.5
gc.mmc012.mcc345
Then Update File /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json For Key nfProfile.nfSetIdList.1 With Value set3.udmset.5
gc.mmc012.mcc345
Then Update Json /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json With Key nfProfile.locality:USEast,nfProfile.nfInstanceId:11111111-aaaa-aaaa-aaaa-111111111111,nfProfile.nfServices.ipEndPoints.port:null,nfProfile.nfServices.fqdn:udm1svc.default.svc,cluster.local,nfProfile.priority:0 For Service udm1svc
Then Soothsayer NF With Serving Scope Register ss2 setnrfl1.nrfset.5gc.mmc012.mcc345 /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json

```

The steps in a feature file when it is parameterized:

**Figure 3-18 Feature File when it is Parameterized**

```

Then Update File /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json For Key nfProfile.nfSetIdList.0 With Value {global_Var_u
dm_nfSetIdList1}
Then Update File /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json For Key nfProfile.nfSetIdList.1 With Value {global_Var_u
dm_nfSetIdList3}
Then Update Json /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg.json With Key nfProfile.nfServices.apiPrefix:{global_Var_SUT_
apiPrefix1},nfProfile.locality:{global_Var_SUT_ServingLocality1},nfProfile.nfInstanceId:{global_Var_udm1_nfInstanceId},nfProfile
.nfServices.ipEndPoints.port:null,nfProfile.nfServices.fqdn:{global_Var_udm1_svc_name}.default.svc.cluster.local,nfProfile.priori
ty:{feature_Var_udm1_Priority} For Service {global_Var_udm1_svc_name}
Then Soothsayer NF With Serving Scope Register ss2 {global_Var_localNrSetId1} /var/lib/jenkins/ocscp_tests/data/udm1_nfset_reg
.json

```

### Running Feature Files

The changes are made in the product config folder, so the same configuration type can be chosen from the Jenkins UI and the test case can be run, and in the logs, it can be seen that the values of variables are being replaced with the ones that are provided in `global.yaml` or `<feature_name>.yaml`.

#### Note

Only variables that bring in some value addition require parameterization, and a change in the value of those variables does not affect the test case. A variable that can have an effect across all the feature files should be kept under `global.yaml`, a variable that is specific to a feature file has to be kept under `<feature_name>.yaml`, and a variable that is related to a specific scenario must be kept under scenario level.

## 3.6.5 Appendix

This section provides supplementary information that may be helpful for a more comprehensive understanding of installing and running SCP test cases in ATS.

### 3.6.5.1 Creating Custom Service Account

By default, ATS will create a service account with the below rules. If the user does not want to use the default service account, then the service account needs to be manually created with the below permission, and the created custom service account name needs to be specified in the `ocats_ocscp_values_24.1.1.yaml` file.

To run SCP-ATS, use the following rules to create a custom service account:

```

rules:
- apiGroups: ["apps"]
 resources: ["deployments", "replicasets"]

```

```

 verbs: ["watch", "get", "list", "create", "delete", "update", "patch"]
- apiGroups: [""]
 resources: ["pods", "services", "pod/logs"]
 verbs: ["watch", "get", "list", "create", "delete"]
- apiGroups: [""]
 resources: ["secrets"]
 verbs: ["get", "list"]
- apiGroups: [""]
 resources: ["configmaps"]
 verbs: ["watch", "get", "list", "create", "delete", "update", "patch"]

```

### 3.6.5.2 Adding New Stubs to SCP-ATS

For adding new stubs related to pystub, add deployment, and service yaml files inside the template folder of ocats-pystub charts and update the deployment name in the stub deployment file as follows:

```
name- .Values.ausf11.deploymentName
```

```
app- .Values.ausf11.deploymentName
```

Update service name, label and selectors, and ports in the service.yaml file.

```
name: {{ .Values.ausf11.service.name }}
```

```
app: {{ .Values.ausf11.deploymentName }}
```

```
port: {{ .Values.ausf11.service.ports.port }}
```

Update the ocats\_ocscp\_values\_24.1.1.yaml file with new information:

```

ausf1:
service:
name: ausf1svc
type: ClusterIP
ports:
port: 8080

```

The images represent pystub service and deployment files for each stubs:

**Figure 3-19 Sample Deployment File**

```

-rw-r--r--. 1 root root 782 Oct 25 14:36 ausf12-service.yaml
-rw-r--r--. 1 root root 2597 Oct 25 14:36 ausf12-deployment.yaml
-rw-r--r--. 1 root root 782 Oct 25 14:36 ausf11-service.yaml
-rw-r--r--. 1 root root 2597 Oct 25 14:36 ausf11-deployment.yaml

```

## 3.7 Installing ATS for SEPP

This section describes Automated Testing Suite (ATS) installation procedures for Security Edge Protection Proxy (SEPP) in a cloud native environment using Continuous Delivery Control Server (CDCS) or Command Line Interface (CLI) procedures:

### 3.7.1 Resource Requirements

#### Total Number of Resources

The resources required to install SEPP-ATS are as follows:

**Table 3-23 Total Number of Resources**

| Resource             | CPUs | Memory(GB) | Storage(GB) |
|----------------------|------|------------|-------------|
| SEPP SUT Total       | 15.1 | 24.128     | 0           |
| DB Tier Total        | 40   | 40         | 20          |
| ATS Total            | 4.5  | 4.5        | 1           |
| Grand Total SEPP ATS | 59.6 | 68.628     | 21          |

#### Resource Details

The details of resources required to install SEPP-ATS are as follows:

**Table 3-24 Resource Details**

| Microservice         | CPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | # Replicas (regular deployment) | # Replicas (ATS deployment) | CPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|----------------------|-----------------------|------------------------------|-----------------------------------|---------------------------------|-----------------------------|-----------------------|------------------------------|-----------------------------------|
| SEPP Pods            |                       |                              |                                   |                                 |                             |                       |                              |                                   |
| n32-ingress-gateway  | 1.5                   | 2                            | 0                                 | 1                               | 1                           | 1.5                   | 2                            | 0                                 |
| n32-egress-gateway   | 1.5                   | 2                            | 0                                 | 1                               | 1                           | 1.5                   | 2                            | 0                                 |
| plmn-ingress-gateway | 1.5                   | 2                            | 0                                 | 1                               | 1                           | 1.5                   | 2                            | 0                                 |
| plmn-egress-gateway  | 1.5                   | 2                            | 0                                 | 1                               | 1                           | 1.5                   | 2                            | 0                                 |
| pn32f-svc            | 1                     | 2                            | 0                                 | 1                               | 1                           | 1                     | 2                            | 0                                 |
| cn32f-svc            | 1                     | 2                            | 0                                 | 1                               | 1                           | 1                     | 2                            | 0                                 |
| cn32c-svc            | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| pn32c-svc            | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| config-mgr-svc       | 1                     | 2                            | 0                                 | 1                               | 1                           | 1                     | 2                            | 0                                 |

Table 3-24 (Cont.) Resource Details

| Microservice                                      | CPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | # Replicas (regular deployment) | # Replicas (ATS deployment) | CPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|---------------------------------------------------|-----------------------|------------------------------|-----------------------------------|---------------------------------|-----------------------------|-----------------------|------------------------------|-----------------------------------|
| nrf-client-nfdiscovery                            | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| nrf-client-nfmanagement                           | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| ocpm-config-server                                | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| appinfo                                           | 0.5                   | 1                            | 0                                 | 1                               | 1                           | 0.5                   | 1                            | 0                                 |
| perfinfo                                          | 0.1                   | 0.128                        | 0                                 | 1                               | 1                           | 0.1                   | 0.128                        | 0                                 |
| nf-mediation                                      | 1                     | 1                            | 0                                 | 1                               | 1                           | 1                     | 1                            | 0                                 |
| alternate-route                                   | 1                     | 1                            | 0                                 | 1                               | 1                           | 1                     | 1                            | 0                                 |
| coherence-svc                                     | 1                     | 2                            | 0                                 | 1                               | 1                           | 1                     | 2                            | 0                                 |
| SEPP SUT Totals                                   |                       |                              |                                   |                                 |                             | 15.1 CPU              | 24.128 GB                    | 0                                 |
| ATS                                               |                       |                              |                                   |                                 |                             |                       |                              |                                   |
| ATS Behave                                        | 3                     | 3                            | 1 (Optional)                      | 1                               | 1                           | 3                     | 3                            | 1                                 |
| ATS Stub (Python)                                 | .5                    | .5                           | 0                                 | 1                               | 1                           | .5                    | .5                           | 0                                 |
| ATS Stub-2 (Python)                               | .5                    | .5                           | 0                                 | 1                               | 1                           | .5                    | .5                           | 0                                 |
| ATS Stub-3 (Python)                               | .5                    | .5                           | 0                                 | 1                               | 1                           | .5                    | .5                           | 0                                 |
| ATS Totals                                        |                       |                              |                                   |                                 |                             | <b>4.5</b>            | <b>4.5</b>                   | <b>1</b>                          |
| DB Tier Pods (minimum of 4 worker nodes required) |                       |                              |                                   |                                 |                             |                       |                              |                                   |
| virt-launcher-dt-1.cluster.local                  | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| virt-launcher-dt-2.cluster.local                  | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| virt-launcher-dt-3.cluster.local                  | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |



Table 3-24 (Cont.) Resource Details

| Microservice                            | CPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | # Replicas (regular deployment) | # Replicas (ATS deployment) | CPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|-----------------------------------------|-----------------------|------------------------------|-----------------------------------|---------------------------------|-----------------------------|-----------------------|------------------------------|-----------------------------------|
| vrt-launcher-dt-4.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-mt-1.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-mt-2.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-mt-3.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-sq-1.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-sq-2.cluster.local         | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| vrt-launcher-db-installer.cluster.local | 4                     | 4                            | 2                                 | 2                               | 1                           | 4                     | 4                            | 2                                 |
| DB Tier Totals                          |                       |                              |                                   |                                 |                             | 40                    | 40                           | 20                                |

### 3.7.1.1 SEPP ATS Compatibility Matrix

The following table lists the versions SEPP ATS and the compatibility with the SEPP and ATS framework:

Table 3-25 SEPP ATS Compatibility Matrix

| SEPP ATS Release | SEPP Release | ATS Framework version |
|------------------|--------------|-----------------------|
| 24.1.0           | 24.1.0       | 24.1.1                |
| 23.4.0           | 23.4.0       | 23.4.0                |
| 23.3.1           | 23.3.1       | 23.3.1                |
| 23.3.0           | 23.3.0       | 23.3.0                |
| 23.2.1           | 23.2.1       | 23.2.2                |
| 23.2.0           | 23.2.0       | 23.2.0                |

## 3.7.2 Downloading the ATS Package

### Locating and Downloading ATS and Simulator Images

To locate and download the ATS Image from MOS:

1. Log in to [My Oracle Support](#) with your credentials.
2. Select the **Patches and Updates** tab to locate the patch.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Core Security Edge Protection Proxy* <release\_number> from **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required ATS patch from the search results. The Patch Details window appears.
8. Click **Download**. The File Download window appears.
9. Click the <p\*\*\*\*\*>\_<release\_number>\_Tekelec>.zip file to download the CNC SEPP ATS package file.
10. Untar the zip file to access all the ATS Images. The ocsepp-ats directory has the following files:  
The csar directory has following files:

```
ocats_ocsepp_csar_24_1_0_0_0.zip
ocats_ocsepp_csar_24_1_0_0_0.zip.sha256
```

#### Note

The above zip file contains all the images and custom values required for 24.1.0 release of OCATS-OCSEPP.

11. The ocats\_ocsepp\_csar\_24\_1\_0\_0\_0.zip file has following definitions:

```

— Definitions
 — ocats_ocsepp_cne_compatibility.yaml
 — ocats_ocsepp.yaml
— Files
 — ChangeLog.txt
 — Helm
 — ocats-sepp-24.1.0.tgz (Helm Charts)
 — Licenses
 — ocats-sepp-24.1.0.tar (BDD client image)
 — Oracle.cert
 — seppstub-24.1.0.tar (Stub server image)
 — Tests
— Scripts/
 — ocats_ocsepp_tests_jenkinsjobs_24.1.0.tgz
 — jobs (For Persistent Volume)
 — ocsepp_tests (For Persistent Volume)
```

```

| └── ocats_ocsepp_values_24.1.0.yaml (Custom values file for
installation)
| └── ocats_ocsepp.mf
| ├── TOSCA-Metadata
| └── TOSCA.meta

```

12. Copy the zip file to Kubernetes cluster where you want to deploy ATS

### 3.7.3 Pushing the Images to Customer Docker Registry

#### Preparing to Deploy ATS and Stub Pod in Kubernetes Cluster

To deploy ATS and Stub Pod in Kubernetes Cluster:

1. Run the following command to extract tar file content:

```
unzip ocats_ocsepp_csar__24_1_0_0_0.zip
```

The following docker image tar files are located at **Files** folder:

- ocats-sepp-24.1.0.tar
  - seppstub-24.1.0.tar
2. Run the following commands in your cluster to load the ATS docker image, 'ocats-sepp-24.1.0.tar' and Stub docker image seppstub-24.1.0.tar, and push it to your registry.

```
$ docker load -i ocats-sepp-24.1.0.tar
$ docker load -i seppstub-24.1.0.tar
```

```
$ docker tag ocats/ocats-sepp:24.1.0 <local_registry>/ocats/ocats-sepp:24.1.0
```

```
$ docker tag ocats/seppstub:24.1.0 <local_registry>/ocats/seppstub:24.1.0
```

```
$ docker push <local_registry>/ocats/ocats-sepp:24.1.0
```

```
$ docker push <local_registry>/ocats/seppstub:24.1.0
```

3. Run the following command to get the helm charts are located in Helm directory of files folder:

```
tar -xvf ocats-sepp-24.1.0.tgz
```

The output of this command is:

```
ocats-sepp/
ocats-sepp/Chart.yaml
ocats-sepp/charts/
ocats-sepp/values.yaml
```

4. Create a copy of the custom values located at **Scripts/ocats\_ocsepp\_values\_24.1.0.yaml** and update it for image name, tag and other parameters as per the requirement.

## 3.7.4 Creating Secrets and Support for TLSv1.2 and TLSv1.3

### 3.7.4.1 Configuring Root Certificate

The following are the steps to configure the root certificate (caroot.cer):

#### ① Note

- Use the same root certificate (caroot.cer) and key which is used for creating the SEPP certificates.
- If the user has already generated the caroot.cer and cakey.pem file while deploying SEPP then user can skip to Generate ATS certificate step.
- Both the ATS and SEPP must have same root certificate.

1. In case caroot.cer is not available for SEPP, Create ssl.conf file for SEPP using the following format:

```
Creation of CSEPP Certs, Fqdn should be changed

#ssl.conf
[req]default_bits = 4096
distinguished_name = req_distinguished_name
req_extensions = req_ext
[req_distinguished_name]
countryName = Country Name (2 letter code)
countryName_default = IN
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = Karnataka
localityName = Locality Name (eg, city)
localityName_default = Bangalore
organizationName = Organization Name (eg, company)
organizationName_default = Oracle
commonName = sepp1.inter.oracle.com
commonName_max = 64
commonName_default = sepp1.inter.oracle.com
[req_ext]
subjectAltName = @alt_names
[alt_names]
IP = 127.0.0.1
DNS.1 = sepp1.inter.oracle.com
```

2. Set the following environment variables:

```
export PEM_PHRASE=NextGen1
export DEPLOYMENT_NAMESPACE=sepp1
```

3. Run the following command to create the required files:

```

openssl req -new -keyout cakey.pem -out careq.pem -passout pass:${
{PEM_PHRASE}} -subj "/C=IN/ST=Karnataka/L=Bangalore/O=Oracle/
CN=sepp1.inter.oracle.com/emailAddress=xyz@oracle.com"

openssl x509 -signkey cakey.pem -req -days 3650 -in careq.pem -out
caroot.cer -extensions v3_ca -passin pass:${{PEM_PHRASE}}

openssl req -x509 -nodes -sha256 -days 365 -newkey rsa:2048 -keyout
rsa_private_key -out rsa_certificate.crt -subj '/C=IN/ST=Karnataka/
L=Bangalore/O=Oracle/CN=sepp1.inter.oracle.com/emailAddress=xyz@oracle.com'

openssl rsa -in rsa_private_key -outform PEM -out rsa_private_key_pkcs1.pem

openssl req -new -key rsa_private_key -out ocsepp.csr -config ssl.conf -
subj '/C=IN/ST=Karnataka/L=Bangalore/O=Oracle/CN=sepp1.inter.oracle.com/
emailAddress=xyz@oracle.com'

openssl x509 -CA caroot.cer -CAkey cakey.pem -CAserial serial.txt -req -in
ocsepp.csr -out ocsepp.cer -days 365 -extfile ssl.conf -extensions req_ext
-passin pass:${{PEM_PHRASE}}

openssl ecparam -genkey -name prime256v1 -noout -out ec_private_key.pem

openssl pkcs8 -topk8 -in ec_private_key.pem -inform pem -out
ecdsa_private_key.pem -outform pem -nocrypt

openssl req -new -key ecdsa_private_key.pem -x509 -nodes -days 365 -out
ecdsa_certificate_pkcs1.crt -subj '/C=IN/ST=Karnataka/L=Bangalore/O=Oracle/
CN=sepp1.inter.oracle.com/emailAddress=xyz@oracle.com'

openssl req -new -key ecdsa_private_key.pem -out ecdsa_certificate.csr -
subj '/C=IN/ST=Karnataka/L=Bangalore/O=Oracle/CN=sepp1.inter.oracle.com/
emailAddress=xyz@oracle.com'

echo NextGen1 > trust.txt
echo NextGen1 > key.txt
echo 1234 > serial.txt

```

### 3.7.4.2 Generating ATS Certificate

The following are the steps to configure the ATS certificate:

1. Create and edit the ssl.conf file as follows:

**Note**

- While trying to access the GUI with DNS, ensure that the common Name\_default is the same as the DNS name being used.
- Ensure that the DNS is in the format <service\_name>.<namespace>.<cluster\_domain>
- The user can add multiple DNS like DNS.1, DNS.2 so on.
- The ATS\_HELM\_RELEASE\_NAME is the release name which will be used to deploy ATS.

- In the alt\_names section of the ssl.conf, IPs has to be listed down through which ATS GUI will be opened. We can add multiple IPs like IP.1, IP.2 so on.
- All stubserver service names (({ATS\_HELM\_RELEASE\_NAME}-stubserver.{ats-namespace}, {ATS\_HELM\_RELEASE\_NAME}-stubserver-2.{ats-namespace}, {ATS\_HELM\_RELEASE\_NAME}-stubserver-3.{ats-namespace})) must be in Subject Alternative Name in certificate.
- Update bddclient service name (\${ATS\_HELM\_RELEASE\_NAME}-bddclient.\${DEPLOYMENT\_NAMESPACE}.svc.cluster.local ) in the commonName, commonName\_default and DNS name in alt\_names section.

**Sample code:**

```
#ssl.conf
[req]
default_bits = 4096
distinguished_name = req_distinguished_name
req_extensions = req_ext

[req_distinguished_name]
countryName = Country Name (2 letter code)
countryName_default = IN
stateOrProvinceName = State or Province Name (full name)
stateOrProvinceName_default = Karnataka
localityName = Locality Name (eg, city)
localityName_default = Bangalore
organizationName = Organization Name (eg, company)
organizationName_default = Oracle
commonName = ${ATS_HELM_RELEASE_NAME}-bddclient.${DEPLOYMENT_NAMESPACE}.svc.cluster.local
commonName_max = 64
commonName_default = ${ATS_HELM_RELEASE_NAME}-bddclient.${DEPLOYMENT_NAMESPACE}.svc.cluster.local

[req_ext]
subjectAltName = @alt_names

[alt_names]
IP.1 = 127.0.0.1
IP.2 = 10.75.217.5

#Mandatory values
```

```
DNS.1 = ${ATS_HELM_RELEASE_NAME}-bddclient.$
${DEPLOYMENT_NAMESPACE}.svc.cluster.local
DNS.2 = ${ATS_HELM_RELEASE_NAME}-stubserver.${DEPLOYMENT_NAMESPACE}
DNS.3 = ${ATS_HELM_RELEASE_NAME}-stubserver-2.${DEPLOYMENT_NAMESPACE}
DNS.4 = ${ATS_HELM_RELEASE_NAME}-stubserver-3.${DEPLOYMENT_NAMESPACE}
DNS.5 = localhost
```

2. Run the following command to create a certificate signing request or csr:

```
$ openssl req -config ssl.conf -newkey rsa:2048 -days 1000 -nodes -keyout
rsa_private_key_pkcs1.key > ssl_rsa_certificate.csr
```

Output:

```
Ignoring -days; not generating a certificate
Generating a RSA private key
...+++++
.....+++++
writing new private key to 'rsa_private_key_pkcs1.key'

You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [IN]:
State or Province Name (full name) [KA]:
Locality Name (eg, city) [BLR]:
Organization Name (eg, company) [ORACLE]:
Common Name (e.g. server FQDN or YOUR name) [ocats]:
[cloud-user@star23-bastion-1 ocats]$
```

3. Run the following command to verify whether all configurations are done:

```
openssl req -text -noout -verify -in ssl_rsa_certificate.csr
```

4. Run the following command to sign the certificate signing request or csr file with root certificate:

```
$ openssl x509 -extfile ssl.conf -extensions req_ext -req -in
ssl_rsa_certificate.csr -days 1000 -CA caroot.cer -CAkey cakey.pem -
set_serial 04 > ssl_rsa_certificate.crt
```

Output:

```
Signature ok
subject=C = IN, ST = KA, L = BLR, O = ORACLE, CN = sepp-ats-rel-
bddclient.testns.svc.cluster.local
Getting CA Private Key
[cloud-user@star23-bastion-1 ocats]$
```

**Note**

When the output prompts for the current password, enter the password was used to create `createkey.pem` file.

5. Verify whether the certificate is properly signed by root certificate:

```
$ openssl verify -CAfile caroot.cer ssl_rsa_certificate.crt
```

Output:

```
ssl_rsa_certificate.crt:OK
```

6. For then jenkins to support GUI access through https, a jks file has to be created. Perform the following steps to generate jks file for Jenkins Server:

- a. Run the following command to generate the .p12 keystore file:

```
$ openssl pkcs12 -inkey rsa_private_key_pkcs1.key -in
ssl_rsa_certificate.crt -export -out certificate.p12
```

Output:

Enter Export Password:

Verifying - Enter Export Password:

**Note**

When the output prompts for the password, enter the password and note it down, as it is required for creating the jks file.

7. Run the following command to convert .p12 file in jks format file to be used in Jenkins Server:

**Note**

- Ensure to use the same password used for creating jks file and for creating the .p12 file.
- Java should be pre-installed to run the keytool utility.

```
$ keytool -importkeystore -srckeystore ./certificate.p12 -srcstoretype
pkcs12 -destkeystore jenkinsserver.jks -deststoretype JKS
```



Output:

```
Importing keystore ./certificate.p12 to jenkinsserver.jks...
Enter destination keystore password:
Re-enter new password:
Enter source keystore password:
Entry for alias 1 successfully imported.
Import command completed: 1 entries successfully imported, 0 entries
failed or cancelled
```

The generated file, `jenkinsserver.jks` must be given to the jenkins server.

### 3.7.4.3 Creating ATS Secret and Configuring Helm chart

The following are the steps to create ATS Secret and configure the Helm chart:

#### Note

The user can decide whether to use the generated CA signed certificate or self-signed certificate.

1. Run the following command to create secret:

```
kubectl create secret generic ocats-sepp-secret --from-
file=jenkinsserver.jks --from-file=ssl_rsa_certificate.crt
--from-file=rsa_private_key_pkcs1.key --from-file=caroot.cer -n
<deployment_namespace>
```

2. Run the following command to verify the secret:

```
$ kubectl describe secret ocats-sepp-secret -n testns
```

Output:

```
Name: ocats-sepp-secret
Namespace: testns
Labels: <none>
Annotations: <none>

Type: Opaque

Data
====
caroot.cer: 1147 bytes
ssl_rsa_certificate.crt: 1424 bytes
jenkinsserver.jks: 2357 bytes
rsa_private_key_pkcs1.key: 1675 bytes
```

### Changes to the Helm charts:

The following changes must be updated in the `ocats_ocsepp_values_<version>.yaml` file:

- The Helm parameter `atsGuiTLSEnabled` must set to true for ATS to get the certificates and support HTTPS for GUI. If the user does not want to open ATS GUI in HTTPS mode, then `atsGuiTLSEnabled` flag should be set to false.

```
atsGuiTLSEnabled: false
```

- The Helm parameter `atsCommunicationTLSEnabled` must set to true for necessary context variable to be created which can be later used to communicate with other services in HTTPS.
  - For non-asm deployment `atsCommunicationTLSEnabled` flag should be set to true.
  - For asm deployment `atsCommunicationTLSEnabled` flag should be set to false.

```
atsCommunicationTLSEnabled: true #If set to true, ATS will get necessary
variables to communicate with SUT, Stub
or other NFs with TLS enabled. It is not required in ASM
environment.
```

- The certificates section of the `bddclient` and `stubserver` section of the `ats` custom values file must be updated as follows:

```
certificates:
 cert_secret_name: "ocats-sepp-secret"
 a_cert: "caroot.cer"
 client_cert: "ssl_rsa_certificate.crt"
 private_key: "rsa_private_key_pkcs1.key"
```

```
This parameter is needed when atsGuiTLSEnabled is set to true. This
file is necessary for ATS GUI to be opened with secured TLS protocol. The
file caroot.pem, used during creation of jks file needs to be passed for
Jenkins/ATS API communication.
jks_file: "jenkinsserver.jks"
```

```
jks_password: "123456" #This is the password given to the jks file
while creation.
```

Add caroot certificate in browser to access the ATS GUI. The `caroot.cer` file certificate created by us will be added to Truststore, in this case the browser.

The following are the steps to add caroot certificate in the browser to access the ATS GUI, either Mozilla Firefox or Chrome:

#### Note

Future versions of these browsers may involve different menu options. For more information on importing root certificate, see the browser documentation to add a self-signed certificate to the browser as a trusted certificate.

1. In the Chrome browser, navigate to the settings and search for certificates.
2. Click the **security** option that appears next to search.
3. Click the **Manage Device Certificate** option. The Keychain Access window opens.
4. Search the tab certificate and drag and drop the downloaded caroot certificate.
5. Find the uploaded certificate in the list, usually listed by a temporary name.
6. Double click the certificate and expand the Trust option.
7. In **When using this certificate** option, assign it to "always trust".
8. Close the window and validate if it asks for the password.
9. Save and restart the browser.
1. In the Mozilla Firefox browser, navigate to the settings and search for certificates.
2. Click the **View Certificate** that appears next to search. This opens a Certificate Manager window.
3. Navigate to the Authorities section, click the **Import** button, and upload the caroot certificate.
4. Click the **Trust** options in the pop-up window and click OK.
5. Save and restart the browser.

### 3.7.4.4 Create ATS Health Check secret

To enable the ATS health check pipeline, the following configurations needs to be updated in the `ocats_ocsepp_values_<version>.yaml` file:

#### Non OCI Environment

1. The following parameters needs to be updated in the base64 encoded version format, `occnehostip`, `occnehostusername`, `occnehostpassword`, `envtype`.
2. On installing ATS, health check secret will be created, and the health check pipeline will be shown in the ATS GUI. If `healthcheck` parameter is set to false, then the health check pipeline will not be visible in the ATS GUI.

```
bddclient
atsFeatures:
 healthcheck: true

 Webscale: false
 healthchecksecretname: "healthchecksecret"
 occnehostip: "" # $(echo -n '10.75.217.42' | base64), where occne host ip
needs to be provided
 occnehostusername: "" # $(echo -n 'cloud-user' | base64), where occne host
username needs to be provided
 occnehostpassword: "" # $(echo -n '*****' | base64), where password of host
needs to be provided
 envtype: "" # $(echo -n 'OCCNE' | base64), where occne keyword needs to be
provided
```

## OCI Environment

For key based health check support in OCI, refer to [ATS Health Check](#) section under the ATS Framework Features.

1. The following parameters needs to be updated in the base64 encoded version in the `ocats_ocsepp_values_<version>.yaml` file:

```
bddclient
atsFeatures:
 healthcheck: true

 envtype: "" # $(echo -n 'OCCNE' | base64), Where occne keyword needs to be
provided

 ociHealthCheck:
 passwordAuthenticationEnabled: false
 bastion:
 ip: "" # $(echo -n '10.75.217.42' | base64), Where occne host ip needs
to be provided
 username: "" # $(echo -n '10.75.217.42' | base64), Where occne host ip
needs to be provided
 password: ""
 operatorInstance:
 ip: "" # $(echo -n '10.75.217.42' | base64), Where occne host ip needs
to be provided
 username: "" # $(echo -n '10.75.217.42' | base64), Where occne host ip
needs to be provided
 password: ""
```

## 3.7.5 Configuring ATS

This section describes how to configure ATS for SEPP.

### 3.7.5.1 Enabling Aspen Service Mesh

To enable Aspen Service Mesh (ASM) for ATS, complete the following procedure:

#### Note

By default, this feature is disabled.

1. If ASM is not enabled on the global level for the namespace, run the following command before deploying ATS:

```
kubectl label --overwrite namespace <namespace_name> istio-
injection=enabled
```

Example:

```
kubectl label --overwrite namespace seppsvc istio-injection=enabled
```

2. Add the following annotations in the **BDD client** section in the `ocats-sepp-custom-values.yaml` file to support ASM:  
To enable or disable the ASM, update the `asmEnabled` flag to true or false.

```
asmEnabled: false
asm:
 configMgrPort: 9092
 stubServerPort: 8080
 plmnIgwPort: 80
 n32IgwPort: 80
```

3. Add the following value in the **stub-server** section in the `ocats-sepp-custom-values.yaml` file to support ASM:  
To enable or disable the ASM, update the `asmEnabled` flag to true or false.

```
asmEnabled: false
```

4. For asm deployment, set the `atsCommunicationTLSEnabled` flag to false as given below:

```
atsCommunicationTLSEnabled: false
#If set to true, ATS will get necessary variables to communicate with
SUT, Stub or other NFs with TLS enabled. It is not required in ASM
environment.
```

5. (Optional) The user can configure the resources assigned to the Aspen Mesh (istio-proxy) sidecars in `ocats_ocsepp_values_24.1.0.yaml` file sections as follows:

```
asm:
istioResources:
 limits:
 cpu: 2
 memory: 1Gi
 requests:
 cpu: 100m
 memory: 128Mi
```

**Note**

It is recommended to use the default values of the resources.

**Note**

- If the SEPP is deployed with ASM enabled and user disables the ASM on the global level, then user needs to redeploy the setup to work without ASM.
- At present, SEPP is not supporting the mediation in ASM environment. If user has deployed the mediation service, run the following command before triggering the ATS suite to delete the mediation service:

```
kubectl delete svc <release-name>-nf-mediation -n <namespace>
kubectl delete deploy <release-name>-nf-mediation -n
<namespace>
```

### 3.7.5.2 Enabling Static Port

To enable static port:

**Note**

ATS supports static port. By default, this feature is disabled.

- In the `ocats_ocsepp_values_23.4.0.yaml` file under service section, set the **staticNodePortEnabled** parameter value to 'true' and **staticNodePort** parameter value with valid nodePort.

```
service:
 customExtension:
 labels: {}
 annotations: {}
 type: LoadBalancer
 ports:
 https:
 port: "8443"
 staticNodePortEnabled: false
 staticNodePort: ""
 http:
 port: "8080"
 staticNodePortEnabled: false
 staticNodePort: ""
```

### 3.7.5.3 Enabling Roaming Hub Mode

SEPP ATS supports two types deployment modes:

- SEPP
- Roaming Hub or Hosted SEPP Mode

Flag has been introduced to select the deployment mode.

#Flag to enable Roaming Hub mode

RHenabled: True

### 3.7.5.4 Enabling Hosted SEPP Mode

The Hosted SEPP Mode can be enabled as follows:

#Flag to enable Hosted SEPP mode

RHenabled: True

#### Customizing Error Code Variable in Hosted SEPP Mode

For handling failure scenarios in Hosted SEPP mode, the following customized error code variable has been introduced:

#Customized error code variable for Hosted SEPP

HSErrCode: "400"

### 3.7.5.5 Configuring Egress Rate Limiting Feature

If the Egress Rate Limiting feature is enabled in SEPP deployment, to run the Egress Rate Limiter test cases in ATS, `EgressRateLimiterFlag` parameter has been introduced in the `ocats-sepp-custom-values.yaml` file. If the Egress Rate Limiting feature is disabled in SEPP deployment, ensure to set `EgressRateLimiterFlag` parameter to false.

`EgressRateLimiterFlag: true/false`

For more information about the feature, see the "Rate Limiting for Egress Roaming Signaling per PLMN" section in *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide* and the "Configuration Parameters" section in *Oracle Communications Cloud Native Core, Security Edge Protection Proxy Installation, Upgrade, and Fault Recovery Guide*.

### 3.7.5.6 Configuring Ingress Rate Limiting Feature

If the Ingress rate limiting feature is enabled in SEPP deployment, to run the Ingress rate limiter test cases in ATS, `IngressRateLimiterFlag` parameter has been introduced in the `ocats_ocsepp_values_<version>.yaml` file. If the Ingress rate limiting feature is disabled in SEPP deployment, ensure to set `IngressRateLimiterFlag` parameter to false.

Flag to enable or disable Egress Rate Limiter:

`IngressRateLimiterFlag: true/false`

For more information about the feature, see the "Rate Limiting for Ingress Roaming Signaling per Remote SEPP Set" section in *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide* and the "Configuration Parameters" section in *Oracle*

*Communications Cloud Native Core, Security Edge Protection Proxy Installation, Upgrade, and Fault Recovery Guide.*

### 3.7.5.7 Configuring the Cache Refresh Timeout Value

SEPP supports Refresh ahead cache. To run the ATS cases, *cacheRefreshTimeout* value must be set to 1000 (ms) so that cache can be updated automatically after every test case is executed when ATS suite is triggered.

To set the cache refresh timeout value:

In the *ocsepp-custom-values.yaml* file, under *cn32f-svc* and *pn32f-svc* microservices, set the *cacheRefreshTimeout* parameter value to 1000 (ms).

```
configs:
 cacheRefreshTimeout: 1000 #(ms)
 cacheRefreshInitTimeout: 50000 #(ms)
```

#### Note

If ATS is not configured, the *cacheRefreshTimeout* value must be 30000 (ms).

### 3.7.5.8 Configuring the Topology Cache Refresh Timeout Value

SEPP supports Refresh ahead cache. To run the ATS cases, *topologycacheRefreshTimeout* value must be set to 1000 (ms) so that cache can be updated automatically after every test case is executed when ATS suite is triggered.

In the *ocsepp-custom-values.yaml* file, under *cn32f-svc* and *pn32f-svc* microservices, set the *topologycacheRefreshTimeout* parameter value to 1000 (ms).

```
topologyHiding:
 timerConfig:
 topologyhidingCacheRefreshTimeout: 30000
 topologyhidingCacheRefreshInitTimeout: 50000
 topologyhidingHistoryUpdateTimeout: 30000
 topologyhidingHistoryRefreshSeconds: 60
 config:
 topologyHidingStateCheck: true
```

### 3.7.5.9 Configuring the Security Counter Measure Cache Refresh Timeout



SEPP supports Refresh ahead cache. To run the ATS cases, you must set the value of the `securityCacheRefreshTimeout` parameter to 10 (ms) so that cache can be automatically updated after every test case is run when ATS suite is triggered.

In the `ocsepp-custom-values.yaml` file, under `cn32f-svc` and `pn32f-svc` microservices, set the *Security Counter Measure Cache Refresh Timeout* parameter value to 1000 (ms).

```
configs:
 securityCacheRefreshTimeout: 1000 #(ms)
 securityCacheRefreshInitTimeout: 50000 #(ms)
```

### 3.7.5.10 Configuring the `n32cHandshakePlmnIdListValidationEnabled`

SEPP supports validation of PLMN ID List in `n32c` capability exchange message which can be turned off. To run the ATS cases, `n32cHandshakePlmnIdListValidationEnabled` value must be set to true so that testcases validating PLMN ID List in `n32c` capability exchange message can be run successfully when ATS suite is triggered.

To set the `n32cHandshakePlmnIdListValidationEnabled` value:

In the `ocsepp_custom_values_<version>.yaml` file, under `localProfile`, set the `n32cHandshakePlmnIdListValidationEnabled` parameter value to true.

```
localProfile:
 name: "SEPP-3"
 plmnIdList: [{"mcc": "111", "mnc": "100"}, {"mcc": "111", "mnc": "101"},
{"mcc": "111", "mnc": "102"}, {"mcc": "111", "mnc": "103"}, {"mcc": "111", "mnc": "104"},
{"mcc": "111", "mnc": "105"}, {"mcc": "111", "mnc": "106"}, {"mcc": "111", "mnc": "107"},
{"mcc": "111", "mnc": "108"}, {"mcc": "111", "mnc": "109"}, {"mcc": "111", "mnc": "110"},
{"mcc": "111", "mnc": "111"}, {"mcc": "111", "mnc": "112"}, {"mcc": "111", "mnc": "113"},
{"mcc": "111", "mnc": "114"}, {"mcc": "111", "mnc": "115"}, {"mcc": "111", "mnc": "116"},
{"mcc": "111", "mnc": "117"}, {"mcc": "111", "mnc": "118"}, {"mcc": "111", "mnc": "119"},
{"mcc": "111", "mnc": "120"}, {"mcc": "111", "mnc": "121"}, {"mcc": "111", "mnc": "122"},
{"mcc": "111", "mnc": "123"}, {"mcc": "111", "mnc": "124"}, {"mcc": "111", "mnc": "125"},
{"mcc": "111", "mnc": "126"}, {"mcc": "111", "mnc": "127"}, {"mcc": "111", "mnc": "128"},
{"mcc": "111", "mnc": "129"}]
 # Do not change this value, this will be always true
 sbiTargetApiRootSupported: true
 # Enable PLMN ID List Validation in Exchange Capability Request, Default
 set to true
 n32cHandshakePlmnIdListValidationEnabled: true
 # PLMN ID List Validation Type in Exchange Capability Request, can be
 SUBSET or STRICT only
 n32cHandshakePlmnIdListValidationType: "SUBSET"
```

Update following PLMN ID list in localProfile section for SEPP mode to run the ATS testcases:

```
[{"mcc": "111", "mnc": "100"}, {"mcc": "111", "mnc": "101"}, {"mcc": "111", "mnc": "102"}, {"mcc": "111", "mnc": "103"}, {"mcc": "111", "mnc": "104"}, {"mcc": "111", "mnc": "105"}, {"mcc": "111", "mnc": "106"}, {"mcc": "111", "mnc": "107"}, {"mcc": "111", "mnc": "108"}, {"mcc": "111", "mnc": "109"}, {"mcc": "111", "mnc": "110"}, {"mcc": "111", "mnc": "111"}, {"mcc": "111", "mnc": "112"}, {"mcc": "111", "mnc": "113"}, {"mcc": "111", "mnc": "114"}, {"mcc": "111", "mnc": "115"}, {"mcc": "111", "mnc": "116"}, {"mcc": "111", "mnc": "117"}, {"mcc": "111", "mnc": "118"}, {"mcc": "111", "mnc": "119"}, {"mcc": "111", "mnc": "120"}, {"mcc": "111", "mnc": "121"}, {"mcc": "111", "mnc": "122"}, {"mcc": "111", "mnc": "123"}, {"mcc": "111", "mnc": "124"}, {"mcc": "111", "mnc": "125"}, {"mcc": "111", "mnc": "126"}, {"mcc": "111", "mnc": "127"}, {"mcc": "111", "mnc": "128"}, {"mcc": "111", "mnc": "129"}]
```

Update following PLMN ID list in localProfile section for Roaming Hub mode to run the ATS testcases:

```
[{"mcc": "111", "mnc": "200"}, {"mcc": "111", "mnc": "201"}, {"mcc": "111", "mnc": "202"}, {"mcc": "111", "mnc": "203"}, {"mcc": "111", "mnc": "204"}, {"mcc": "111", "mnc": "205"}, {"mcc": "111", "mnc": "206"}, {"mcc": "111", "mnc": "207"}, {"mcc": "111", "mnc": "208"}, {"mcc": "111", "mnc": "209"}, {"mcc": "111", "mnc": "210"}, {"mcc": "111", "mnc": "211"}, {"mcc": "111", "mnc": "212"}, {"mcc": "111", "mnc": "213"}, {"mcc": "111", "mnc": "214"}, {"mcc": "111", "mnc": "215"}, {"mcc": "111", "mnc": "216"}, {"mcc": "111", "mnc": "217"}, {"mcc": "111", "mnc": "218"}, {"mcc": "111", "mnc": "219"}, {"mcc": "111", "mnc": "220"}, {"mcc": "111", "mnc": "221"}, {"mcc": "111", "mnc": "222"}, {"mcc": "111", "mnc": "223"}, {"mcc": "111", "mnc": "224"}, {"mcc": "111", "mnc": "225"}, {"mcc": "111", "mnc": "226"}, {"mcc": "111", "mnc": "227"}, {"mcc": "111", "mnc": "228"}, {"mcc": "111", "mnc": "229"}, {"mcc": "111", "mnc": "230"}, {"mcc": "111", "mnc": "231"}, {"mcc": "111", "mnc": "232"}, {"mcc": "111", "mnc": "233"}, {"mcc": "111", "mnc": "234"}, {"mcc": "111", "mnc": "235"}, {"mcc": "111", "mnc": "236"}, {"mcc": "111", "mnc": "237"}, {"mcc": "111", "mnc": "238"}, {"mcc": "111", "mnc": "239"}, {"mcc": "111", "mnc": "240"}, {"mcc": "111", "mnc": "241"}, {"mcc": "111", "mnc": "242"}, {"mcc": "111", "mnc": "243"}, {"mcc": "111", "mnc": "244"}, {"mcc": "111", "mnc": "245"}, {"mcc": "111", "mnc": "246"}, {"mcc": "111", "mnc": "247"}, {"mcc": "111", "mnc": "248"}, {"mcc": "111", "mnc": "249"}, {"mcc": "111", "mnc": "250"}, {"mcc": "111", "mnc": "251"}, {"mcc": "111", "mnc": "252"}, {"mcc": "111", "mnc": "253"}, {"mcc": "111", "mnc": "254"}, {"mcc": "111", "mnc": "255"}, {"mcc": "111", "mnc": "256"}, {"mcc": "111", "mnc": "257"}, {"mcc": "111", "mnc": "258"}, {"mcc": "111", "mnc": "259"}, {"mcc": "111", "mnc": "260"}, {"mcc": "111", "mnc": "261"}, {"mcc": "111", "mnc": "262"}, {"mcc": "111", "mnc": "263"}, {"mcc": "111", "mnc": "264"}, {"mcc": "111", "mnc": "265"}, {"mcc": "111", "mnc": "266"}, {"mcc": "111", "mnc": "267"}, {"mcc": "111", "mnc": "268"}, {"mcc": "111", "mnc": "269"}, {"mcc": "111", "mnc": "270"}, {"mcc": "111", "mnc": "271"}, {"mcc": "111", "mnc": "272"}, {"mcc": "111", "mnc": "273"}, {"mcc": "111", "mnc": "274"}, {"mcc": "111", "mnc": "275"}, {"mcc": "111", "mnc": "276"}, {"mcc": "111", "mnc": "277"}, {"mcc": "111", "mnc": "278"}, {"mcc": "111", "mnc": "279"}, {"mcc": "111", "mnc": "280"}, {"mcc": "111", "mnc": "281"}, {"mcc": "111", "mnc": "282"}, {"mcc": "111", "mnc": "283"}, {"mcc": "111", "mnc": "284"}, {"mcc": "111", "mnc": "285"}, {"mcc": "111", "mnc": "286"}, {"mcc": "111", "mnc": "287"}, {"mcc": "111", "mnc": "288"}, {"mcc": "111", "mnc": "289"}, {"mcc": "111", "mnc": "290"}, {"mcc": "111", "mnc": "291"}, {"mcc": "111", "mnc": "292"}, {"mcc": "111", "mnc": "293"}, {"mcc": "111", "mnc": "294"}, {"mcc": "111", "mnc": "295"}, {"mcc": "111", "mnc": "296"}, {"mcc": "111", "mnc": "297"}, {"mcc": "111", "mnc": "298"}, {"mcc": "111", "mnc": "299"}, {"mcc": "111", "mnc": "300"}, {"mcc": "111", "mnc": "301"}, {"mcc": "111", "mnc": "302"}, {"mcc": "111", "mnc": "303"}, {"mcc": "111", "mnc": "304"}, {"mcc": "111", "mnc": "305"}, {"mcc": "111", "mnc": "306"}, {"mcc": "111", "mnc": "307"}, {"mcc": "111", "mnc": "308"}, {"mcc": "111", "mnc": "309"}, {"mcc": "111", "mnc": "310"}, {"mcc": "111", "mnc": "311"}, {"mcc": "111", "mnc": "312"}]
```

[illegible]

```
{
 "mcc": "111", "mnc": "481"}, {"mcc": "111", "mnc": "482"}, {"mcc": "111", "mnc": "483"},
 {"mcc": "111", "mnc": "484"}, {"mcc": "111", "mnc": "485"}, {"mcc": "111", "mnc": "486"},
 {"mcc": "111", "mnc": "487"}, {"mcc": "111", "mnc": "488"}, {"mcc": "111", "mnc": "489"},
 {"mcc": "111", "mnc": "490"}, {"mcc": "111", "mnc": "491"}, {"mcc": "111", "mnc": "492"},
 {"mcc": "111", "mnc": "493"}, {"mcc": "111", "mnc": "494"}, {"mcc": "111", "mnc": "495"},
 {"mcc": "111", "mnc": "496"}, {"mcc": "111", "mnc": "497"}, {"mcc": "111", "mnc": "498"},
 {"mcc": "111", "mnc": "499"}, {"mcc": "111", "mnc": "500"}, {"mcc": "111", "mnc": "501"},
 {"mcc": "111", "mnc": "502"}, {"mcc": "111", "mnc": "503"}, {"mcc": "111", "mnc": "504"},
 {"mcc": "111", "mnc": "505"}, {"mcc": "111", "mnc": "506"}, {"mcc": "111", "mnc": "507"},
 {"mcc": "111", "mnc": "508"}, {"mcc": "111", "mnc": "509"}, {"mcc": "111", "mnc": "510"},
 {"mcc": "111", "mnc": "511"}, {"mcc": "111", "mnc": "512"}, {"mcc": "111", "mnc": "513"},
 {"mcc": "111", "mnc": "514"}, {"mcc": "111", "mnc": "515"}, {"mcc": "111", "mnc": "516"},
 {"mcc": "111", "mnc": "517"}, {"mcc": "111", "mnc": "518"}, {"mcc": "111", "mnc": "519"},
 {"mcc": "111", "mnc": "520"}, {"mcc": "111", "mnc": "521"}, {"mcc": "111", "mnc": "522"},
 {"mcc": "111", "mnc": "523"}, {"mcc": "111", "mnc": "524"}, {"mcc": "111", "mnc": "525"},
 {"mcc": "111", "mnc": "526"}, {"mcc": "111", "mnc": "527"}, {"mcc": "111", "mnc": "528"},
 {"mcc": "111", "mnc": "529"}, {"mcc": "111", "mnc": "530"}, {"mcc": "111", "mnc": "531"},
 {"mcc": "111", "mnc": "532"}, {"mcc": "111", "mnc": "533"}, {"mcc": "111", "mnc": "534"},
 {"mcc": "111", "mnc": "535"}, {"mcc": "111", "mnc": "536"}, {"mcc": "111", "mnc": "537"},
 {"mcc": "111", "mnc": "538"}, {"mcc": "111", "mnc": "539"}, {"mcc": "111", "mnc": "540"},
 {"mcc": "111", "mnc": "541"}, {"mcc": "111", "mnc": "542"}, {"mcc": "111", "mnc": "543"},
 {"mcc": "111", "mnc": "544"}, {"mcc": "111", "mnc": "545"}, {"mcc": "111", "mnc": "546"},
 {"mcc": "111", "mnc": "547"}, {"mcc": "111", "mnc": "548"}, {"mcc": "111", "mnc": "549"},
 {"mcc": "111", "mnc": "550"}, {"mcc": "111", "mnc": "551"}, {"mcc": "111", "mnc": "552"},
 {"mcc": "111", "mnc": "553"}, {"mcc": "111", "mnc": "554"}, {"mcc": "111", "mnc": "555"},
 {"mcc": "111", "mnc": "556"}, {"mcc": "111", "mnc": "557"}, {"mcc": "111", "mnc": "558"},
 {"mcc": "111", "mnc": "559"}, {"mcc": "111", "mnc": "560"}, {"mcc": "111", "mnc": "561"},
 {"mcc": "111", "mnc": "562"}, {"mcc": "111", "mnc": "563"}, {"mcc": "111", "mnc": "564"},
 {"mcc": "111", "mnc": "565"}, {"mcc": "111", "mnc": "566"}, {"mcc": "111", "mnc": "567"},
 {"mcc": "111", "mnc": "568"}, {"mcc": "111", "mnc": "569"}, {"mcc": "111", "mnc": "570"},
 {"mcc": "111", "mnc": "571"}, {"mcc": "111", "mnc": "572"}, {"mcc": "111", "mnc": "573"},
 {"mcc": "111", "mnc": "574"}, {"mcc": "111", "mnc": "575"}, {"mcc": "111", "mnc": "576"},
 {"mcc": "111", "mnc": "577"}, {"mcc": "111", "mnc": "578"}, {"mcc": "111", "mnc": "579"},
 {"mcc": "111", "mnc": "580"}, {"mcc": "111", "mnc": "581"}, {"mcc": "111", "mnc": "582"},
 {"mcc": "111", "mnc": "583"}, {"mcc": "111", "mnc": "584"}, {"mcc": "111", "mnc": "585"},
 {"mcc": "111", "mnc": "586"}, {"mcc": "111", "mnc": "587"}, {"mcc": "111", "mnc": "588"},
 {"mcc": "111", "mnc": "589"}, {"mcc": "111", "mnc": "590"}, {"mcc": "111", "mnc": "591"},
 {"mcc": "111", "mnc": "592"}, {"mcc": "111", "mnc": "593"}, {"mcc": "111", "mnc": "594"},
 {"mcc": "111", "mnc": "595"}, {"mcc": "111", "mnc": "596"}, {"mcc": "111", "mnc": "597"},
 {"mcc": "111", "mnc": "598"}, {"mcc": "111", "mnc": "599"}]
```

### 3.7.5.11 Request Timeout

SEPP Supports requestTimeout parameter to avoid the request timeout of services. The requestTimeout is increased in the given services to avoid any request timeout that may happen due to the processing of response by the stub server.

- The user has to set requestTimeout to 2000 in the cn32f-svc, pn32f-svc, plmn-egress-gateway, n32-egress-gateway, and n32-ingress-gateway sections of the ocsepp\_custom\_values\_<version>.yaml file.
- The user has to set requestTimeout to 5000 in the n32-ingress-gateway section of the ocsepp\_custom\_values\_<version>.yaml file.

```
requestTimeout: 2000 #(ms)
```

**Note**

- If the ATS cases are failing due to the request timeout, increase the request timeout of a particular service which is creating the issue. For example set request timeout to "5000".
- If ATS is not configured, requestTimeout must be set to the following values:

**Table 3-26 requestTimeout**

| Service             | REQUEST TIMEOUT |
|---------------------|-----------------|
| cn32f-svc           | 2000            |
| pn32f-svc           | 1100            |
| plmn-egress-gateway | 1000            |
| n32-egress-gateway  | 1500            |
| n32-ingress-gateway | 700             |

**Updating n32-ingress-gateway config map**

In the `ocsepp_custom_values.yaml` file, under `n32-ingress-gateway` section, set the `requestTimeout` parameter value to 5000 (ms) for updating the config map.

```

routesConfig:
- id: n32f
 #Below field is used to provide an option to enable/disable route level
 xfccHeaderValidation, it will override global configuration for
 xfccHeaderValidation.enabled
 metadata:
 requestTimeout: 1200

```

### 3.7.5.12 Idle Timeout

The Idle timeout must be increased in the given services to avoid any pending request transaction to timeout due to its idle state in the request queue.

In the `ocsepp_custom_values_<version>.yaml` file, under `n32-ingress-gateway` section, set the `jettyIdleTimeout` parameter value to 5000(ms).

```

#Jetty Idle Timeout Settings (ms)
jettyIdleTimeout: 5000 #(ms)

```

**Note**

- If the ATS is not configured, the `jettyIdleTimeout` value should be 3000 (ms).
- If the ATS cases are failing due to the request timeout, increase idle timeout of a particular service which is failing. For example set idle timeout to "5000".

### 3.7.5.13 EvictSanHeaderCacheDelay

The SEPP supports `evictSanHeaderCacheDelay` to update the cache on the PN32F service with the updated values of Remote SEPP and Remote SEPP set associated with the SAN header.

The user has to set `evictSanHeaderCacheDelay` parameter to 100 in `pn32f-svc` microservice section of custom yaml file.

```
configs:
 evictSanHeaderCacheDelay: 100 #(ms)
```

#### Note

If ATS is not configured `evictSanHeaderCacheDelay` must be set to 50000 in the `pn32f-svc` section of `ocsepp_custom_values_<version>.yaml` file.

### 3.7.5.14 Configuring the Reroute Attempts for Egress Gateway

The `attempts` parameter is used in alternate routing, when the request has to be routed to multiple SEPP.

Under config manager section in `ocsepp_custom_values_<version>.yaml` file, the default value for `attempts` is set to 0. Set `attempts` to 3 before running ATS test execution.

```
alternateRoute:
 sbiReRoute:
 sbiRoutingErrorActionSets: [{"id": "action_0", "action": "reroute",
 "attempts": 3, "blacklist": {"enabled": false, "duration": 60000}}]
```

### 3.7.5.15 NRF Discovery Cache Refresh Timeout

The `nrfDiscoveryCacheRefreshTimeout` parameter defines the timer value when the UDR Discovery request is triggered if coherence map containing UDR Profile information is empty. After this timer expiry, UDR Discovery request will be initiated by NRF Client towards NRF.

In `ocsepp_custom_values_<version>.yaml` file, under `pn32f-svc` section, set the `nrfDiscoveryCacheRefreshTimeout` parameter value to 1000 (ms).

```
configs:
 nrfDiscoveryCacheRefreshTimeout: 1000 #(ms)
```

#### Note

To run the Cat-3 testcases, user must have preinstalled coherence service while SEPP installation.

### 3.7.5.16 Configuring Alternate Routing based on the DNS SRV Record for Home Network Functions

The Alternate Routing based on the DNS SRV Record for Home Network Functions feature can be enabled as follows:

Update the following parameters in `ocsepp_custom_values.yaml` file to configure the DNS SRV feature:

At Global Section:

```
global:
 alternateRouteServiceEnable : true
```

At Alternate route section:

```
alternate-route:
 global:
 alternateRouteServiceEnable: true
```

Update the Target host:

```
alternate-route: #Static virtual FQDN Config
 staticVirtualFqdns:
 - name: https://sepp.ats.test.routing.com
 alternateFqdns:
 #Below Target FQDNs needs to be updated for ATS DNS SRV scenarios
 - target: <ats-release-name>-stubserver
 port: 8443
 priority: 100
 weight: 90
 - target: <ats-release-name>-stubserver-2
 port: 8443
 priority: 100
 weight: 10
 - target: <ats-release-name>-stubserver-3
 port: 8443
 priority: 1
 weight: 90
```

Here, update the target parameter with the user-specific ATS release name.

Example: If the user defines the `<ats-release-name>` as `sepp-ats-rel` and `<stub-server-name>` as `stubserver`, then update the target as `sepp-ats-rel-stubserver`.

#### Note

For DNS SRV and SOR features, the `plmn-egress-gateway` must be deployed in REST mode. In the `ocsepp_custom_values.yaml` file, update:

```
plmn-egress-gateway:
 routeConfigMode: REST
```

**Note**

In DNS SRV feature, the rerouting to next peer is decided on the basis of error codes or exceptions. If some exception arises which is not present in the json file `DNSSRVCriteriaSet.json` (may cause scenario failures), user must add those exceptions in the exceptions list at the following paths:

In the file `/var/lib/jenkins/ocsepp_tests/data/DNSSRVCriteriaSet.json`, update

```
"exceptions": [
 "java.net.SocketException",
 "java.nio.channels.ClosedChannelException"
]
```

In the file `/var/lib/jenkins/ocsepp_tests/cust_data/DNSSRVCriteriaSet.json`, update

```
"exceptions": [
 "java.net.SocketException",
 "java.nio.channels.ClosedChannelException"
]
```

### 3.7.5.17 Configuring Load Sharing among Multiple Remote SEPP Nodes

The load sharing among multiple Remote SEPP nodes feature can be configured as follows:

1. Enable the following parameters in `ocsepp_custom_values.yaml` file to configure the load sharing feature:

- a. Enable the following flag at `alternate_route` section:

```
alternate-route:
 global:
 alternateRouteServiceEnable: true
```

- b. Replace the release name of the Target host:

```
alternate-route: #Static virtual FQDN Config
staticVirtualFqdns:
- name: http://sepp.ats.loadsharing.com
 alternateFqdns:
 #Below Target FQDNs needs to be updated for ATS DNS SRV scenarios
- target: <ats-release-name>-stubserver
 port: 8443
 priority: 100
 weight: 90
- target: <ats-release-name>-stubserver-2
 port: 8443
 priority: 100
 weight: 10
- target: <ats-release-name>-stubserver-3
 port: 8443
```



```
priority: 1
weight: 90
```

**Note**

If the user has chosen the ats release name as sepp-ats-rel  
Modify  
target: <release-name>-stubserver with actual release name of ATS  
deployment.  
Example: sepp-ats-rel-stubserver

2. Enable the following parameter at NrfClient Global parameters:

```
alternateRouteServiceEnable: true
```

### 3.7.5.18 Enabling Persistent Volume Storage

ATS supports Persistent storage to retain ATS historical build execution data, test cases, and one-time environment variable configurations.

**Note**

By default, this feature is disabled.

To enable persistent storage:

1. Create a PVC and associate the same to the ATS pod.
2. Set the **PVEnabled** flag to **true**.
3. Set **PVClaimName** to PVC that is created for ATS.

```
bddclient:
 PVEnabled: false
 PVClaimName: "sepp-pvc"
```

For more details on Persistent Volume Storage, you can refer to [Persistent Volume for 5G ATS](#)

### 3.7.6 Deploying ATS in Kubernetes Cluster

**Note**

It is important to ensure that all the three components; ATS, Stub and SEPP are in the same namespace.

If the namespace does not exist, run the following command to create a namespace:

```
kubectl create namespace ocsepp
```

### Deploying ATS:

```
helm install -name <release_name> ocats-sepp-23.4.0.tgz --namespace
<namespace_name> -f <values-yaml-file>
```

Example:

```
helm install -name ocats ocats-sepp-23.4.0.tgz --namespace ocsepp -f ocats-
sepp-custom-values.yaml
```

## 3.7.7 Verifying ATS Deployment

Run the following command to verify ATS deployment:

```
helm status <release_name>
```

Checking Pod Deployment:

```
kubectl get pod -n seppsvc
```

Checking Service Deployment:

```
kubectl get service -n seppsvc
```

**Figure 3-20 Checking Pod and Service Deployment**

|                           |              |               |               |                             |
|---------------------------|--------------|---------------|---------------|-----------------------------|
| sepp-ats-rel-bddclient    | LoadBalancer | 10.233.24.74  | 10.75.152.26  | 8080:32385/TCP              |
| 21h                       |              |               |               |                             |
| sepp-ats-rel-stubserver   | LoadBalancer | 10.233.32.171 | 10.75.153.203 | 8080:32312/TCP,8091:32115/T |
| CP,8443:32550/TCP         | 21h          |               |               |                             |
| sepp-ats-rel-stubserver-2 | LoadBalancer | 10.233.27.4   | 10.75.153.146 | 8080:32045/TCP,8091:32171/T |
| CP,8443:30155/TCP         | 21h          |               |               |                             |
| sepp-ats-rel-stubserver-3 | LoadBalancer | 10.233.29.122 | 10.75.152.153 | 8080:31926/TCP,8091:31699/T |
| CP,8443:32407/TCP         | 21h          |               |               |                             |

**Figure 3-21 Checking Service Deployment**

|                                                                                            |     |         |   |     |
|--------------------------------------------------------------------------------------------|-----|---------|---|-----|
| [root@ocne-1-8-4-cluster-mohit184-bastion-1 cloud-user]# kubectl get po -n sepp   grep ats |     |         |   |     |
| sepp-ats-rel-bddclient-595b7d66b9-k76zj                                                    | 1/1 | Running | 0 | 31s |
| sepp-ats-rel-stubserver-2-d9f6dc6f-4f4t5                                                   | 1/1 | Running | 0 | 21s |
| sepp-ats-rel-stubserver-3-6cb9f99f6-wxf8k                                                  | 1/1 | Running | 0 | 16s |
| sepp-ats-rel-stubserver-6f4785b7d4-7tbq9                                                   | 1/1 | Running | 0 | 11d |
| [root@ocne-1-8-4-cluster-mohit184-bastion-1 cloud-user]#                                   |     |         |   |     |

### 3.7.8 Post Installation Steps (If persistent volume is used)

If persistent volume is used, follow the post-installation steps as mentioned in the [Persistent Volume for 5G ATS](#) section.

## 3.8 Installing ATS for UDR

Before installing ATS for UDR, it is important to ensure the resource requirements are met. For information about the resource requirements, see [Resource Requirements](#).

The UDR ATS installation procedure includes:

1. [Locating and downloading the ATS images](#)
2. [Preparing to Deploy ATS and Stub Pods](#)
3. [Loading UDR ATS Images](#)
4. [Loading UDR Stub Images in the SLF-NewFeatures or SLF-Regression Pipeline](#)
5. [Loading UDR Stub Images in the UDR-NewFeatures or UDR-Regression Pipeline](#)
6. [Configuring ATS](#)
  - [Configuring static port](#)
  - [Configuring UDR Part of Label](#)
  - [Enabling Persistent Storage](#)
  - [Settings for Oauth2 test cases](#)
  - [IPv6 on ATS and NRF-Stub](#)
  - [Configuring Health-Check pipeline](#)
  - [Service account requirements](#)
  - [Configuring service mesh](#)
7. [Deploying ATS in Kubernetes Cluster](#)
8. [Deploying Stub Pods in Kubernetes Cluster](#)
9. [Verifying UDR-ATS Deployment](#)
10. [Post-Installation Steps](#)

### 3.8.1 Resource Requirements

ATS resource requirements for UDR, SLF, and EIR are as follows:

**Table 3-27 UDR - Total Number of Resources**

| Resource Name       | vCPUs | Memory (GB) | Storage (GB) |
|---------------------|-------|-------------|--------------|
| UDR SUT Totals      | 39    | 39          | 34           |
| DB Tier Totals      | 20    | 41          | 37           |
| ATS Totals          | 15    | 11          | 0            |
| Grand Total UDR-ATS | 74    | 91          | 71           |

**Table 3-28 SLF - Total Number of Resources**

| Resource Name       | vCPUs (excluding sidecar) | vCPUs (including sidecar) | Memory (GB) (excluding sidecar) | Memory (including sidecar) | Storage (GB) |
|---------------------|---------------------------|---------------------------|---------------------------------|----------------------------|--------------|
| SLF-SUT Totals      | 83                        | 113                       | 83                              | 114                        | 64           |
| DB Tier Totals      | 20                        | 41                        | 28                              | 53                         | 37           |
| ATS Totals          | 8                         | 11                        | 6                               | 8                          | 0            |
| Grand Total SLF-ATS | 111                       | 165                       | 117                             | 175                        | 98           |

**Table 3-29 EIR - Total Number of Resources**

| Resource Name       | vCPUs | Memory (GB) | Storage (GB) |
|---------------------|-------|-------------|--------------|
| EIR SUT Totals      | 25    | 25          | 19           |
| DB Tier Totals      | 20    | 41          | 37           |
| ATS Totals          | 7     | 6           | 0            |
| Grand Total EIR-ATS | 52    | 72          | 59           |

**Table 3-30 UDR Operational Resource Requirements**

| Microservice               | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|----------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| ocudr-ingressgateway-sig   | 2                      | 2                            | 1 GB                              | 2                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-ingressgateway-prov  | 2                      | 2                            | 1 GB                              | 2                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-nudr-dr-service      | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-dr-prov-service | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-notify-service  | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-oc-diam-gateway      | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-diameter-proxy  | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |

Table 3-30 (Cont.) UDR Operational Resource Requirements

| Microservice                                            | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|---------------------------------------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| ocudr-egressgateway                                     | 2                      | 2                            | 1 GB                              | 1                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-nudr-config                                       | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-config-server                                | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-nrf-client-nfmanagement                      | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-appinfo                                           | 0.5                    | 1                            | 1 GB                              | 1                             | 1                         | 0.5                    | 1                            | 1                                 |
| ocudr-alternate-route                                   | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-migration                                    | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-bulk-import                                  | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-ondemand-migration                           | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-performance                                       | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-performance-prov                                  | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| UDR Additional Resources (Hooks/Init/Update Containers) | -                      | -                            | -                                 | -                             | -                         | 6                      | 6                            | 6                                 |
| provgw-prov-ingressgateway                              | 2                      | 2                            | 0                                 | 2                             | 1                         | 2                      | 2                            | 1                                 |

Table 3-30 (Cont.) UDR Operational Resource Requirements

| Microservice                                               | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|------------------------------------------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| provgw-provgw-service                                      | 1                      | 1                            | 0                                 | 2                             | 1                         | 1                      | 1                            | 1                                 |
| provgw-provgw-config                                       | 1                      | 1                            | 0                                 | 1                             | 1                         | 1                      | 1                            | 1                                 |
| provgw-provgw-config-server                                | 1                      | 1                            | 0                                 | 1                             | 1                         | 1                      | 1                            | 1                                 |
| provgw-prov-egressgateway                                  | 2                      | 2                            | 0                                 | 2                             | 1                         | 2                      | 2                            | 1                                 |
| Provgw Additional Resources (Hooks/Init/Update Containers) | -                      | -                            | -                                 | -                             | -                         | 5                      | 5                            | 5                                 |
| UDR-SUT Total (UDR and ProvGw)                             | -                      | -                            | -                                 | -                             | -                         | 39                     | 39                           | 34                                |

Table 3-31 SLF Operational Resource Requirements

| Microservice             | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|--------------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| ocudr-ingressgateway-sig | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |

Table 3-31 (Cont.) SLF Operational Resource Requirements

| Micro service                      | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|------------------------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| ocudr-ingressgateway-prov          | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |
| ocudr-nudr-dr-service              | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-nudr-dr-prov-service         | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-ingressgateway               | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |
| ocudr-nudr-config                  | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-nudr-config-server           | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-nudr-nrf-client-nfmanagement | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-appinfo                      | 0.5                                        | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 0.5                                       | 1                                         | 1                                               | 1.5                                             | 1                                 |
| ocudr-alternate-route              | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |

Table 3-31 (Cont.) SLF Operational Resource Requirements

| Micro service                                           | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|---------------------------------------------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| ocudr-nudr-bulk-import                                  | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-performanc                                        | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-performanc-prov                                   | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| ocudr-nudr-export-tool                                  | 2                                          | 1                                            | 2                                                | 1                                                  | 3 GB                              | 1                             | 1                         | 2                                         | 3                                         | 2                                               | 3                                               | 3                                 |
| SLF Additional Resources (Hooks/Init/Update Containers) | -                                          | -                                            | -                                                | -                                                  | -                                 | -                             | -                         | 6                                         | 7                                         | 6                                               | 7                                               | 3                                 |
| provgw-prov-ingress-gateway                             | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |
| provgw-prov-egress-gateway                              | 2                                          | 0.5                                          | 2                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 2                                         | 2.5                                       | 2                                               | 2.5                                             | 1                                 |



Table 3-31 (Cont.) SLF Operational Resource Requirements

| Micro service                                              | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|------------------------------------------------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| provgw-provgw-service                                      | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 2                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| provgw-provgw-config                                       | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| provgw-provgw-config-server                                | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| provgw-auditor-service                                     | 1                                          | 0.5                                          | 1                                                | 0.5                                                | 1 GB                              | 1                             | 1                         | 1                                         | 1.5                                       | 1                                               | 1.5                                             | 1                                 |
| Provgw Additional Resources (Hooks/Init/Update Containers) | -                                          | -                                            | -                                                | -                                                  | -                                 | -                             | -                         | 5                                         | 6                                         | 5                                               | 6                                               | 3                                 |

Table 3-31 (Cont.) SLF Operational Resource Requirements

| Microservice                             | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|------------------------------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| SLF-SUT Total Required (SLFs and ProvGw) | -                                          | -                                            | -                                                | -                                                  | -                                 | -                             | -                         | 83                                        | 113                                       | 83                                              | 114                                             | 64                                |

Table 3-32 EIR Operational Resource Requirements

| Microservice               | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|----------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| ocudr-ingressgateway-sig   | 2                      | 2                            | 1 GB                              | 2                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-ingressgateway-prov  | 2                      | 2                            | 1 GB                              | 2                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-nudr-dr-service      | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-dr-prov-service | 1                      | 1                            | 1 GB                              | 2                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-ingressgateway       | 2                      | 2                            | 1 GB                              | 2                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-nudr-config          | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-config-server   | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |

**Table 3-32 (Cont.) EIR Operational Resource Requirements**

| Microservice                                            | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|---------------------------------------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| ocudr-nudr-nrf-client-nfmanagement                      | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-appinfo                                           | 0.5                    | 1                            | 1 GB                              | 1                             | 1                         | 0.5                    | 1                            | 1                                 |
| ocudr-alternate-route                                   | 2                      | 2                            | 1 GB                              | 1                             | 1                         | 2                      | 2                            | 1                                 |
| ocudr-nudr-bulk-import                                  | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-performance                                       | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-performance-prov                                  | 1                      | 1                            | 1 GB                              | 1                             | 1                         | 1                      | 1                            | 1                                 |
| ocudr-nudr-export-tool                                  | 2                      | 2                            | 3 GB                              | 1                             | 1                         | 2                      | 2                            | 3                                 |
| EIR Additional Resources (Hooks/Init/Update Containers) | -                      | -                            | -                                 | -                             | -                         | 6                      | 6                            | 3                                 |
| EIR-SUT Total Required                                  | -                      | -                            | -                                 | -                             | -                         | 25                     | 25                           | 19                                |

**Table 3-33 ATS Resource Requirements for UDR mode**

| Microservice           | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment)    | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|------------------------------|------------------------|------------------------------|-----------------------------------|
| ATS Behave             | 4                      | 4                            | 0                                 | -                             | 1                            | 4                      | 4                            | 0                                 |
| NRF-Stub (python stub) | 2                      | 1                            | 0                                 | -                             | 2 (Two separate deployments) | 4                      | 2                            | 0                                 |

Table 3-33 (Cont.) ATS Resource Requirements for UDR mode

| Microservice              | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment)    | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|---------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|------------------------------|------------------------|------------------------------|-----------------------------------|
| Notify-Stub (python stub) | 2                      | 1                            | 0                                 | -                             | 2 (Two separate deployments) | 4                      | 2                            | 0                                 |
| diam-stub                 | 1                      | 1                            | 0                                 | -                             | 2 (Two separate deployments) | 2                      | 2                            | 0                                 |
| fourg-stub                | 1                      | 1                            | 0                                 | -                             | 1                            | 1                      | 1                            | 0                                 |
| ATS Totals                | -                      | -                            | -                                 | -                             | -                            | 15                     | 11                           | 0                                 |

Table 3-34 ATS Resource Requirements for SLF Mode

| Microservice       | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment)    | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (excludes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|--------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|------------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| ocats-udr          | 4                                          | 2                                            | 4                                                | 1                                                  | 0                                 | 1                             | 1                            | 4                                         | 6                                         | 4                                               | 5                                               | 0                                 |
| nrf-ocstub-python  | 2                                          | .5                                           | 1                                                | .5                                                 | 0                                 | 1                             | 2 (Two separate deployments) | 4                                         | 5                                         | 2                                               | 3                                               | 0                                 |
| ATS Total required | -                                          | -                                            | -                                                | -                                                  | -                                 | -                             | -                            | 8                                         | 11                                        | 6                                               | 8                                               | 0                                 |

Table 3-35 ATS Resource Requirements for EIR mode

| Microservice | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|--------------|------------------------|------------------------------|-----------------------------------|-------------------------------|---------------------------|------------------------|------------------------------|-----------------------------------|
| ocats-udr    | 3                      | 4                            | 0                                 | 1                             | 1                         | 3                      | 4                            | 0                                 |

Table 3-35 (Cont.) ATS Resource Requirements for EIR mode

| Microservice                        | vCPUs Required per Pod | Memory Required per Pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment)    | vCPUs Required - Total | Memory Required - Total (GB) | Storage PVC Required - Total (GB) |
|-------------------------------------|------------------------|------------------------------|-----------------------------------|-------------------------------|------------------------------|------------------------|------------------------------|-----------------------------------|
| nrf-ocstub-python (For deployments) | 2                      | 1                            | 0                                 | 1                             | 2 (Two separate deployments) | 4                      | 2                            | 0                                 |
| ATS Total required                  | -                      | -                            | -                                 | -                             | -                            | 7                      | 6                            | 0                                 |

**Note**

ATS total resource calculation includes 2 nrf-stub deployments and 1 ATS deployment.

Table 3-36 UDR DB Tier Pods

| Micro service name              | Container name         | Number of Pods | vCPUs Requirement Per Pod | Memory Requirement Per Pod | PVC Requirement                | Total Resources           |
|---------------------------------|------------------------|----------------|---------------------------|----------------------------|--------------------------------|---------------------------|
| Management node                 | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 4 CPUs<br>10 GB Memory    |
| Data node                       | mysqlndbcluster        | 2              | 2 CPUs                    | 4 GB                       | 8*2 → 16 GB                    | 5 CPUs<br>9 GB Memory     |
|                                 | db-backup-executor-svc |                | 100m CPU                  | 128 MB                     |                                |                           |
| APP SQL node                    | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 5 CPUs<br>10.5 GB Memory  |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                                |                           |
| SQL node (Used for Replication) | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 5 CPUs<br>10.5 GB Memory  |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                                |                           |
| DB Monitor Service              | db-monitor-svc         | 1              | 200m CPUs                 | 500 MB                     | -                              | 0.5 CPUs<br>0.5 GB Memory |
| DB Backup Manager Service       | replication-svc        | 1              | 200m CPU                  | 500 MB                     | -                              | 0.5 CPUs<br>0.5 GB Memory |
| Total                           | -                      | -              | -                         | -                          | 34+3 (Replication Svc) → 37 GB | 20 CPUs<br>41 GB Memory   |

**Table 3-37 SLF DB Tier Pods**

| Micro service name              | Container name         | Number of Pods | vCPUs Requirement Per Pod | Memory Requirement Per Pod | PVC Requirement                | Total Resources         |
|---------------------------------|------------------------|----------------|---------------------------|----------------------------|--------------------------------|-------------------------|
| Management node                 | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 8 CPUs<br>12 GB Memory  |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
| Data node                       | mysqlndbcluster        | 2              | 2 CPUs                    | 4 GB                       | 8*2 → 16 GB                    | 9 CPUs<br>11 GB Memory  |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
|                                 | db-backup-executor-svc |                | 100m CPU                  | 128 MB                     |                                |                         |
| APP SQL node                    | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 9 CPUs<br>13 GB Memory  |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                                |                         |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
| SQL node (Used for Replication) | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB                     | 9 CPUs<br>13 GB Memory  |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                                |                         |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
| DB Monitor Service              | db-monitor-svc         | 1              | 200m CPUs                 | 500 MB                     | -                              | 3 CPUs<br>2 GB Memory   |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
| DB Backup Manager Service       | replication-svc        | 1              | 200m CPU                  | 500 MB                     | -                              | 3 CPUs<br>2 GB Memory   |
|                                 | istio-proxy            |                | 2 CPUs                    | 1 GB                       |                                |                         |
| Total                           | -                      | -              | -                         | -                          | 34+3 (Replication Svc) → 37 GB | 41 CPUs<br>53 GB Memory |

**Table 3-38 EIR DB Tier Pods**

| Micro service name              | Container name         | Number of Pods | vCPUs Requirement Per Pod | Memory Requirement Per Pod | PVC Requirement | Total Resources          |
|---------------------------------|------------------------|----------------|---------------------------|----------------------------|-----------------|--------------------------|
| Management node                 | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB      | 4 CPUs<br>10 GB Memory   |
| Data node                       | mysqlndbcluster        | 2              | 2 CPUs                    | 4 GB                       | 8*2 → 16 GB     | 5 CPUs<br>9 GB Memory    |
|                                 | db-backup-executor-svc |                | 100m CPU                  | 128 MB                     |                 |                          |
| APP SQL node                    | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB      | 5 CPUs<br>10.5 GB Memory |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                 |                          |
| SQL node (Used for Replication) | mysqlndbcluster        | 2              | 2 CPUs                    | 5 GB                       | 3*2 → 6 GB      | 5 CPUs<br>10.5 GB Memory |
|                                 | init-sidecar           |                | 100m CPU                  | 256 MB                     |                 |                          |

Table 3-38 (Cont.) EIR DB Tier Pods

| Micro service name        | Container name  | Number of Pods | vCPUs Requirement Per Pod | Memory Requirement Per Pod | PVC Requirement                   | Total Resources         |
|---------------------------|-----------------|----------------|---------------------------|----------------------------|-----------------------------------|-------------------------|
| DB Monitor Service        | db-monitor-svc  | 1              | 200m CPUs                 | 500 MB                     | -                                 | 1 CPUs<br>1 GB Memory   |
| DB Backup Manager Service | replication-svc | 1              | 200m CPU                  | 500 MB                     | -                                 | 1 CPUs<br>1 GB Memory   |
| Total                     | -               | -              | -                         | -                          | 34+3<br>(Replication Svc) → 37 GB | 20 CPUs<br>41 GB Memory |

Table 3-39 SLF Performance based Resource Details

| -                        | vCPUs (excluding sidecar)                 | vCPUs (including sidecar)           | Memory (GB) (excluding sidecar)                  | Memory (GB) (including sidecar)                    | Storage (GB)                      | -                             | -                         | -                                         | -                                         | -                                                | -                                                | -                                 |
|--------------------------|-------------------------------------------|-------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|--------------------------------------------------|--------------------------------------------------|-----------------------------------|
| SLF-SUT Totals           | 58                                        | 105                                 | 46                                               | 93                                                 | 22                                | -                             | -                         | -                                         | -                                         | -                                                | -                                                | -                                 |
| DB Tier Totals           | 40                                        | 80                                  | 40                                               | 80                                                 | 20                                | -                             | -                         | -                                         | -                                         | -                                                | -                                                | -                                 |
| ATS Totals               | 6                                         | 4                                   | 1                                                | 3                                                  | 0                                 | -                             | -                         | -                                         | -                                         | -                                                | -                                                | -                                 |
| Grand Total SLF-ATS      | 104                                       | 193                                 | 94                                               | 183                                                | 42                                | -                             | -                         | -                                         | -                                         | -                                                | -                                                | -                                 |
| Micro service            | CPUs Required per Pod (excluding sidecar) | CPUs required for sidecar container | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | CPUs Required - Total (excluding sidecar) | CPUs Required - Total (including sidecar) | Memory Required - Total (GB) (excluding sidecar) | Memory Required - Total (GB) (including sidecar) | Storage PVC Required - Total (GB) |
| ocudr-ingressgateway-sig | 6                                         | 4                                   | 4                                                | 4                                                  | 1 GB                              | 2                             | 2                         | 12                                        | 20                                        | 8                                                | 16                                               | 2                                 |

Table 3-39 (Cont.) SLF Performance based Resource Details

| -                                                          | vCPU<br>s<br>(excl<br>uding<br>sidec<br>ar) | vCPU<br>s<br>(inclu<br>ding<br>sidec<br>ar) | Mem<br>ory<br>(GB)<br>(excl<br>uding<br>sidec<br>ar) | Mem<br>ory<br>(GB)<br>(inclu<br>ding<br>sidec<br>ar) | Stora<br>ge<br>(GB) | - | - | -  | -  | - | -  | - |
|------------------------------------------------------------|---------------------------------------------|---------------------------------------------|------------------------------------------------------|------------------------------------------------------|---------------------|---|---|----|----|---|----|---|
| ocudr-<br>ingres<br>sgate<br>way-<br>prov                  | 6                                           | 4                                           | 4                                                    | 4                                                    | 1 GB                | 2 | 2 | 12 | 20 | 8 | 16 | 2 |
| ocudr-<br>nudr-<br>drserv<br>ice                           | 5                                           | 4                                           | 4                                                    | 4                                                    | 1 GB                | 2 | 2 | 10 | 18 | 8 | 16 | 2 |
| ocudr-<br>nudr-<br>dr-<br>provs<br>ervice                  | 5                                           | 4                                           | 4                                                    | 4                                                    | 1 GB                | 2 | 2 | 10 | 18 | 8 | 16 | 2 |
| ocudr-<br>egres<br>sgate<br>way                            | 1                                           | 2                                           | 1                                                    | 2                                                    | 1 GB                | 1 | 1 | 1  | 3  | 1 | 3  | 1 |
| ocudr-<br>nudr-<br>config                                  | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |
| ocudr-<br>nudr-<br>config<br>-<br>server                   | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |
| ocudr-<br>nudr-<br>nrf-<br>client-<br>nfman<br>agem<br>ent | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |
| ocudr-<br>appinf<br>o                                      | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |
| ocudr-<br>altern<br>ate-<br>route                          | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |
| ocudr-<br>perfor<br>manc<br>e                              | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                | 1 | 1 | 1  | 2  | 1 | 2  | 1 |



Table 3-39 (Cont.) SLF Performance based Resource Details

| -                                                                                           | vCPU<br>s<br>(excl<br>uding<br>sidec<br>ar) | vCPU<br>s<br>(inclu<br>ding<br>sidec<br>ar) | Mem<br>ory<br>(GB)<br>(excl<br>uding<br>sidec<br>ar) | Mem<br>ory<br>(GB)<br>(inclu<br>ding<br>sidec<br>ar) | Stora<br>ge<br>(GB)        | -                                | - | -  | -   | -  | -  | -  |
|---------------------------------------------------------------------------------------------|---------------------------------------------|---------------------------------------------|------------------------------------------------------|------------------------------------------------------|----------------------------|----------------------------------|---|----|-----|----|----|----|
| ocudr-<br>perfor<br>manc<br>e-prov                                                          | 1                                           | 1                                           | 1                                                    | 1                                                    | 1 GB                       | 1                                | 1 | 1  | 2   | 1  | 2  | 1  |
| Additi<br>onal<br>SLF<br>Resou<br>rces<br>(Hook<br>s/Init/<br>Updat<br>e<br>Contai<br>ners) | -                                           | -                                           | -                                                    | -                                                    | -                          | -                                | - | 6  | 12  | 6  | 12 | 6  |
| SLF<br>SUT<br>Totals                                                                        | -                                           | -                                           | -                                                    | -                                                    | -                          | -                                | - | 58 | 105 | 46 | 93 | 22 |
| Micro<br>servic<br>e                                                                        | Contai<br>ner                               | Numb<br>er of<br>Pods                       | CPU<br>Requi<br>remen<br>t Per<br>Pod                | Memo<br>ry<br>Requi<br>remen<br>t Per<br>Pod         | PVC<br>Requi<br>remen<br>t | Total<br>Resou<br>rces           | - | -  | -   | -  | -  | -  |
| Mana<br>geme<br>nt<br>node                                                                  | mysql<br>ndbcl<br>uster                     | 2                                           | 2<br>CPUs                                            | 5 GB                                                 | 3*2 →<br>6 GB              | 8<br>CPUs<br>12 GB<br>Memo<br>ry | - | -  | -   | -  | -  | -  |
|                                                                                             | istio-<br>proxy                             |                                             | 2<br>CPUs                                            | 1 GB                                                 |                            |                                  |   |    |     |    |    |    |
| Data<br>node                                                                                | mysql<br>ndbcl<br>uster                     | 2                                           | 2<br>CPUs                                            | 4 GB                                                 | 8*2 →<br>16 GB             | 9<br>CPUs<br>11 GB<br>Memo<br>ry | - | -  | -   | -  | -  | -  |
|                                                                                             | istio-<br>proxy                             |                                             | 2<br>CPUs                                            | 1 GB                                                 |                            |                                  |   |    |     |    |    |    |
|                                                                                             | db-<br>backu<br>p-<br>execut<br>or-svc      |                                             | 100m<br>CPU                                          | 128<br>MB                                            |                            |                                  |   |    |     |    |    |    |
| APP<br>SQL<br>node                                                                          | mysql<br>ndbcl<br>uster                     | 2                                           | 2<br>CPUs                                            | 5 GB                                                 | 3*2 →<br>6 GB              | 9<br>CPUs<br>13 GB<br>Memo<br>ry | - | -  | -   | -  | -  | -  |
|                                                                                             | init-<br>sideca<br>r                        |                                             | 100m<br>CPU                                          | 256<br>MB                                            |                            |                                  |   |    |     |    |    |    |
|                                                                                             | istio-<br>proxy                             |                                             | 2<br>CPUs                                            | 1 GB                                                 |                            |                                  |   |    |     |    |    |    |

Table 3-39 (Cont.) SLF Performance based Resource Details

| -                                                | vCPUs<br>(excluding<br>sidecar) | vCPUs<br>(including<br>sidecar) | Memory<br>(GB)<br>(excluding<br>sidecar) | Memory<br>(GB)<br>(including<br>sidecar) | Storage<br>(GB) | -                                 | - | - | - | - | -                                               | -                                  |
|--------------------------------------------------|---------------------------------|---------------------------------|------------------------------------------|------------------------------------------|-----------------|-----------------------------------|---|---|---|---|-------------------------------------------------|------------------------------------|
| SQL<br>node<br>(Used<br>for<br>Replica-<br>tion) | mysql<br>ndbcl<br>uster         | 2                               | 2<br>CPUs                                | 5 GB                                     | 3*2 →<br>6 GB   | 9<br>CPUs<br>13 GB<br>Memo-<br>ry | - | - | - | - | -                                               | -                                  |
|                                                  | init-<br>sidecar                |                                 | 100m<br>CPU                              | 256<br>MB                                |                 |                                   |   |   |   |   |                                                 |                                    |
|                                                  | istio-<br>proxy                 |                                 | 2<br>CPUs                                | 1 GB                                     |                 |                                   |   |   |   |   |                                                 |                                    |
| DB<br>Monit<br>or<br>Servic<br>e                 | db-monit<br>or-svc              | 1                               | 200m<br>CPUs                             | 1 GB                                     | -               | 3<br>CPUs<br>2 GB<br>Memo-<br>ry  | - | - | - | - | -                                               | -                                  |
|                                                  | istio-<br>proxy                 |                                 | 2<br>CPUs                                | 500<br>MB                                |                 |                                   |   |   |   |   |                                                 |                                    |
| DB<br>Backu<br>p<br>Mana<br>ger<br>Servic<br>e   | replic<br>ation-<br>svc         | 1                               | 200m<br>CPU                              | 500<br>MB                                | -               | 3<br>CPUs<br>2 GB<br>Memo-<br>ry  | - | - | - | - | -                                               | -                                  |
|                                                  | istio-<br>proxy                 |                                 | 2<br>CPUs                                | 1 GB                                     |                 |                                   |   |   |   |   |                                                 |                                    |
| Total                                            | -                               | -                               | -                                        | -                                        | -               | -                                 | - | - | - | - | 34+3(<br>Replica-<br>tion<br>Svc)<br>→ 37<br>GB | 41<br>CPUs<br>53 GB<br>Memo-<br>ry |

Table 3-40 ATS and stub requirement for SLF Performance

| Micro<br>servi-<br>ce | vCPUs<br>Requi-<br>red<br>per<br>Pod<br>(excl-<br>uding<br>sidecar) | vCPUs<br>requi-<br>red<br>for<br>sidecar<br>conta-<br>iner<br>per<br>pod | Mem-<br>ory<br>Requi-<br>red<br>per<br>Pod<br>(GB)<br>(excl-<br>uding<br>sidecar) | Mem-<br>ory<br>requi-<br>red<br>for<br>sidecar<br>conta-<br>iner<br>per<br>pod(<br>GB) | Stora-<br>ge<br>PVC<br>Requi-<br>red<br>per<br>Pod<br>(GB) | Repli-<br>cas<br>(regul-<br>ar<br>depl-<br>oyment) | Repli-<br>cas<br>(ATS<br>depl-<br>oyment) | vCPUs<br>Requi-<br>red -<br>Total<br>(excl-<br>udes<br>sidecar) | vCPUs<br>Requi-<br>red -<br>Total<br>(incl-<br>udes<br>sidecar) | Mem-<br>ory<br>Requi-<br>red -<br>Total<br>(GB)<br>(incl-<br>udes<br>sidecar) | Mem-<br>ory<br>Requi-<br>red -<br>Total<br>(GB)<br>(incl-<br>udes<br>sidecar) | Stora-<br>ge<br>PVC<br>Requi-<br>red -<br>Total<br>(GB) |
|-----------------------|---------------------------------------------------------------------|--------------------------------------------------------------------------|-----------------------------------------------------------------------------------|----------------------------------------------------------------------------------------|------------------------------------------------------------|----------------------------------------------------|-------------------------------------------|-----------------------------------------------------------------|-----------------------------------------------------------------|-------------------------------------------------------------------------------|-------------------------------------------------------------------------------|---------------------------------------------------------|
| ocats-<br>udr         | 6                                                                   | 2                                                                        | 8                                                                                 | 2                                                                                      | 0                                                          | 1                                                  | 1                                         | 6                                                               | 8                                                               | 8                                                                             | 10                                                                            | 0                                                       |

Table 3-40 (Cont.) ATS and stub requirement for SLF Performance

| Micro service      | vCPUs Required per Pod (excluding sidecar) | vCPUs required for sidecar container per pod | Memory Required per Pod (GB) (excluding sidecar) | Memory required for sidecar container per pod (GB) | Storage PVC Required per Pod (GB) | Replicas (regular deployment) | Replicas (ATS deployment) | vCPUs Required - Total (excludes sidecar) | vCPUs Required - Total (includes sidecar) | Memory Required - Total (GB) (includes sidecar) | Memory Required - Total (GB) (includes sidecar) | Storage PVC Required - Total (GB) |
|--------------------|--------------------------------------------|----------------------------------------------|--------------------------------------------------|----------------------------------------------------|-----------------------------------|-------------------------------|---------------------------|-------------------------------------------|-------------------------------------------|-------------------------------------------------|-------------------------------------------------|-----------------------------------|
| ATS Total required | -                                          | -                                            | -                                                | -                                                  | -                                 | -                             |                           | 6                                         | 8                                         | 8                                               | 10                                              | 0                                 |

## 3.8.2 Locating and Downloading ATS Image

To locate and download the ATS Image from MOS:

1. Log in to [My Oracle Support](#) using appropriate credentials.
2. Select the **Patches & Updates** tab.
3. In the **Patch Search** window, click **Product or Family (Advanced)**.
4. Enter *Oracle Communications Cloud Native Core - 5G* in the **Product** field.
5. Select *Oracle Communications Cloud Native Core Unified Data Repository <release\_number>* from the **Release** drop-down.
6. Click **Search**. The **Patch Advanced Search Results** list appears.
7. Select the required ATS patch from the list. The **Patch Details** window appears.
8. Click **Download**. The **File Download** window appears.
9. Click the **<p\*\*\*\*\*\_<release\_number>\_Tekelec>.zip** file.
10. Untar the zip file to access all the ATS images. The *ocats\_udr\_pkg\_24\_1\_1\_0\_0.tgz* file has following files:
  - *ocats\_udr\_pkg\_24\_1\_1\_0\_0.tgz*
  - *ocats\_udr\_pkg\_24\_1\_1\_0\_0-README.txt*
  - *ocats\_udr\_pkg\_24\_1\_1\_0\_0.tgz.sha256*
  - *ocats-udr-custom-configtemplates-24.1.1.0.0.zip*
  - *ocats-udr-custom-configtemplates-24.1.1.0.0-README.txt*
  - *ocats\_udr\_pkg\_24\_1\_1\_0\_0.mcafee.sw\_packaging.log*

The *ocats\_udr\_pkg\_24\_1\_1\_0\_0.tgz* file contains:

```
|_ _ ocats-udr-ats-pkg-24.1.1.tgz
|
|_ _ _ _ _ ocats-udr-24.1.1.tgz (Helm Charts)
```

```

| |_ _ _ _ _ ocats-udr-images-24.1.1.tar(Docker Images)
| |_ _ _ _ _ OCATS-UDR-Readme.txt
| |_ _ _ _ _ ocats-udr-24.1.1.tgz.sha256
| |_ _ _ _ _ ocats-udr-images-24.1.1.tar.sha256
| |_ _ _ _ _ ocats-udr-data-24.1.1.tgz
| |_ _ _ _ _ ocats-udr-data-24.1.1.tgz.sha256
|_ _ ocats-udr-stub-pkg-24.1.1.tgz
 |_ _ _ _ _ ocstub-py-24.1.1.tgz (Helm Charts)
 |_ _ _ _ _ fourg-stub-24.1.1.tgz (Helm Charts)
 |_ _ _ _ _ diam-stub-24.1.1.tgz (Helm Charts)
 |_ _ _ _ _ ocstub-py-24.1.1.tar(Docker Images)
 |_ _ _ _ _ ocats-udr-fourg-stub-images-24.1.1.tar(Docker Images)
 |_ _ _ _ _ ocats-udr-diam-stub-images-24.1.1.tar(Docker Images)
 |_ _ _ _ _ OCATS-UDR-STUB-Readme.txt
 |_ _ _ _ _ ocstub-py-24.1.1.tgz.sha256
 |_ _ _ _ _ fourg-stub-24.1.1.tgz.sha256
 |_ _ _ _ _ diam-stub-24.1.1.tgz.sha256
 |_ _ _ _ _ ocstub-py-24.1.1.tar.sha256
 |_ _ _ _ _ ocats-udr-fourg-stub-images-24.1.1.tar.sha256
 |_ _ _ _ _ ocats-udr-diam-stub-images-24.1.1.tar.sha256

```

The ocats-udr-custom-configtemplates-24.1.1.0.0.zip file contains:

ocats-udr-custom-configtemplates-24.1.1.0.0.zip

```

 |_ _ _ _ _ ocats-udr-custom-values-24.1.1.yaml(Custom values for
UDR-ATS)

```

```

 |_ _ _ _ _ ocstub-py-custom-values-24.1.1.yaml (Custom values for
COMMON-PYTHON-STUB)

```

```

 |_ _ _ _ _ fourg-stub-custom-values-24.1.1.yaml (Custom values for
FOURG-STUB)

```

```

 |_ _ _ _ _ diam-stub-custom-values-24.1.1.yaml(Custom values for
DIAMETER-STUB)

```

Copy the `ocats_uds_pkg_24_1_1_0_0.tgz` tar file and `ocats-uds-custom-configtemplates-24.1.1.0.0.zip` file to OCCNE, OCI, or Kubernetes cluster, where you want to deploy ATS.

### 3.8.3 Preparing to Deploy ATS and Stub Pods

To deploy ATS and stub pods in Kubernetes cluster:

#### ① Note

Deploy ATS and Subscriber Location Function (SLF) in the same namespace.

- [Loading UDR ATS Images](#)
- Loading UDR Stub Images
  - [SLF-NewFeatures or SLF-Regression pipeline](#)
  - [UDR-NewFeatures or UDR-Regression](#)

### 3.8.4 Loading UDR ATS Image

To load UDR ATS image:

1. Run the following command to extract the tar file content.  
`tar -xvf ocats_uds_pkg_24_1_1_0_0.tgz`

The output of this command is:

- `ocats-uds-ats-pkg-24.1.1.tgz`
- `ocats-uds-stub-pkg-24.1.1.tgz`

2. Run the following command to extract the helm charts and docker images of ATS.  
`tar -xvf ocats-uds-ats-pkg-24.1.1.tgz`

The output of this command is:

- `ocats-uds-24.1.1.tgz`
- `ocats-uds-images-24.1.1.tar`
- `ocats-uds-data-24.1.1.tgz`

3. Run the following command to extract the helm charts and docker images of ATS.  
`tar -xvf ocats-uds-stub-pkg-24.1.1.tgz`

The output of this command is:

- `ocstub-py-24.1.1.tgz` (Helm charts)
- `fourg-stub-24.1.1.tgz` (Helm charts)
- `diam-stub-24.1.1.tgz` (Helm charts)
- `ocats-uds-notify-stub-images-24.1.1.tar` (Docker image)
- `ocats-uds-fourg-stub-images-24.1.1.tar` (Docker Images)
- `ocats-uds-diam-stub-images-24.1.1.tar` (Docker Images)
- `ocstub-py-24.1.1.tar` (Docker image)

The `ocats-udr-images-24.1.1.tar` file contains docker images (`ocats-udr-images-24.1.1.tar`) of ATS for UDR 24.1.1.

4. Run the following command in your cluster to load the ATS image.

```
docker load --input ocats-udr-images-24.1.1.tar
```

**Note**

For CNE 1.8.0 and above, you can use Podman instead of Docker. See the following sample Podman command:

```
sudo podman load --input ocats-udr-images-24.1.1.tar
```

5. Run the following commands to tag and push the ATS image to your registry.

```
docker tag ocats-udr-images:24.1.1 <registry>/ocats-udr-images:24.1.1
```

```
docker push <registry>/ocats-udr-images:24.1.1
```

In the previous command, `<registry>` is the name of docker image repository.

**Note**

For CNE 1.8.0 and above, you can use Podman instead of Docker to tag and push the Docker image. Run the following sample Podman command to tag and push Docker image:

```
sudo podman tag ocats-udr-images:24.1.1 <customer repo>/ <image name>:<image version>
```

```
sudo podman push <customer repo>/<image name>:<image version>
```

6. Run the following command to untar the helm charts (`ocats-udr-24.1.1.tgz`) and update the registry name, image name, and tag (if required) in the `ocats-udr-custom-values-24.1.1.yaml` file.

```
tar -xvf ocats-udr-24.1.1.tgz
```

Output:

```
ocats-udr/Chart.yaml
```

```
ocats-udr/values.yaml
```

```
ocats-udr/templates/NOTES.txt
```

```
ocats-udr/templates/_helpers.tpl
```

```
ocats-udr/templates/deployment.yaml
ocats-udr/templates/ingress.yaml
ocats-udr/templates/service.yaml
ocats-udr/templates/serviceaccount.yaml
```

Output:

```
ocats-udr
├── Chart.yaml
├── templates
│ ├── deployment.yaml
│ ├── _helpers.tpl
│ ├── ingress.yaml
│ ├── NOTES.txt
│ ├── serviceaccount.yaml
│ └── service.yaml
└── values.yaml
```

### 3.8.5 Loading UDR Stub Images in the SLF-NewFeatures, EIR-NewFeatures, SLF-Regression, or EIR-Regression Pipeline

To load the UDR stub images:

#### **Note**

For the SLF-NewFeatures, EIR-NewFeatures, or SLF-Regression pipeline, deploy only ocstub-py Helm chart. The ocstub-py-24.1.1.tar contains common-python-stub image (ocstub-py:24.1.1).

1. Run the following command to extract the tar file content.  
`tar -xvf ocats_udr_pkg_24.1.1_0_0.tgz`

The output of this command is:

- ocats-udr-ats-pkg-24.1.1.tgz
- ocats-udr-stub-pkg-24.1.1.tgz

2. Run the following command to extract the stub tar file content.  
`tar -xvf ocats-udr-stub-pkg-24.1.1.tgz`

The output of this command is:

- ocstub-py-24.1.1.tgz (Helm charts)
  - fourg-stub-24.1.1.tgz (Helm charts)
  - diam-stub-24.1.1.tgz (Helm charts)
  - ocats-udr-notify-stub-images-24.1.1.tar (Docker image)
  - ocats-udr-fourg-stub-images-24.1.1.tar (Docker Images)
  - ocats-udr-diam-stub-images-24.1.1.tar (Docker Images)
  - ocstub-py-24.1.1.tar (Docker image)
3. To load the UDR stub images (ocstub-py-24.1.1.tar) in the SLF-NewFeatures, EIR-NewFeatures, or SLF-Regression pipeline, run the following command in your cluster.

```
docker load --input ocstub-py-24.1.1.tar
```

4. Run the following commands to tag and push the stub image to your registry.

```
docker tag ocstub-py:24.1.1 <registry>/ocstub-py:24.1.1
```

```
docker push <registry>/ocstub-py:24.1.1
```

In the previous command, <registry> is the name of docker image repository.

5. Run the following command to untar the common python stub helm charts (ocstub-py-24.1.1.tgz) to get ocstub-py charts:

```
tar -xvf ocstub-py-24.1.1.tgz
```

Output:

```
ocstub-py/Chart.yaml
ocstub-py/values.yaml
ocstub-py/templates/_helpers.tpl
ocstub-py/templates/deployment.yaml
ocstub-py/templates/ingress.yaml
ocstub-py/templates/service.yaml
ocstub-py/templates/serviceaccount.yaml
ocstub-py/README.md
```

**Output:**

```
ocstub-py
├── Chart.yaml
├── templates
│ ├── deployment.yaml
│ ├── _helpers.tpl
│ ├── ingress.yaml
│ └── serviceaccount.yaml
```



```
| └── service.yaml
| └── values.yaml
| └── README.md
```

### 3.8.6 Loading UDR Stub Images in the UDR-NewFeatures or UDR-Regression Pipeline

#### Note

- To run UDR-NewFeatures/UDR-Regression, ocstub-py, fourg-stub, and diam-stub must be deployed
- ocstub-py-24.1.1.tar contains common-python-stub image (ocstub-py:24.1.1)
- ocats-udr-fourg-stub-images-24.1.1.tar contains fourg-stub image (ocats-udr-fourg-stub-images:24.1.1)
- ocats-udr-diam-stub-images-24.1.1.tar contains diam-stub image (ocats-udr-diam-stub-images:24.1.1)

To load the UDR Stub Images:

1. Run the following command to extract the tar file content.  
`tar -xvf ocats-udr-ats-pkg-24.1.1.tgz`

The output of this command is:

- ocats-udr-ats-pkg-24.1.1.tgz
  - ocats-udr-stub-pkg-24.1.1.tgz
2. Run the following command to extract the stub tar file content.  
`tar -xvf ocats-udr-stub-pkg-24.1.1.tgz`
    - ocstub-py-24.1.1.tgz (Helm charts)
    - fourg-stub-24.1.1.tgz (Helm charts)
    - diam-stub-24.1.1.tgz (Helm charts)
    - ocats-udr-notify-stub-images-24.1.1.tar (Docker image)
    - ocats-udr-fourg-stub-images-24.1.1.tar (Docker Images)
    - ocats-udr-diam-stub-images-24.1.1.tar (Docker Images)
    - ocstub-py-24.1.1.tar (Docker image)
  3. Run the following commands in your cluster to load the stub images.

```
docker load --input ocstub-py-24.1.1.tar
docker load --input ocats-udr-fourg-stub-images-24.1.1.tar
docker load --input ocats-udr-diam-stub-images-24.1.1.tar
```

4. Run the following commands to tag and push the stub image to your registry.

```
docker tag ocstub-py:24.1.1 <registry>/ocstub-py:24.1.1
docker push <registry>/ocstub-py:24.1.1
docker tag ocats-udr-fourg-stub-images:24.1.1 <registry>/ocats-udr-
fourg-stub-images:24.1.1
docker push <registry>/ocats-udr-fourg-stub-images:24.1.1
docker tag ocats-udr-diam-stub-images:24.1.1 <registry>/ocats-udr-diam-
stub-images:24.1.1
docker push <registry>/ocats-udr-diam-stub-images:24.1.1
```

5. Run the following command to untar all the stub charts:

```
tar -xvf ocstub-py-24.1.1.tgz
tar -xvf fourg-stub-24.1.1.tgz
tar -xvf diam-stub-24.1.1.tgz
```

Output of each helm chart is as follows:

ocstub-py:

```
├── Chart.yaml
├── templates
│ ├── deployment.yaml
│ ├── _helpers.tpl
│ ├── ingress.yaml
│ ├── serviceaccount.yaml
│ └── service.yaml
├── values.yaml
└── README.md
```

fourg-stub

```
├── Chart.yaml
├── templates
│ ├── deployment.yaml
│ ├── _helpers.tpl
│ ├── ingress.yaml
│ ├── NOTES.txt
│ └── service.yaml
```

```
└── values.yaml

diam-stub
├── Chart.yaml
├── templates
│ ├── deployment.yaml
│ ├── _helpers.tpl
│ ├── ingress.yaml
│ ├── NOTES.txt
│ └── service.yaml
└── values.yaml
```

### 3.8.7 Configuring ATS

It is important to configure the following features before deploying ATS for UDR:

- [Configuring Docker Registry](#)
- [Enabling Static Port](#)
- [Enabling Persistent Volume](#)
- [Settings for OAuth2 Test Cases](#)
- [Enabling IPv6 on ATS](#)
- [Configuring ATS to Run Health-Check Pipeline](#)
- [Creating Service Account](#)
- [Enabling Service Mesh](#)
- [Configuring NRF Stub](#)
- [Configuring Notify Stub](#)
- [Configuring Fourg Stub](#)
- [Configuring Diameter Stub](#)

#### Note

- The deployment of notify-stub, fourg-stub, and diam-stub are not applicable to SLF pipelines.
- Service name used by each of the stubs must be unique for successful deployment.

### 3.8.7.1 Configuring Docker Registry

Update the docker registry file as follows:

```
image:
 repository: <docker registry>:<docker port>/ocats-udr-images
```

### 3.8.7.2 Enabling Static Port

To enable static port:

#### ① Note

ATS supports static port. By default, this feature is not enabled.

In the ocats-udr-custom-values-24.1.1.yaml file under **service** section, set the `staticNodePortEnabled` parameter as `true` and `staticNodePort` parameter with valid `nodePort` value.

Here is a sample configuration for enabling static port in the ocats-udr-custom-values-24.1.1.yaml file:

```
service:
 customExtension:
 labels: {}
 annotations: {}
 type: LoadBalancer
 port: "8080"
 staticNodePortEnabled: true
 staticNodePort: "31083"
```

### 3.8.7.3 Enabling Persistent Volume

To enable persistent volume, create a PVC and associate the same to the ATS pod.

#### ① Note

- For information about creating PVC, see [Persistent Volume for 5G ATS](#).
- For information about PVC size calculation, see [UDR Application Log Collection](#).

To enable persistent volume, set the following parameters in the values.yaml file:

1. Set the **PVEnabled** flag to 'true'.
2. Set **PVClaimName** to 'PVC' that user has created for ATS.

```
deployment:
 customExtension:
 labels: {}
 annotations: {}
```

```
PVEnabled: true
PVClaimName: "ocats-udr-pvc"
```

### 3.8.7.4 Settings for OAuth2 Test Cases

UDR-ATS supports test cases that verifies OAuth2 validation scenarios at both UDR Provisioning Ingress Gateway and UDR Signaling Ingress Gateway. To run the OAuth2 related test cases on ATS:

1. Generate four ECDSA private keys in pem format (two each for UDR Provisioning Ingress Gateway and UDR Signaling Ingress Gateway) and enter the keys name in the **privateKey** field.
2. Generate four public certificates using the private keys (two each for UDR Provisioning Ingress Gateway and UDR Signaling Ingress Gateway) and enter the certificate names under **publicKey**.

#### Sample Commands to Generate ECDSA Private Keys and Certificates

```
openssl ecparam -genkey -name prime256v1 -noout -out ec_private_key1.pem
openssl pkcs8 -topk8 -in ec_private_key1.pem -inform pem -out
ec_private_key_pkcs8.pem -outform pem -nocrypt
openssl req -new -key ec_private_key_pkcs8.pem -x509 -nodes -days 365 -out
4bc0c762-0212-416a-bd94-b7f1fb348bd4.crt -subj "/C=IN/ST=KA/L=BLR/O=ORACLE/
OU=CGBU/CN=ocnrf-endpoint.ocnrf.svc.cluster.local"
```

#### Note

- You can configure the above mentioned inputs only if OAuth2 is configured on UDR. For information to configure OAuth2 on UDR, see the "Configuring OAuth2.0" section in the *Oracle Communications Cloud Native Core Unified Data Repository User Guide*.
- ATS configures Ingress Gateway with secret name, keyId, certificate name, and instanceId based on the inputs provided in the ocats-udr-custom-values-24.1.1.yaml file.
- ATS supports only ES256 algorithm to generate token for this release. User should generate ECDSA private key to test OAuth2 feature.

3. Update the ocats-udr-custom-values-24.1.1.yaml file with the public and private keys generated in the previous steps. A sample code snippet is as follows:

#### Sample: OAuth Configuration

```
deployment: oauthKeys:
 - keyId: '664b344e74294c8fa5d2e7dfaaba407'
 udrSecret: 'oauthsecret1'
 privateKey: 'ec_private_key1.pem'
 publicKey: '4bc0c762-0212-416a-bd94-b7f1fb348bd4.crt'
 issuerId: '4bc0c762-0212-416a-bd94-b7f1fb348bd4'
 reqType: 'igw-sig'
 - keyId: '664b344e74294c8fa5d2e7dfaaba408'
 udrSecret: 'oauthsecret2'
 privateKey: 'ec_private_key2.pem'
 publicKey: '4bc0c762-0212-416a-bd94-b7f1fb348bd5.crt'
 issuerId: '4bc0c762-0212-416a-bd94-b7f1fb348bd5'
```

```

 reqType: 'igw-sig'
 - keyId: '664b344e74294c8fa5d2e7dfaaaba409'
 udrSecret: 'oauthsecret3'
 privateKey: 'ec_private_key3.pem'
 publicKey: '4bc0c762-0212-416a-bd94-b7f1fb348bd6.crt'
 issuerId: '4bc0c762-0212-416a-bd94-b7f1fb348bd6'
 reqType: 'igw-prov'
 - keyId: '664b344e74294c8fa5d2e7dfaaaba410'
 udrSecret: 'oauthsecret4'
 privateKey: 'ec_private_key4.pem'
 publicKey: '4bc0c762-0212-416a-bd94-b7f1fb348bd7.crt'
 issuerId: '4bc0c762-0212-416a-bd94-b7f1fb348bd7'
 reqType: 'igw-prov'

```

In the above code snippet,

- **issuerId** is any uuid that follows NfInstanceId format.
  - **keyId** is user-defined value.
  - **reqType** indicates the mapping between secret created and Ingress Gateway. igw-sig indicates secrets are to be used for UDR Signaling Ingress Gateway and igw-prov indicates UDR Provisioning Ingress Gateway.
4. Create four secrets in the namespace where ATS and UDR are installed, with public certificate and enter the value for the **udrSecret** field.

### 3.8.7.5 Enabling IPv6 on ATS

If you are deploying ATS setup on IPv6 system, then enable the following flag in the ocats-udr-custom-values.yaml file:

#### Enabling IPv6 on ATS

```

deployment:
 ipv6enabled: true

```

### 3.8.7.6 Configuring ATS to Run Health-Check Pipeline

To run the SLF-Health-Check, UDR-Health-Check, or EIR-HealthCheck features, provide the following inputs in the values.yaml file if ATS is being deployed on OCCNE.

```

deployment:
 webscale: false
 occnehostip: <base64 encoded occne bastion ip>
 occnehostusername: <base64 encoded occne login user name>
 occnehostpassword: <base64 encoded occne login password>

```

And, provide the following inputs to run the SLF-Health-Check, UDR-Health-Check, or EIR-HealthCheck features, if ATS is being deployed on Webscale.

```

deployment:
 webscale: true
 #Provide Webscale Environment details with base64 encoding
 webscalejumpserverip: <base64 encoded jump server ip>
 webscalejumpserverusername: <base64 encoded jump server username>

```

```

webscalejumpserverpassword: <base64 encoded jump server password>
webscaleprojectname: <base64 encoded webscale project name>
webscalelabserverFQDN: <base64 encoded lab server fqdn>
webscalelabserverport: <base64 encoded lab server port>
webscalelabserverusername: <base64 encoded lab server username>
webscalelabserverpassword: <base64 encoded lab server password>

```

You can configure the name of the secret that the ATS creates for healthcheck pipeline:

```
healthchecksecretname: ats-healthcheck-secret
```

For more information, see [ATS Health Check](#).

#### Note

UDR-ATS creates a secret with the name 'healthcheck-secret' on Kubernetes to store the above inputs.

### 3.8.7.7 Creating Service Account

To run SLF-ATS, it is mandatory to create a service account using the following inputs:

```

rules:
- apiGroups: ["apps"]
 resources: ["deployments", "replicasets", "deployments/scale", "statefulsets/
scale"]
 verbs: ["get", "watch", "list", "update"]
- apiGroups: [""]
 resources: ["pods", "deployments", "pods/log", "configmaps", "pods/exec"]
 verbs: ["get", "watch", "list", "update", "create"]

```

To run UDR-ATS, it is mandatory to create a service account using the following inputs:

```

rules:
- apiGroups: ["apps"]
 resources: ["deployments", "replicasets", "deployments/scale", "statefulsets/
scale"]
 verbs: ["get", "watch", "list", "update"]
- apiGroups: [""]
 resources: ["pods", "deployments", "pods/log", "configmaps", "pods/
exec", "services"]
 verbs: ["get", "watch", "list", "update", "create", "delete"]
- apiGroups: ["batch"]
 resources: ["jobs"]
 verbs: ["get", "create", "delete", "update", "list"]

```

**Note**

For information about creating service account, see the *Oracle Communications Cloud Native Core Unified Data Repository Installation and Upgrade Guide* available on MOS.

### 3.8.7.8 Enabling Service Mesh

**Note**

The UDR-NewFeatures, UDR-Regression, EIR-NewFeatures, and EIR-Regression pipelines do not support deployment on service mesh enabled system.

To enable service mesh:

1. If service mesh is not enabled at the global level for namespace, then run the following command to enable service mesh at the namespace level before deploying UDR-ATS.  
`kubectl label --overwrite namespace <namespace_name> istio-injection=enabled`

**Example:**

```
kubectl label --overwrite namespace ocudr istio-injection=enabled
```

2. Add the following annotation in the **lbDeployment** section under global section of the `ocats-udr-custom-values-24.1.1.yaml` file:

```
global:
 # ***** Sub-Section Start: Custom Extension Global Parameters

#*****

customExtension:
 allResources:
 labels: {}
 annotations: {}

 lbServices:
 labels: {}
 annotations: {}

 lbDeployments:
 labels: {}
 annotations:
 traffic.sidecar.istio.io/excludeInboundPorts: "8080"
 traffic.sidecar.istio.io/excludeOutboundPorts: "443,9000,22,9090"

 nonlbServices:
 labels: {}
 annotations: {}

 nonlbDeployments:
 labels: {}
 annotations: {}
```



3. Use the following code snippet to create an envoy filter for both UDR and ATS:

```

apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
 name: <user defined name for envoy filter>
 namespace: <namespace where ATS is deployed>
spec:
 workloadSelector:
 labels:
 app: ocats-udr
 configPatches:
 - applyTo: NETWORK_FILTER
 match:
 listener:
 filterChain:
 filter:
 name: "envoy.http_connection_manager"
 patch:
 operation: MERGE
 value:
 typed_config:
 '@type': type.googleapis.com/
envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManag
er
 server_header_transformation: PASS_THROUGH

apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
 name: <user defined name for envoy filter>
 namespace: <namespace where ATS is deployed>
spec:
 workloadSelector:
 labels:
 app: ocats-udr
 configPatches:
 - applyTo: NETWORK_FILTER
 match:
 listener:
 filterChain:
 filter:
 name: "envoy.http_connection_manager"
 patch:
 operation: MERGE
 value:
 typed_config:
 '@type': type.googleapis.com/
envoy.config.filter.network.http_connection_manager.v2.HttpConnectionManag
er
 forward_client_cert_details: ALWAYS_FORWARD_ONLY

```

For more information about envoy filter and service mesh configuration, see the *Oracle Communications Cloud Native Core Unified Data Repository Installation and Upgrade Guide*.

4. After deploying service mesh, create Peer Authentication on the pods for inter pod communication. A sample template is as follows:

```
apiVersion: security.istio.io/v1beta1
kind: PeerAuthentication
metadata:
 name: <ATS Peer Authentication name>
 namespace: <ATS deployment namespace>
spec:
 selector:
 matchLabels:
 app: ocats-udr
 mtls:
 mode: PERMISSIVE

```

```
apiVersion: security.istio.io/v1beta1
kind: PeerAuthentication
metadata:
 name: <ATS Stubs Peer Authentication name>
 namespace: <ATS Stubs deployment namespace>
spec:
 selector:
 matchLabels:
 app: ocstub-py mtl:
 mode: PERMISSIVE
```

5. ATS sidecar must have at least 2 CPUs and 2 Gi memory. ATS sidecar is configured as follows:

```
deployment:
 customExtension:
 annotations:
 sidecar.istio.io/proxyCPU: "2000m"
 sidecar.istio.io/proxyCPULimit: "2000m"
 sidecar.istio.io/proxyMemory: "2Gi"
 sidecar.istio.io/proxyMemoryLimit: "2Gi"
 proxy.istio.io/config: |
 concurrency: 4
```

### 3.8.7.9 Configuring Stubs

From ATS release 24.1.0:

- The common python stub chart and image must be used for two NRF-stubs, Notify-stub (supports TLS and non-TLS in single deployment), and SCP stub.
- Two different deployments of NRF (primary and secondary) using common python stub chart and image must be deployed for SLF-ATS and EIR-ATS.
- Two different deployments of NRF (primary and secondary), one deployment for notify stub (TLS or non-TLS scenarios) and one deployment for SCP stub must be deployed for UDR-ATS.

### 3.8.7.9.1 Configuring NRF Stub

UDR-ATS, SLF-ATS, and EIR-ATS requires two NRF stubs, one as primary NRF and other as secondary NRF. These NRF stubs is deployed using ocstub-py-custom-values-24.1.1.yaml.

To configure NRF stub:

- Provide the docker registry details with ocstub-py-custom-values-24.1.1.yaml file where images are pushed as follows:

```
image:
 repository: <docker registry>:<docker port>
```

- Set the env.NF to SLF, UDR, or 5G EIR depending upon the deployed NF:

```
env:
 ...
 NF: SLF
```

```
env:
 ...
 NF: UDR
```

```
env:
 ...
 NF: 5G_EIR
```

- Set the appendReleaseName parameter to false for backward compatibility and bypass\_additional\_port parameter to true to avoid creating HTTP1 port. Each deployment should have unique service name:

```
service:
 ...
 name: <user-defined nrf stub service name> #Example nrf-stub-service1
 for primary NRF stub and nrf-stub-service2 for secondary NRF stub
 appendReleaseName: false #this is to avoid adding release name at the
 beginning of the service name thereby supporting backward compatibility
 ...
 bypass_additional_port: true #to avoid creating HTTP1 port.
```

### 3.8.7.9.2 Configuring Notify Stub

UDR-ATS requires notify-stub to be deployed in two modes using ocstub-py-custom-values-24.1.1.yaml:

- HTTP mode and TLS mode: To receive TLS and non-TLS notifications from UDR
- SCP mode: To validate SCP routing scenarios

To configure notify stub for Notify and SCP stub modes:

- Update the docker registry where common python stub image is pushed as follows:

```
image:
 repository: <docker registry>:<docker port>
```

- Set the env.NF to UDR depending upon the whether the deployed NF is UDR:

```
env:
 ...
 NF: UDR
```

- Set the appendReleaseName parameter to false for backward compatibility and bypass\_additional\_port parameter to true to avoid creating HTTP1 port. Common python stub deployed as HTTP and TLS server must have unique service name:

```
service:
 ...
 name: <user-defined notify stub service name> #Example notify-stub-
service for TLS/non-TLS stub
 appendReleaseName: false #this is to avoid adding release name at the
beginning of the service name thereby supporting backward compatibility
 ...
 bypass_additional_port: true #to avoid creating HTTP1 port.
```

- Configure the stub with private key, certificates. and secret that is used by UDR Egress Gateway for TLS notification validation:

```
env:
 ...
 cert_secret_name: <TLS secret of UDR egressgateway> #E.g., ocudr-gateway-
secret
 ca_cert: <CA certificate used for UDR egressgateway deployment> #E.g.,
caroot.cer
 client_cert: <Client certificate used for UDR egressgateway deployment>
#E.g., apigatewayrsa.cer
 private_key: <Private key used for UDR egressgateway deployment> #E.g.,
rsa_private_key_pkcs1.pem
 ...
 CLIENT_CERT_REQ: true
```

- To deploy notify-stub as SCP set the appendReleaseName parameter to false for backward compatibility and bypass\_additional\_port parameter to true to avoid creating HTTP1 port. Common python stub deployed as HTTP and TLS server must have unique service name. Set the CLIENT\_CERT\_REQ to false as its default value:

```
service:
 ...
 name: <user-defined scp stub service name> #Example scp-stub-service for
SCP stub
 appendReleaseName: false #this is to avoid adding release name at the
beginning of the service name thereby supporting backward compatibility
 ...
 bypass_additional_port: true #to avoid creating HTTP1 port.
```

### 3.8.7.9.3 Configuring Fourg Stub

UDR-ATS requires one fourg stub to test the migration scenarios. Configure the docker registry where diameter stub image is pushed as follows:

```
image:
 repository: <docker registry>:<docker port>/ocats-udr-fourg-stub-images
```

### 3.8.7.9.4 Configuring Diameter Stub

UDR-ATS requires two diameter stubs as two different diameter peers.

- diam-toola: peernode1 (seagull1a.seagull.com)
- diam-toolb: peernode2 (seagull1b.seagull.com)

#### ① Note

UDR-ATS uses diam-stub-24.1.1.yaml file for diameter stub configuration.

To configure diameter stubs:

- Update the docker registry where diameter stub image is pushed as follows:

```
image:
 repository: <docker registry>:<docker port>/ocats-udr-diam-stub-images
```

- Configure diam-toola as follows:

```
deployment:
 SourceIdentity: seagull1a.seagull.com
 SourceRealm: seagulla.com
```

- Configure diam-toolb as follows:

```
deployment:
 SourceIdentity: seagull1b.seagull.com
 SourceRealm: seagullb.com
```

## 3.8.8 Deploying ATS in Kubernetes Cluster

You can deploy ATS Pod in Kubernetes cluster using Helm commands.

Run the following command to deploy ATS.

```
helm install --name <release_name> --namespace <namespace_name> -f <values-yaml-file> ocats-udr
```

#### Example:

```
helm install --name ocats-udr-24.1.1 --namespace ocudr -f ocats-udr-custom-values-24.1.1.yaml ocats-udr
```

### 3.8.8.1 Deploying ATS Pod and Stubs in Kubernetes Cluster

ATS resource allocation must be done by referring to the [Resource Requirements](#) section to support parallel test execution feature. For more information on Parallel Test Execution, see [Parallel Test Execution](#). If you must enable the Application Log Collection, see [UDR Application Log Collection](#).

The CPU and memory utilization depends on the number of behave command executed at given point of time. UDR-ATS runs six behave commands at a time and SLF-ATS runs seven behave commands at a time.

You can deploy stub pod in Kubernetes cluster using Helm commands.

Run the following command to deploy to deploy ATS and stubs:

```
helm install --name <release_name> --namespace <namespace_name> -f <values-yaml-file> ocats-udr
```

#### Example:

```
helm install --name ocats-udr-24.1.1 --namespace ocudr -f ocats-udr-custom-values-24.1.1.yaml ocats-udr
```

#### SLF-NewFeatures, EIR-NewFeatures or SLF-Regression Pipeline

Run the following command to deploy NRF-STUB:

```
helm install --name <release_name> --namespace <namespace_name> -f <values-yaml-file> ocstub-py
```

#### Example:

```
helm install --name stub1 --namespace ocudr -f ocstub-py-custom-values-24.1.1.yaml ocstub-py
helm install --name stub2 --namespace ocudr -f ocstub-py-custom-values-24.1.1.yaml ocstub-py
```

#### Note

To test DNS SRV feature in SLF-Regression and EIR-NewFeatures, NRF stub needs to be deployed two times to act as primary and secondary NRF.

#### UDR-NewFeatures or UDR-Regression Pipelines using Helm

Run the following commands to deploy each of the required stubs using Helm:

```
helm install --name <release_name> --namespace <namespace_name> -f <values-yaml-file> ocstub-py
```

```
helm install --name <release_name> --namespace <namespace_name> -f <fourg-stub-values-yaml-file> fourg-stub
```

```
helm install --name <release_name> --namespace <namespace_name> -f <diam-tool-
values-yaml-file> diam-stub
```

#### Example:

```
helm install --name nrfstub --namespace ocudr -f ocstub-py-custom-
values-24.1.1.yaml ocstub-py
```

```
helm install --name notify --namespace ocudr -f ocstub-py-custom-
values-24.1.1.yaml ocstub-py
```

```
helm install --name scp --namespace ocudr -f ocstub-py-custom-
values-24.1.1.yaml ocstub-py
```

```
helm install --name fourgstub --namespace ocudr -f fourg-stub-custom-
values-24.1.1.yaml fourg-stub
```

```
helm install --name diamtoola --namespace ocudr -f diam-tool-custom-
values-24.1.1.yaml diam-stub
```

```
helm install --name diamtoolb --namespace ocudr -f diam-tool-custom-
values-24.1.1.yaml diam-stub
```

## 3.8.9 Verifying ATS Deployment

To verify ATS deployment, run the following command:

```
helm status <release_name>
```

**Figure 3-22 Verifying ATS Deployment**

```
[cloud-user@cne-23-4-rc2-bastion-1 ravi]$ helm status slfats -n udr-jrk
NAME: slfats
LAST DEPLOYED: Sat Dec 9 08:04:40 2023
NAMESPACE: udr-jrk
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
Copyright 2018 (C), Oracle and/or its affiliates. All rights reserved.

Thank you for installing ocats-udr.

Your release is named slfats , Release Revision: 1.
To learn more about the release, try:

$ helm status slfats
$ helm get slfats
```

To view UDR, four UDRs, Provisioning Gateway, two NRF stubs, one bulk import and ATS deployed in the SLF namespace, run `kubectl get pods -n <ns>` command. The output is as follows:

Figure 3-23 Sample SLF Namespace

| NAME                                              | READY | STATUS    | RESTARTS | AGE   |
|---------------------------------------------------|-------|-----------|----------|-------|
| bulk-import-nudr-bulk-import-0                    | 2/2   | Running   | 0        | 2d12h |
| oso-prom-arjun-alm-0                              | 2/2   | Running   | 0        | 2d14h |
| oso-prom-arjun-alm-1                              | 2/2   | Running   | 0        | 2d14h |
| oso-prom-arjun-svr-566d9ff8b9-gmw5b               | 2/2   | Running   | 0        | 2d14h |
| provgw-auditor-service-0                          | 2/2   | Running   | 0        | 14h   |
| provgw-prov-egressgateway-5f4d994cf-s54w5         | 2/2   | Running   | 0        | 4h29m |
| provgw-prov-ingressgateway-6f9c7c85f4-rvsw8       | 2/2   | Running   | 0        | 3h41m |
| provgw-provgw-config-5649965c5c-h2fh1             | 2/2   | Running   | 0        | 2d14h |
| provgw-provgw-config-server-5f45d6f6d7-b65ld      | 2/2   | Running   | 0        | 2d14h |
| provgw-provgw-service-6976fc7bf9-7psqk            | 2/2   | Running   | 0        | 14h   |
| provgw-test                                       | 0/1   | Completed | 0        | 22h   |
| slfats-ocats-udr-559c6bb9b-fjrr4                  | 2/2   | Running   | 0        | 23h   |
| slfexport-nudr-export-tool-0                      | 2/2   | Running   | 0        | 2d12h |
| stub1-ocstub-py-64c7885669-shv9l                  | 2/2   | Running   | 0        | 2d14h |
| stub2-ocstub-py-5f7db9475c-s6gvt                  | 2/2   | Running   | 0        | 3h16m |
| udr1-alternate-route-657dcf98bf-sb5x7             | 2/2   | Running   | 0        | 4h47m |
| udr1-appinfo-f9c466857-fshtx                      | 2/2   | Running   | 0        | 4h47m |
| udr1-egressgateway-75b5556f9-8hj2k                | 2/2   | Running   | 0        | 4h47m |
| udr1-ingressgateway-prov-644df74c5d-b8qvc         | 2/2   | Running   | 0        | 2m27s |
| udr1-ingressgateway-prov-644df74c5d-fqxw6         | 2/2   | Running   | 0        | 2m20s |
| udr1-ingressgateway-sig-7d795b75b9-98jgt          | 2/2   | Running   | 0        | 4h47m |
| udr1-ingressgateway-sig-7d795b75b9-vt2vd          | 2/2   | Running   | 0        | 2m21s |
| udr1-nudr-config-74d5b6c849-ckttz                 | 2/2   | Running   | 0        | 4h47m |
| udr1-nudr-config-server-5f7bc8fb47-t65p8          | 2/2   | Running   | 0        | 4h47m |
| udr1-nudr-dr-provservice-5fccdccb75-kxnws         | 2/2   | Running   | 0        | 2m16s |
| udr1-nudr-dr-provservice-5fccdccb75-qx5hc         | 2/2   | Running   | 0        | 120m  |
| udr1-nudr-drservice-566756c5f4-d6mch              | 2/2   | Running   | 0        | 2m15s |
| udr1-nudr-drservice-566756c5f4-w466w              | 2/2   | Running   | 0        | 4h46m |
| udr1-nudr-nrf-client-nfmanagement-cd89f5974-nb9g2 | 2/2   | Running   | 0        | 4h47m |
| udr1-performance-6c566968fb-65zhm                 | 2/2   | Running   | 0        | 4h47m |
| udr1-test                                         | 0/1   | Completed | 0        | 23h   |
| udr2-alternate-route-849bb855b9-s7nn9             | 2/2   | Running   | 0        | 4h46m |
| udr2-appinfo-9fcf58c75-gxvgf                      | 2/2   | Running   | 0        | 4h46m |
| udr2-egressgateway-7d6b99fc4f-tkw6k               | 2/2   | Running   | 0        | 4h46m |
| udr2-ingressgateway-prov-ffb6d96b-dlb4v           | 2/2   | Running   | 0        | 4h46m |
| udr2-ingressgateway-sig-56c48f9478-5mkb8          | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-config-79d448476b-2sjbm                 | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-config-server-5f9c7f675b-ccw24          | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-dr-provservice-57bbfb8677-7qsdm         | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-drservice-fbd446777-dn6p9               | 2/2   | Running   | 0        | 4h46m |
| udr2-performance-7749fdc9bc-hbvkl2                | 2/2   | Running   | 0        | 4h46m |
| udr2-test                                         | 0/1   | Completed | 0        | 22h   |
| udr3-alternate-route-777886f569-mvrr4             | 2/2   | Running   | 0        | 4h46m |
| udr3-appinfo-847b54fffb8-hgqgh                    | 2/2   | Running   | 0        | 4h46m |
| udr3-egressgateway-6bf666f89b-cssdg               | 2/2   | Running   | 0        | 4h46m |
| udr3-ingressgateway-prov-777fb6f4f6-kr5l5         | 2/2   | Running   | 0        | 142m  |
| udr3-ingressgateway-sig-db75d5445-dqf5d           | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-config-5bbd58ccc5-gf9m4                 | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-config-server-f766688cc-1s5dq           | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-dr-provservice-7c8846c9cd-kfdwb         | 2/2   | Running   | 0        | 2m23s |
| udr3-nudr-drservice-5c54ff6854-wrjcc              | 2/2   | Running   | 0        | 4h45m |
| udr3-performance-7bb8f74cb9-9t66j                 | 2/2   | Running   | 0        | 4h45m |
| udr3-test                                         | 0/1   | Completed | 0        | 22h   |
| udr4-alternate-route-85f8fcd4dc-7stsm             | 2/2   | Running   | 0        | 4h45m |
| udr4-appinfo-6fcc6dddc7-69pbk                     | 2/2   | Running   | 0        | 4h45m |
| udr4-egressgateway-7d59d66db8-sq8fg               | 2/2   | Running   | 0        | 4h45m |
| udr4-ingressgateway-prov-687fdb5b-78hcr           | 2/2   | Running   | 0        | 4h45m |
| udr4-ingressgateway-sig-756f46bd5d-lvkm4          | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-config-549d676f54-hcv7p                 | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-config-server-b678f67f4-sgrrq           | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-dr-provservice-6f74dbdd87-cp9p7         | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-drservice-65dbbcc785-mgb5n              | 2/2   | Running   | 0        | 4h45m |
| udr4-performance-6b7dcd5bbc-bhxsm                 | 2/2   | Running   | 0        | 4h45m |
| udr4-test                                         | 0/1   | Completed | 0        | 22h   |

Following is the sample output of the command, 'kubectl get pods -n <ns>'. It shows OCUDR namespace with one UDR, one Provisioning Gateway, two diam-tool stubs, one http-



server stub, one SCP stub, one bulk import, two NRF stubs, one fourg-stub and ATS after installing UDR-ATS for UDR-Pipelines:

**Figure 3-24 Sample UDR Namespace**

```
[cloud-user@cne-cicd-23-2-bastion-1 ~]$ kubectl get pods -n ats-10229873
```

| NAME                                                         | READY | STATUS    | RESTARTS | AGE |
|--------------------------------------------------------------|-------|-----------|----------|-----|
| ats-10229873-ocats-udr-9cf87b687-7bd56                       | 1/1   | Running   | 0        | 25h |
| bulk-import-nudr-bulk-import-0                               | 2/2   | Running   | 0        | 24h |
| diamtoola-10229873-diam-stub-77d9c869c5-8wbjn                | 1/1   | Running   | 0        | 25h |
| diamtoolb-10229873-diam-stub-7c84658986-xvpf7                | 1/1   | Running   | 0        | 25h |
| fourgstub-10229873-fourg-stub-7b7f9ccb99-qxd44               | 1/1   | Running   | 0        | 25h |
| migration-tool-nudr-migration-tool-0                         | 1/1   | Running   | 0        | 24h |
| notifystub-10229873-ocstub-py-5d77fd9df5-fl4pf               | 1/1   | Running   | 0        | 25h |
| nrfstub1-10229873-ocstub-py-686cc65fc6-k5lgm                 | 1/1   | Running   | 0        | 25h |
| nrfstub2-10229873-ocstub-py-566b9f7864-mdqfp                 | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-alternate-route-6f6f4f78b7-thd9n              | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-appinfo-65c5b488b4-rnpvh                      | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-appinfo-65c5b488b4-xqq6c                      | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-egressgateway-59d4846bcb-6gfmm                | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-ingressgateway-prov-5478b666bd-lxmrc          | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-ingressgateway-sig-64d59bfdbd-p2d87           | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-config-85f58f87d4-85sd4                  | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-config-server-56f9645cbd-lf8ds           | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-config-server-56f9645cbd-whnqj           | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-diam-gateway-0                           | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-diameterproxy-5bdbd84c56-bb2jq           | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-dr-provservice-7b76c8ffb9-kgxmv          | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-drservice-56c65f44c6-wm7bx               | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-notify-service-7b78448795-d98zq          | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-nrf-client-nfmanagement-67c6596dbc-mswdx | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-nudr-ondemand-migration-757b4cfb95-5b9m7      | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-performance-59784bc899-9qbst                  | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-performance-59784bc899-xb7wf                  | 1/1   | Running   | 0        | 25h |
| ocudr-10229873-test                                          | 0/1   | Completed | 0        | 24h |
| provgw-prov-egressgateway-5679d6ddf-m69cl                    | 1/1   | Running   | 0        | 24h |
| provgw-prov-ingressgateway-84d6fc9876-zmbln                  | 1/1   | Running   | 0        | 24h |
| provgw-provgw-config-85f8bddcb9-4gnlm                        | 1/1   | Running   | 0        | 24h |
| provgw-provgw-config-server-7bffc5d665-cpwhv                 | 1/1   | Running   | 0        | 24h |
| provgw-provgw-config-server-7bffc5d665-wktdj                 | 1/1   | Running   | 0        | 24h |
| provgw-provgw-service-5d569c4447-4bgjd                       | 1/1   | Running   | 0        | 24h |
| scpstub-10229873-ocstub-py-75fb9c98b6-w82n7                  | 1/1   | Running   | 0        | 25h |
| udrexport-nudr-export-tool-0                                 | 1/1   | Running   | 0        | 24h |

Following is the sample output of the command, 'kubectl get pods -n <ns>' - It shows OCUDR namespace with one UDR, two nrf stubs, one bulk import and ATS after installation for EIR pipeline:

**Figure 3-25 Sample EIR Namespace**

```
[cloud-user@cne-cicd-223-bastion-1 ~]$ kubectl get pods -n eir-5801444
```

| NAME                                                        | READY | STATUS  | RESTARTS | AGE |
|-------------------------------------------------------------|-------|---------|----------|-----|
| bulk-import-nudr-bulk-import-0                              | 2/2   | Running | 0        | 19m |
| eirats-ocats-udr-794b8bd9f8-m7pjb                           | 1/1   | Running | 0        | 26m |
| export-tool-nudr-export-tool-0                              | 1/1   | Running | 0        | 19m |
| oceir-5801444-alternate-route-5c5c44c9db-dbgrr              | 1/1   | Running | 0        | 21m |
| oceir-5801444-appinfo-7c9747d-zkj94                         | 1/1   | Running | 0        | 21m |
| oceir-5801444-egressgateway-787f8fc947-nxqgm                | 1/1   | Running | 0        | 21m |
| oceir-5801444-ingressgateway-prov-fcf49477d-kf4k8           | 2/2   | Running | 0        | 21m |
| oceir-5801444-ingressgateway-sig-7b9bb5c687-9gfjr           | 2/2   | Running | 0        | 21m |
| oceir-5801444-nudr-config-f5b6bb68f-gkqx6                   | 1/1   | Running | 0        | 21m |
| oceir-5801444-nudr-config-server-7b764d6d9c-rdn8g           | 1/1   | Running | 0        | 21m |
| oceir-5801444-nudr-dr-provservice-6c4cdc6cc6-scxjt          | 1/1   | Running | 0        | 21m |
| oceir-5801444-nudr-drservice-85bf7c5b9d-kwpjf               | 1/1   | Running | 0        | 21m |
| oceir-5801444-nudr-nrf-client-nfmanagement-6fdd454844-t7ndl | 1/1   | Running | 0        | 21m |
| oceir-5801444-performance-bbb875b9b-b69cs                   | 1/1   | Running | 0        | 21m |
| oceir-5801444-performance-bbb875b9b-pcst5                   | 1/1   | Running | 0        | 21m |
| stub1-nrf-ocstub-python-fff7b85fb-sh9rn                     | 1/1   | Running | 0        | 26m |
| stub2-nrf-ocstub-python-7d756758f-f2hhd                     | 1/1   | Running | 0        | 26m |

If you have installed ATS with sidecar, ensure ATS shows two containers in the READY state as "2/2". A sample output of the command, 'kubectl get pods -n <ns>' for SLF-Pipelines is as follows:

Figure 3-26 ATS Deployed with Sidecar

| NAME                                              | READY | STATUS    | RESTARTS | AGE   |
|---------------------------------------------------|-------|-----------|----------|-------|
| bulk-import-nudr-bulk-import-0                    | 2/2   | Running   | 0        | 2d12h |
| oso-prom-arjun-alm-0                              | 2/2   | Running   | 0        | 2d14h |
| oso-prom-arjun-alm-1                              | 2/2   | Running   | 0        | 2d14h |
| oso-prom-arjun-svr-566d9ff8b9-gmw5b               | 2/2   | Running   | 0        | 2d14h |
| provgw-auditor-service-0                          | 2/2   | Running   | 0        | 14h   |
| provgw-prov-egressgateway-5f4d994cf-s54w5         | 2/2   | Running   | 0        | 4h29m |
| provgw-prov-ingressgateway-6f9c7c85f4-rvsw8       | 2/2   | Running   | 0        | 3h41m |
| provgw-provgw-config-5649965c5c-h2fh1             | 2/2   | Running   | 0        | 2d14h |
| provgw-provgw-config-server-5f45d6f6d7-b65ld      | 2/2   | Running   | 0        | 2d14h |
| provgw-provgw-service-6976fc7bf9-7psqk            | 2/2   | Running   | 0        | 14h   |
| provgw-test                                       | 0/1   | Completed | 0        | 22h   |
| slfats-ocats-udr-559c6bb9b-fjrr4                  | 2/2   | Running   | 0        | 23h   |
| slfexport-nudr-export-tool-0                      | 2/2   | Running   | 0        | 2d12h |
| stub1-ocstub-py-64c7885669-shv91                  | 2/2   | Running   | 0        | 2d14h |
| stub2-ocstub-py-5f7db9475c-s6gvt                  | 2/2   | Running   | 0        | 3h16m |
| udr1-alternate-route-657dcf98bf-sb5x7             | 2/2   | Running   | 0        | 4h47m |
| udr1-appinfo-f9c466857-fshtx                      | 2/2   | Running   | 0        | 4h47m |
| udr1-egressgateway-75b5556f9-8hj2k                | 2/2   | Running   | 0        | 4h47m |
| udr1-ingressgateway-prov-644df74c5d-b8qvc         | 2/2   | Running   | 0        | 2m27s |
| udr1-ingressgateway-prov-644df74c5d-fqxx6         | 2/2   | Running   | 0        | 2m20s |
| udr1-ingressgateway-sig-7d795b75b9-98jgt          | 2/2   | Running   | 0        | 4h47m |
| udr1-ingressgateway-sig-7d795b75b9-vt2vd          | 2/2   | Running   | 0        | 2m21s |
| udr1-nudr-config-74d5b6c849-ckttz                 | 2/2   | Running   | 0        | 4h47m |
| udr1-nudr-config-server-5f7bc8fb47-t65p8          | 2/2   | Running   | 0        | 4h47m |
| udr1-nudr-dr-provservice-5fccdccb75-kxnws         | 2/2   | Running   | 0        | 2m16s |
| udr1-nudr-dr-provservice-5fccdccb75-qx5hc         | 2/2   | Running   | 0        | 120m  |
| udr1-nudr-dr-service-566756c5f4-d6mch             | 2/2   | Running   | 0        | 2m15s |
| udr1-nudr-dr-service-566756c5f4-w466w             | 2/2   | Running   | 0        | 4h46m |
| udr1-nudr-nrf-client-nfmanagement-cd89f5974-nb9g2 | 2/2   | Running   | 0        | 4h47m |
| udr1-performance-6c566968fb-65shh                 | 2/2   | Running   | 0        | 4h47m |
| udr1-test                                         | 0/1   | Completed | 0        | 23h   |
| udr2-alternate-route-849bb855b9-s7nn9             | 2/2   | Running   | 0        | 4h46m |
| udr2-appinfo-9fcf58c75-gxvgf                      | 2/2   | Running   | 0        | 4h46m |
| udr2-egressgateway-7d6b99fc4f-tkw6k               | 2/2   | Running   | 0        | 4h46m |
| udr2-ingressgateway-prov-ffb6d96b-dlb4v           | 2/2   | Running   | 0        | 4h46m |
| udr2-ingressgateway-sig-56c48f9478-5mkb8          | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-config-79d448476b-2sajbm                | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-config-server-5f9c7f675b-ccw24          | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-dr-provservice-57bbfb8677-7qsdx         | 2/2   | Running   | 0        | 4h46m |
| udr2-nudr-dr-service-fbd446777-dn6p9              | 2/2   | Running   | 0        | 4h46m |
| udr2-performance-7749fdc9bc-hbvkk2                | 2/2   | Running   | 0        | 4h46m |
| udr2-test                                         | 0/1   | Completed | 0        | 22h   |
| udr3-alternate-route-777886f569-mvrr4             | 2/2   | Running   | 0        | 4h46m |
| udr3-appinfo-847b54f5fb8-hgqgh                    | 2/2   | Running   | 0        | 4h46m |
| udr3-egressgateway-6bf666f89b-cssdg               | 2/2   | Running   | 0        | 4h46m |
| udr3-ingressgateway-prov-777fb6f4f6-kr515         | 2/2   | Running   | 0        | 142m  |
| udr3-ingressgateway-sig-db75d5445-dqf5d           | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-config-5bbd58ccc5-gf9m4                 | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-config-server-f766688cc-ls5dq           | 2/2   | Running   | 0        | 4h46m |
| udr3-nudr-dr-provservice-7c8846c9cd-kfdwb         | 2/2   | Running   | 0        | 2m23s |
| udr3-nudr-dr-service-5c54ff6854-wrjcc             | 2/2   | Running   | 0        | 4h45m |
| udr3-performance-7bb8f74cb9-9t66j                 | 2/2   | Running   | 0        | 4h45m |
| udr3-test                                         | 0/1   | Completed | 0        | 22h   |
| udr4-alternate-route-85f8fcd4dc-7stsm             | 2/2   | Running   | 0        | 4h45m |
| udr4-appinfo-6fcc6dddc7-69pbk                     | 2/2   | Running   | 0        | 4h45m |
| udr4-egressgateway-7d59d66db8-sq8fg               | 2/2   | Running   | 0        | 4h45m |
| udr4-ingressgateway-prov-687fdb5b-78hcr           | 2/2   | Running   | 0        | 4h45m |
| udr4-ingressgateway-sig-756f46bd5d-lvkm4          | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-config-549d676f54-hcv7p                 | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-config-server-b678f67f4-sgrxq           | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-dr-provservice-6f74dbdd87-cp9p7         | 2/2   | Running   | 0        | 4h45m |
| udr4-nudr-dr-service-65dbbcc785-mgb5n             | 2/2   | Running   | 0        | 4h45m |
| udr4-performance-6b7dcd5bbc-bhxsm                 | 2/2   | Running   | 0        | 4h45m |
| udr4-test                                         | 0/1   | Completed | 0        | 22h   |

## 3.8.10 Post Installation Steps

### If Provisioning Gateway is upgraded using helm upgrade command

Following are the post installation steps:

1. Perform the post installation steps as described in Cloud Native Core Provisioning Gateway Guide to change segDetails to UDR Provisioning Ingress Gateway fqdn and port. For more information, see *Cloud Native Core Provisioning Gateway Guide*.

2. Run the following command:

```
kubectl exec -it -n <ns> <ATS pod> bash
```

3. Run the following command:

```
curl -X PUT http://<provgw helm release>-provgw-config.<ns>:5001/provgw-config/v1/udr.provgwservice.cfg/PROVGW-SERVICE -d '{"tracingEnabled": false, "soapService_udrIp": "ocudr-ingressgateway-prov.ocudr", "soapService_udrSignallingIp": "ocudr-ingressgateway-sig.ocudr", "retryErrorCodes": [500, 503], "retryCount": 3, "retryInterval": 2}'
```

4. Run the exit command.

### If PV (Persistent Volume) is enabled for UDR ATS

Following are the post installation steps:

1. Run the following command to extract the ocslf\_tests (for SLF Pipelines), ocudr\_tests (for UDR Pipelines), or oceir\_tests (for EIR Pipeline) and jobs from ocats-udr-data-24.1.1.tgz..  
tar -xvf ocats-udr-data-24.1.1.tgz
2. Run the following command to create certs and oauth\_keys in ocslf\_tests folder (for SLF pipeline runs), oceir\_tests (for EIR pipeline runs), or ocudr\_tests (for UDR pipeline runs):

```
mkdir -p ocslf_tests/certs ocslf_tests/oauth_keys
```

```
mkdir -p oceir_tests/certs oceir_tests/oauth_keys
```

```
mkdir -p ocudr_tests/certs ocudr_tests/oauth_keys
```

3. Run the following commands to copy the ocslf\_tests and jobs folder to the ATS pod only if it is intended to run SLF Pipelines.

```
kubectl cp ocslf_tests <namespace>/<pod-name>:/var/lib/jenkins
kubectl cp jobs <namespace>/<pod-name>:/var/lib/jenkins/.jenkins
```

4. Run the following commands to copy the ocudr\_tests and jobs folder to the ATS pod only if it is intended to run UDR Pipelines.

```
kubectl cp ocudr_tests <namespace>/<pod-name>:/var/lib/jenkins
kubectl cp jobs <namespace>/<pod-name>:/var/lib/jenkins/.jenkins
```

5. Run the following commands to copy the oceir\_tests and jobs folder to the ATS pod only if it is intended to run EIR Pipelines.

```
kubectl cp oceir_tests <namespace>/<pod-name>:/var/lib/jenkins
kubectl cp jobs <namespace>/<pod-name>:/var/lib/jenkins/.jenkins
```

6. Run the following command to restart the pod:  
kubectl delete pod <pod-name> -n <namespace>

7. For SLF-NewFeatures and SLF-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private keys used to create secret for TLS support on Provisioning Gateway to ocsf\_tests/certs folder in the path as follows:

**Note**

Provisioning Gateway must use all three files as part of TLS support.

- a. kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/ocsf\_tests/certs  
**Example:** kubectl cp caroot.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocsf\_tests/certs
- b. kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/ocsf\_tests/certs  
**Example:** kubectl cp apigatewayrsa.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocsf\_tests/certs
- c. kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/ocsf\_tests/certs  
**Example:** kubectl cp rsa\_private\_key\_pkcs1.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocsf\_tests/certs

**Note**

Provisioning Gateway should use the above three files as part of TLS support.

8. For UDR-NewFeatures and UDR-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private key that are used to create secret for TLS support on Provisioning Gateway to ocudr\_tests or ocudr\_certs folder in the path as follows:

**Note**

For TLS validation, use the same set of copied certificates for Provisioning Gateway Ingress Gateway, UDR Ingress Gateway, and UDR Egress Gateway.

- a. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp caroot.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`
  - b. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp apigatewayrsa.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`
  - c. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp rsa_private_key_pkcs1.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`
9. For EIR-NewFeatures and EIR-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private key that are used to create secret for TLS support on UDR ingressgateway-prov and ingressgateway-sig to oceir\_tests/certs folder in the path as follows:
- a. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp caroot_sig.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - b. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp apigatewayrsa_sig.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - c. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp rsa_private_key_pkcs1_sig.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - d. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp caroot_prov.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - e. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp apigatewayrsa_prov.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - f. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp rsa_private_key_pkcs1_prov.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
10. To run Oauth2 validation scenarios on SLF-Regression, copy each private keys in pem format to oauth\_keys folder in /var/lib/jenkins/ocslf\_tests path.

```
kubectl cp <private key pem file> <namespace>/<pod-name>:/var/lib/jenkins/ocslf_tests/oauth_keys
```



Example:

```
kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/ocslf_tests/oauth_keys
```

11. To run OAuth2 validation scenarios on UDR-Regression, copy each private keys in pem format to oauth\_keys folder in /var/lib/jenkins/ocudr\_tests path.

```
kubectl cp <private key pem file> <namespace>/<pod-name>:/var/lib/jenkins/
ocudr_tests/oauth_keys
```

Example:

```
kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/oauth_keys
```

12. To run OAuth2 validation scenarios on EIR-Regression, copy each private keys in pem format to oauth\_keys folder in /var/lib/jenkins/oceir\_tests path.

```
kubectl cp <private key pem file> <namespace>/<pod-name>:/var/lib/jenkins/
oceir_tests/oauth_keys
```

Example:

```
kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/oauth_keys
```

#### If PV is disabled:

Following are the post installation steps:

1. For SLF-NewFeatures and SLF-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private key used to create secret for TLS support on Provisioning Gateway to ocslf\_tests/certs folder in the path as follows:

#### Note

Provisioning Gateway must use all three files as part of TLS support.

- a. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/ocslf_tests/certs`  
**Example:** `kubectl cp caroot.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocslf_tests/certs`
- b. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/ocslf_tests/certs`  
**Example:** `kubectl cp apigatewayrsa.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocslf_tests/certs`
- c. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/ocslf_tests/certs`

**Example:** `kubectl cp rsa_private_key_pkcs1.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocslf_tests/certs`

2. For UDR-NewFeatures and UDR-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private keys used to create secret for TLS support on Provisioning Gateway to `ocslf_tests/certs` folder in the path as follows:

**Note**

For TLS validation, use the same set of copied certificates for Provisioning Gateway Ingress Gateway, UDR Ingress Gateway, and UDR Egress Gateway.

- a. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp caroot.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`
- b. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp apigatewayrsa.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`
- c. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/certs`  
**Example:** `kubectl cp rsa_private_key_pkcs1.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/certs`

**Note**

The Ingress Gateway of UDR should use the above three files as part of TLS support.

3. For EIR-NewFeatures and EIR-Regression pipelines, copy the root certificate authority (CA), signed server certificate with root CA private key, and private keys used to create secret for TLS support on UDR ingressgateway-prov and ingressgateway-sig to `oceir_tests/certs` folder in the path as follows:
  - a. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp caroot_sig.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - b. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp apigatewayrsa_sig.cer ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - c. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`  
**Example:** `kubectl cp rsa_private_key_pkcs1_sig.pem ocudr/ats-ocats-udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`
  - d. `kubectl cp <root CA> <namespace>/<pod-name>:/var/lib/jenkins/oceir_tests/certs`

**Example:** `kubectl cp caroot_prov.cer ocudr/ats-ocats-  
udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`

- e. `kubectl cp <server certificate> <namespace>/<pod-name>:/var/lib/jenkins/  
oceir_tests/certs`

**Example:** `kubectl cp apigatewayrsa_prov.cer ocudr/ats-ocats-  
udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`

- f. `kubectl cp <private key> <namespace>/<pod-name>:/var/lib/jenkins/  
oceir_tests/certs`

**Example:** `kubectl cp rsa_private_key_pkcs1_prov.pem ocudr/ats-ocats-  
udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/certs`

4. To run OAuth2 validation scenarios on SLF-Regression, copy each private keys in pem format to `oauth_keys` folder in `/var/lib/jenkins/ocslf_tests` path.

```
kubectl cp <private key pem file>
 <namespace>/<pod-name>:/var/lib/jenkins/ocslf_tests/oauth_keys
```

**Example:**

```
kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/ocslf_tests/oauth_keys
```

5. To run OAuth2 validation scenarios on UDR-Regression, copy each private keys in pem format to `oauth_keys` folder in `/var/lib/jenkins/ocudr_tests` path.

```
kubectl cp <private key pem file>
 <namespace>/<pod-name>:/var/lib/jenkins/ocudr_tests/oauth_keys
```

**Example:**

```
kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/ocudr_tests/oauth_keys
```

6. To run OAuth2 validation scenarios on EIR-Regression, copy each private keys in pem format to `oauth_keys` folder in `/var/lib/jenkins/oceir_tests` path.

```
kubectl cp <private key pem file> <namespace>/<pod-name>:/var/lib/jenkins/
oceir_tests/oauth_keys
```

**Example:**

```
Example: kubectl cp ec_private_key1.pem ocudr/ats-ocats-
udr-696df7c84d-2qc7h:/var/lib/jenkins/oceir_tests/oauth_keys
```



# 4

## Running NF Test Cases Using ATS

This section describes how to run NF test cases using ATS. It includes:

- [Running BSF Test Cases using ATS](#)
- [Running NEF Test Cases using ATS](#)
- [Running NRF Test Cases using ATS](#)
- [Running NSSF Test Cases using ATS](#)
- [Running Policy Test Cases using ATS](#)
- [Running SCP Test Cases using ATS](#)
- [Running SEPP Test Cases using ATS](#)
- [Running UDR Test Cases using ATS](#)

### 4.1 Running BSF Test Cases using ATS

This section describes how to run BSF test cases using ATS.

#### Note

Restart the NRF-client pod of BSF for UDR and CHF discovery as part of each test case.

#### 4.1.1 Prerequisites

To run Binding Support Function test cases, ensure that the following prerequisites are met:

- Deploy BSF 24.1.0 with default helm configurations using helm charts to run all test cases. The ATS version must be compatible with BSF 24.1.0. For more information on how to install BSF, see *Oracle Communications Cloud Native Core, Binding Support Function Installation, Upgrade, and Fault Recovery Guide*.
- Go-STUB must be installed in the same namespace where ocbf is installed.
- Add the following to Kubernetes namespace to grant role access:

PolicyRule:

| Resources                  | Non-Resource URLs | Resource Names | Verbs      |
|----------------------------|-------------------|----------------|------------|
| -----                      | -----             | -----          | -----      |
| pods/log                   | [ ]               | [ ]            | [get list] |
| configmaps                 | [ ]               | [ ]            | [watch get |
| list delete update create] |                   |                |            |
| pods                       | [ ]               | [ ]            | [watch get |
| list delete update create] |                   |                |            |
| secrets                    | [ ]               | [ ]            | [watch get |
| list delete update create] |                   |                |            |

|                            |    |    |            |
|----------------------------|----|----|------------|
| services                   | [] | [] | [watch get |
| list delete update create] |    |    |            |
| deployments.apps           | [] | [] | [watch get |
| list update]               |    |    |            |
| replicasets.apps           | [] | [] | [watch get |
| list update]               |    |    |            |
| deployments.extensions     | [] | [] | [watch get |
| list update]               |    |    |            |
| replicasets.extensions     | [] | [] | [watch get |
| list update]               |    |    |            |

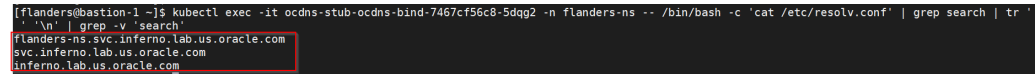
- ATS Prometheus metrics validation works only when:
  - the metrics suffixes are not configured
  - installation has a single pod for each microservice in the BSF deployment
- You can customize test cases in the custom test case folders (cust\_newfeatures, cust\_regression and cust\_performance). You can add new test cases, remove unwanted test cases and modify existing test cases. It does not impact the original product packaged test cases available in the newfeatures, regression and performance folders respectively. For more details about custom test case folders, see [Custom Folder Implementation](#).
- Install Prometheus server in the cluster.
- Database cluster is in the running state with all required tables. Verify that there are no previous entries in the database before running test cases.
- Do not initiate a job in two different pipelines at the same time.
- For using Controlled Shutdown feature, ensure that enableControlledShutdown parameter is enabled on BSF. If this parameter is not enabled, the test case for this feature will fail.
- In the application-config configmap, configure the following parameters with the respective values:
  - primaryNrfApiRoot=nf1stub.<namespace\_gostubs\_are\_deployed\_in>.svc:8080  
For example:  
primaryNrfApiRoot=nf1stub.ocats.svc:8080
  - secondaryNrfApiRoot=nf11stub.<namespace\_gostubs\_are\_deployed\_in>.svc:8080  
For example:  
secondaryNrfApiRoot=nf11stub.ocats.svc:8080
  - virtualNrfFqdn = nf1stub.<namespace\_gostubs\_are\_deployed\_in>.svc  
For example:  
virtualNrfFqdn=nf1stub.ocats.svc
- Before running ATS Test suite, restart nrf-client-nfmanagement pod.
  - Run the following command to get all the configmaps in your namespace.  
  
kubectll get configmaps -n <BSF\_namespace>
  - Edit the alternate route service deployment pointing towards DNS Stub.

Run the following command to get searches information from dns-bind pod to enable communication between Alternate Route and dns-bind service.

```
kubectl exec -it <dns-bind pod> -n <NAMESPACE> -- /bin/bash -c
'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
```

Example:

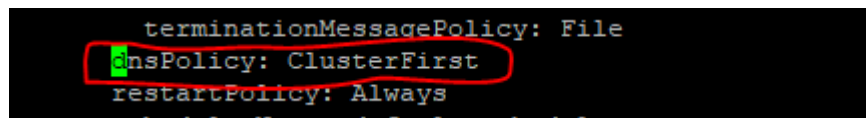
**Figure 4-1 Editing Alternate Route Service deployment pointing towards DNS Stub**



```
[flanders@bastion-1 ~]$ kubectl exec -it ocdns-stub-ocdns-bind-7467cf56c8-5dqq2 -n flanders-ns -- /bin/bash -c 'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
flanders-ns.svc.inferno.lab.us.oracle.com
svc.inferno.lab.us.oracle.com
inferno.lab.us.oracle.com
```

By default, Alternate Route Service points to CoreDNS.

**Figure 4-2 Alternate Route Service settings in deployment file**



```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

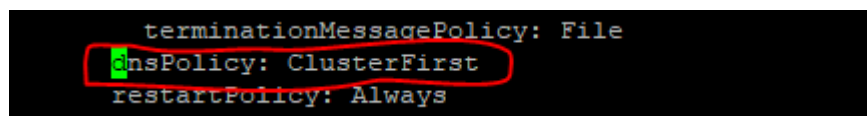
Change the deployment file to add following content in alternate service to query DNS stub.

```
kubectl edit deployment ocsbf-occnp-alternate-route -n ocsbf
```

- \* Add the nameservers IPAddress.
- \* Add all the search information.
- \* Set dnsPolicy to None.

```
dnsConfig:
 nameservers:
 - <dns_stub_cluster_ip_address>
 searches:
 - dns-bind search
 - dns-bind search
 - dns-bind search
 dnsPolicy: None
```

**Figure 4-3 dnsConfig**



```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

- If Service Mesh check is enabled, create a destination rule to fetch the metrics from the Prometheus. For destination rule to communicate between TLS enabled entity (ATS) and non-TLS entity (Prometheus), Prometheus is kept outside of the service mesh. The rule can be created as follows:

```
kubectl apply -f - <<EOF

apiVersion:networking.istio.io/v1alpha3
kind:DestinationRule
metadata:
 name:prometheus-dr
 namespace:ocats
spec:
 host:oso-prometheus-server.ocbsf.svc.cluster.local
 trafficPolicy:
 tls:
 mode:DISABLE
EOF
```

- \* name: indicates the name of destination rule.
- \* namespace: indicates where the ATS is deployed.
- \* host: indicates the hostname of the prometheus server.
- For Bsf\_To\_Nrf\_Late\_Arrival feature file to be executed, nfinstanced should be configured in Bsf\_To\_Nrf\_Late\_Arrival.yaml parameterization file. This nfinstanced must be the same as the one under BSF nfinstanced, that is configured during BSF installation. The nfinstanced must to be configured in ATS UI under pipeline configuration (BSF\_NFINSTANCE\_ID).
- To ensure the consistent functioning of ATS related to the audit service, modify the Audit deployment by reducing the value of AUDIT\_NOTIFY\_SCHEDULER\_POLLING\_INTERVAL\_MILLISEC from 30000 to 1000 (changing from 30 seconds to 1 second).

```
$ kubectl get deploy -n ocbsf | grep 'audit'
ocbsf-ocpm-audit-service 1/1 1
1 10h
```

```
$ kubectl edit deploy ocbsf-ocpm-audit-service -n ocbsf
```

The deployment will open. Scroll down until below fields are seen

```
- name: AUDIT_NOTIFY_SCHEDULER_POLLING_INTERVAL_MILLISEC
value: "30000"
```

Update the value of AUDIT\_NOTIFY\_SCHEDULER\_POLLING\_INTERVAL\_MILLISEC to 1000 if it isn't already.

## 4.1.2 Logging into ATS

Before logging into ATS GUI, it is important to get the worker Node External IP and nodeport of the service, 'ocats-bsf'.

Run the following command to get the external IP for worker node:

**Example:**

```
kubectl get nodes -owide
```

On running the abovementioned command, the following is the output:

```
ocbsf-k8s-node-1 Ready <none> 111d v1.16.7 192.168.200.26
10.75.152.111 Oracle Linux Server 7.8
4.14.35-1902.303.5.3.el7uek.x86_64 containerd://1.2.10
```

Run the following command to get the nodeport:

```
kubectl get svc -n <BSF_namespace>
```

**Example:**

```
kubectl get svc -n ocbsf
```

The output of the command is:

```
ocbsf-ocats-ocats-bsf LoadBalancer 10.233.53.144 10.75.225.49
8080:31944/TCP 19h
```

To log in to ATS, open the web browser and type the following URL:

`http://<Worker-Node-IP>:<Node-Port-of-ATS>`

If the ocats-bsf Service has an external IP available, <SVC external IP> can also be used to log in to ATS.

`http://<External IP of ATS Service>:8080`

**Example:**

`http://10.75.225.49:8080`

## Running ATS

To run ATS test cases, perform the following steps:

1. Enter the **username** as bsfuser and **password** as bsfpasswd.
2. Click **Sign in**.

**Note**

To modify default login password, see [Modifying Login Password](#).

**Figure 4-4 Pre-configured Pipelines**

| S   | W | Name            | Last Success  | Last Failure | Last Duration |   | Fav |
|-----|---|-----------------|---------------|--------------|---------------|---|-----|
| ... | ☀ | BSF_HealthCheck | N/A           | N/A          | N/A           | ▶ | ★   |
| ✓   | ☀ | BSF_NewFeatures | 1 day 4 hr #1 | N/A          | 6 min 31 sec  | ▶ | ★   |
| ✓   | ☀ | BSF_Regression  | 1 day 7 hr #1 | N/A          | 2 hr 5 min    | ▶ | ★   |

cont: S M L    Icon legend    📡 Atom feed for all    📡 Atom feed for failures    📡 Atom feed for just latest builds

On successful log in, users should see the following pipelines:

- **BSF-NewFeatures:** This pipeline has all the new test cases delivered for BSF 24.1.0.
- **BSF-Performance:** This pipeline is not operational as of now. It is reserved for future releases of ATS.
- **BSF-HealthCheck:** This pipeline checks if BSF and ATS are deployed correctly. This shows only when the user has enabled this feature at the time of installing BSF ATS.
- **BSF-Regression:** This pipeline has all the test cases delivered in BSF ATS - 23.4.0

### 4.1.3 Running BSF\_NewFeatures Pipeline

#### BSF\_NewFeatures Pipeline

This is a pre-configured pipeline where users can run all the BSF new test cases. To configure its parameters, which is a one time activity, perform the following steps:

1. Click **BSF\_NewFeatures** in the Name column and then, click **Configure** in the left navigation pane as shown below:

**Figure 4-5 BSF New Feature Pipeline**

|                                                            | Preparation | Execute-Tests | 1st Stage | 1st Stage 1st Group | 1st Stage 2nd Group | 1st Stage 3rd Group | 2nd Stage | 2nd Stage 1st Group | 2nd Stage 2nd Group | 2nd Stage 3rd Group | 3rd Stage | 3rd Stage 1st Group | 3rd Stage 2nd Group |
|------------------------------------------------------------|-------------|---------------|-----------|---------------------|---------------------|---------------------|-----------|---------------------|---------------------|---------------------|-----------|---------------------|---------------------|
| Average stage times:<br>(Average full run time: ~6min 31s) | 6s          | 35ms          | 3s        | 5min 5s             | 4s                  | 4s                  | 105ms     | 4s                  | 4s                  | 4s                  | 100ms     | 4s                  | 4s                  |
| Dec 14 10:23 No Changes                                    | 6s          | 35ms          | 3s        | 5min 5s             | 4s                  | 4s                  | 105ms     | 4s                  | 4s                  | 4s                  | 100ms     | 4s                  | 4s                  |

2. The BSF\_NewFeatures, **General** tab appears. Make sure that the screen loads completely.
3. Scroll-down to the end. The control moves from **General** tab to the **Pipeline** tab as shown below:

Script ?

```

7 //t = TIME_INT_POD_UP u = POD_UP_RETRIES v = ELK_WAIT_TIME w = ELK_HOST
8 //x = ELK_PORT y = STUB_LOG_COLLECTION z = LOG_METHOD A = svc_cfg_to_be_read B = bulk_import_to_complete
9
10 //Description of Variables:
11
12 //SELECTED_NF : BSF
13 //BSF_NAMESPACE : BSF Namespace:
14 //PROMSVC_NAME : Prometheus Server Service name
15 //GOSTUB_NAMESPACE : Gostub namespace
16 //SECURITY : secure or unsecure
17 //BSF_NFINSTANCE_ID : nfInstanceId in BSF application-config config map
18 //POD_RESTART_TIME : Greater or equal to 45
19 //BSF_TIME : Greater or equal to 60
20 //NF_NOTIF_TIME : Greater or equal to 140
21 //PEER_CONNECTION_ESTABLISHMENT : Greater or equal to 30
22 //RERUN_COUNT : Rerun failing scenario count
23 //TIME_INT_POD_DOWN : The interval after which we check the POD status if its down

```

You can change the parameter values from "a" - to - "x" as per user requirement. The parameter details are available as comments from line number 2 - to - 7. In the **Script** area of the Pipeline section, you can change the values of the following parameters:

- **b:** Change this parameter to update the namespace where BSF was deployed in your bastion.
- **d:** Change this parameter to update the namespace where your gostubs are deployed in your bastion.
- **e:** Set this parameter as 'unsecure', if you intend to run ATS in TLS disabled mode. Else, set this parameter as 'secure'.
- **f:** Configure this parameter to set BSF\_NFINSTANCE\_ID.
- **g:** Set a value more than 45 seconds for this parameter. The default wait time for the pod to come up is 45 seconds. Every TC requires restart of the nrf-client-management pod.
- **h:** Set a value more than 60 seconds for this parameter. The default wait time to add a configurations to the database is 60 secs.
- **i:** Set this parameter to more than 140 secs. The default wait time for Nf\_Notification Test Cases is given as 140 secs.
- **k:** Use this parameter to set the waiting time to initialize Test Suite.
- **l:** Use this parameter to set the waiting time to get response from Stub.
- **m:** Use this parameter to set the waiting time after adding BSF Configuration.
- **n:** Use this parameter to set the waiting time for Peer connection establishment.
- **o:** Use this parameter to set the waiting time before sending next message.
- **p:** Use this parameter to set Prometheus Server IP.
- **q:** Use this parameter to set Prometheus Server Port.
- **r:** Use this parameter to set the interval after which the POD status is checked when it is down.
- **s:** Use this parameter to set number of retry attempt in which will check the pod down status
- **t:** Use this parameter to set the interval after which we check the POD status if its UP.
- **u:** Use this parameter to set number of retry attempt in which will check the pod up status.
- **v:** Use this parameter to set Wait time to connect to Elastic Search.

- **w:** Use this parameter to set Elastic Search HostName.
- **x:** Use this parameter to set Elastic Search Port.
- **y:** Use this parameter to set To Enable/Disable Stub logs collection.
- **z:** Use this parameter to set Log collection endpoint either Elasticsearch or Kubernetes.
- **A:** Use this parameter to set Timer to wait for importing service configurations.
- **B** Use `bulk_import_to_complete` to add custom time in Jenkins post bulk imports.

(Optional) To collect application logs per failed scenario, user can configure the values for the following parameter:

- **z:** If you want log collection to happen through Elastic search, set the value for this parameter as `Elasticsearch`. If not, specify the value as `Kubernetes`. If you want to collect logs through Elastic search, it is required to configure the values for the following parameters:
    - **v:** Specifies the wait time to connect to Elastic search (`ELK_WAIT_TIME`).
    - **w:** Specifies the host name of Elastic search (`ELK_HOST`). For example, `ocne-elastic-elasticsearch-master.ocne-infra/`
    - **x:** Specifies the port for Elastic search (`ELK_PORT`). For example, `9200`.
  - **y:** If you want to collect stub logs, set the value for this parameter as `yes`. If not, specify the value as `no`.
4. Click **Save** after updating the parameter values. The **BSF\_NewFeatures** Pipeline page appears.

**Note**

It is recommended to save a copy of the pipeline script in your local machine that you may refer while restarting ATS pods.

**NOT\_SUPPORTED**

Do not modify anything other than the parameter values described in this section.

### Extracting Application Logs

If any scenarios fail after running the pipeline job, perform the following steps:

1. Log in to the ATS pod:

```
kubectl exec -it pod/ocats-bsf-6f6dfc76b5-jbgzt -n ocbsf bash
```

2. Go to Jenkins build directory:

```
cd /var/lib/jenkins/.jenkins/jobs/BSF_Regression/builds/2
```

For example:

```
cd $JENKINS_HOME/jobs/$JOB_NAME/builds/$BUILD_NUMBER/
```



3. Extract the `applogs.zip` file:

```
unzip applogs.zip
```

4. After successfully unzipping the file, open the `applog` folder to view the pod logs for failed scenarios:

```
(env) [jenkins@ocats-bsf-6f6dfc76b5-jbgzt applog]$ pwd
/var/lib/jenkins/.jenkins/jobs/BSF_Regression/builds/2/applog
(env) [jenkins@ocats-bsf-6f6dfc76b5-jbgzt applog]$ ls -ltrh
total 760K
-rw-r--r--. 1 jenkins jenkins 250K Nov 19 11:08 Initial_Run-
Register_BSF_With_NFSetIDList.log
-rw-r--r--. 1 jenkins jenkins 249K Nov 19 11:08 1st_Rerun-
Register_BSF_With_NFSetIDList.log
-rw-r--r--. 1 jenkins jenkins 255K Nov 19 11:09 2nd_Rerun-
Register_BSF_With_NFSetIDList.log
```

## Running BSF Test Cases

To run BSF test cases, perform the following steps:

1. Click the **Build with Parameters** link available in the left navigation pane of the **BSF-NewFeatures** Pipeline screen. The following page appears.

**Figure 4-6 BSF New Features**

The screenshot displays the 'Oracle Communications Cloud Native Core' interface for the 'Automated Test Suite - BSF'. It is divided into two main sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'.

**EXECUTION OPTIONS:**

- TestSuite:** NewFeatures
- Configuration\_Type:** Product\_Config (dropdown)
- FilterWithTags:** NO (dropdown)
- Include\_Regression:** NO (dropdown)
- Fetch\_Log\_Upon\_Failure:** YES (dropdown)

**FEATURES AND TESTCASES:**

- Features:** All (dropdown)
- Stage:** stage1
- Group:** group1
  - BSF\_AAR\_fakeAVP
  - internal\_BSF\_API\_Validations

At the bottom, there are two buttons: a green 'Build' button with a play icon and a grey 'Cancel' button.

### Note

Make sure that the value of **FilterWithTags** and **Include\_NewFeatures** is selected as **NO**.

2. If you want to collect logs for any given build, select YES from the drop-down menu of **Fetch\_Log\_Upon\_Failure**.
3. Click **Build** and select **Console Output** to view the test results. The following is a sample test result output:

**Figure 4-7 Sample: Test Result Output in Console**

 **Console Output**

```
Started by user bsfuser
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/.jenkins/workspace/BSF_NewFeatures
[Pipeline] {
[Pipeline] withEnv
[Pipeline] {
[Pipeline] sh
+ sh /var/lib/jenkins/ocbsf_tests/preTestConfig-NewFeatures-BSF.sh -a BSF -b vega-infodev1 -c ocne-prometheus-server -d vega-infodev1 -e unsecure -f fe7d992b-0541-4c7d-ab84-c6d70b1b0666 -g 45 -h 60 -i 140 -j 2 -k 0 -l 1 -m 1 -n 30 -o 1 -p ocne-prometheus-server.ocne-infra -q 80 -r 30 -s 5 -t 30 -u 5 -v 0 -w ocne-elastic-elasticsearch-master.ocne-infra -x 9200 -y yes -z kubernetes -A 15 -B 60
export NF=BSF
export BSF_Namespace=vega-infodev1
export gostub_namespace=vega-infodev1
export SECURITY=unsecure
export bsf_nfInstanceId=fe7d992b-0541-4c7d-ab84-c6d70b1b0666
export pod_restart_wait=45
export bsf_add_wait=60
export nrf_notif_wait=140
export RERUN_COUNT=2
export test_suite_to_initialize=0
export stub_response_to_be_set=1
export bsf_configuration_addition=1
export peer_connection_establishment=30
export next_message=1
export PROHSVCIP=ocne-prometheus-server.ocne-infra
export PROHSVCPORT=80
export TIME_INT_POD_DOWN=30
export POD_DOWN_RETRY=5
```

 **Note**

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

## Queuing Jenkins Jobs

Using this feature, you can queue a second job even when current job is still running. The second job can be triggered either from the same or a different pipeline.

The following table lists various scenarios on how queuing works:

**Table 4-1 Queuing Jenkins Jobs**

| Concurrent Builds | New Features Current Build | New Features New Build | Regression Current Build | Regression New Build Build | Result                                       |
|-------------------|----------------------------|------------------------|--------------------------|----------------------------|----------------------------------------------|
| Enabled           | Running                    | Triggered              | NA                       | NA                         | New-Build of New-Features is added to queue. |
| Enabled           | Running                    | NA                     | NA                       | Triggered                  | New-Build of Regression is added to queue.   |

Table 4-1 (Cont.) Queuing Jenkins Jobs

| Concurrent Builds | New Features Current Build | New Features New Build | Regression Current Build | Regression New Build Build | Result                                       |
|-------------------|----------------------------|------------------------|--------------------------|----------------------------|----------------------------------------------|
| Disabled          | NA                         | NA                     | Running                  | Triggered                  | New-Build of Regression is added to queue.   |
| Disabled          | NA                         | Triggered              | Running                  | NA                         | New-Build of New-Features is added to queue. |

## 4.1.4 Running BSF-HealthCheck Pipeline

This is a pre-configured pipeline where ATS performs a test probe with SUT. It triggers helm test & provides the results in Jenkins Console Logs.

You can run Policy-HealthCheck pipeline to check if all BSF pods are up and running. If yes, it provides the status as successful. If any pod is down due to any reason, then the pipeline fails.

To configure the parameters for **BSF-HealthCheck** pipeline, perform the following steps:

1. Click **BSF-HealthCheck** in the **Name** column.
2. Click **Configure** in the left navigation pane.
3. When you scroll-down, the **General** tab becomes active. Be sure that the screen loads completely.
4. Continue scrolling down until the **Pipeline** tab becomes active. The following is a screen capture that shows the Pipeline script:

Figure 4-8 Helm Test Script

```

1 - node{
2 def myVar = 'initial_value'
3 def buildVar = 'initial_value'
4 properties{
5 [
6 parameters(
7 [string(defaultValue: '', name: 'Helm_releases',description: "Provide Release Name with Comma Separated if more than 1"),
8 string(defaultValue: '', name: 'Namespace', description: "Provide the namespace")]
9)
10]
11 }
12 stage('Helm-Test-Init') {
13 catchError(buildResult: 'SUCCESS', stageResult: 'UNSTABLE') {
14 //a = helm releases [Provide Release Name with Comma Separated if more than 1]
15 //b= Namespace, If not applicable then remove the argument
16 }
17 }
18 }

```

5. In the **Script** area under the **Pipeline** section, users may customize the values for the following parameters:

### NOT\_SUPPORTED

Do not modify values of parameters other than the ones described in this section.

- **a:** Change this parameter to update the helm release name where Policy is deployed in your bastion.

- **b:** Change this parameter to update the namespace where Policy is deployed in your bastion.

6. Click **Save** to update the values.

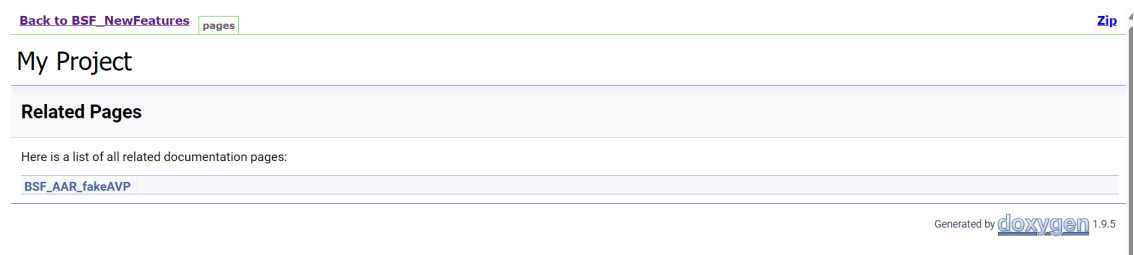
### Running Helm Test

To run BSF test cases, click **Build Now**.

## 4.1.5 BSF New Features Documentation

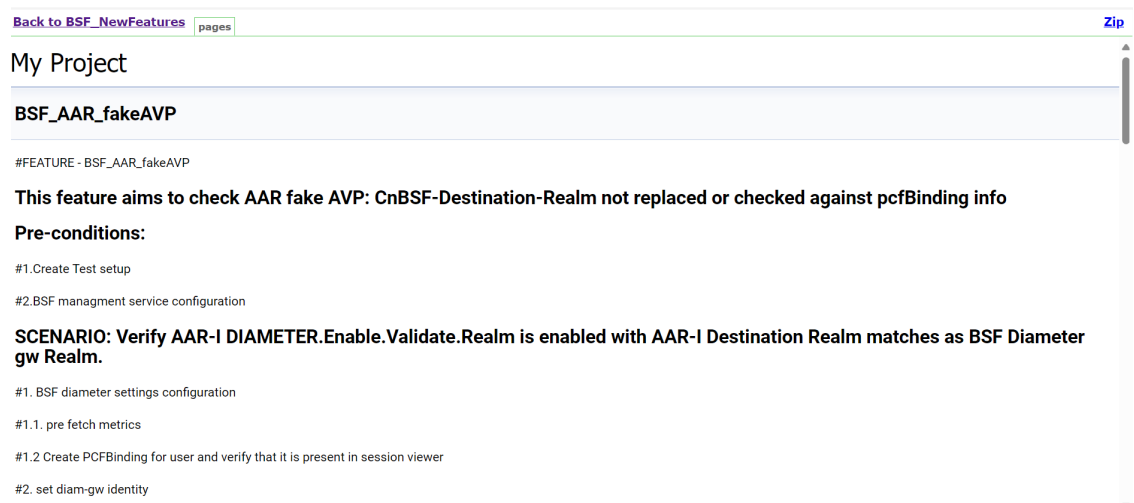
To view BSF functionalities, go to BSF-NewFeatures pipeline and click **Documentation** link in the left navigation pane.

**Figure 4-9 BSF-NewFeatures Feature List**



Click any functionality to view its test cases and scenarios of each test case. For example, when you click **FEATURE - BSF\_DIAMETER\_Message\_Retry\_AAR**, the following test description appears:

**Figure 4-10 Test Cases and Scenarios of Feature-BSF Diameter Message Retry**



Based on the functionalities covered under Documentation, the **Build Requires Parameters** screen displays test cases. To navigate back to the Pipeline BSF-NewFeatures screen, click **Back to BSF-NewFeatures** link available on top left corner of the screen.

## Test Result Analyzer

Using the Test Result Analyzer plug-in available in ATS, user can view consolidated and detailed reports. For more information, see [Test Results Analyzer](#) section.

## Test Case Mapping to Features and Display Total Counts

With this feature, users can view Total Count of Features, TestCases/Scenarios and TestCase mapping to each Feature of BSF in ATS View. For more information, see [Support for Test Case Mapping and Count](#) section.

## 4.1.6 Running BSF Regression Pipeline

This section describes how to run test cases for Binding Support Function (BSF) Regression pipeline.

The **BSF\_Regression** pipeline is a pre-configured pipeline where all the test cases of previous releases are available. For example, for BSF 24.1.0, this pipeline has all the test cases released till BSF 23.4.0.

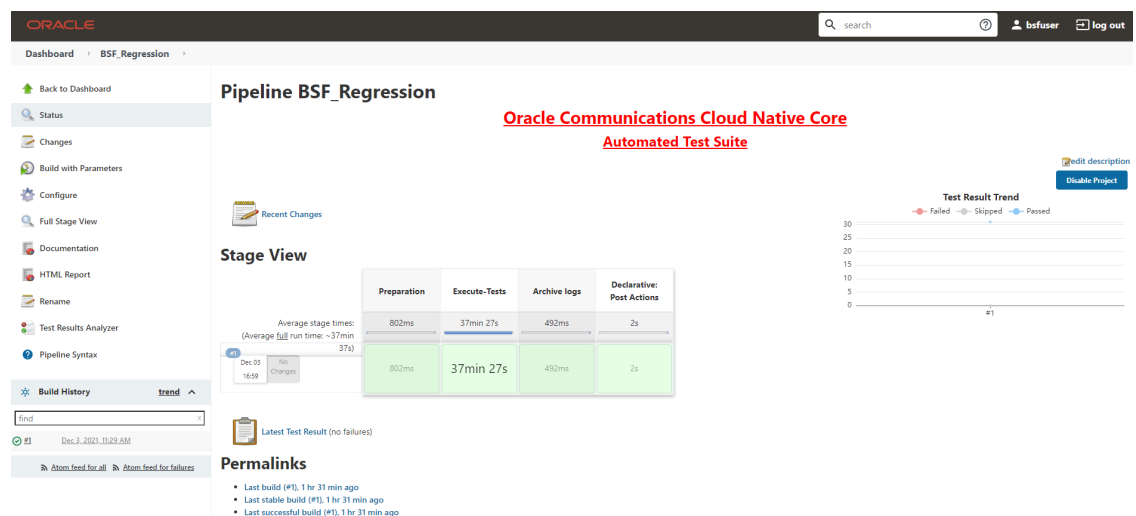
To view **BSF\_Regression** pipeline details, perform the following steps:

1. Click **BSF\_Regression** in the **Name** column.
2. Click **Build with Parameters** in the left navigation pane.

Copy the required test cases that are available in the BSF folder and place them appropriately within the **custom** folder for BSF\_Regression. Reload the page to view the test cases available in the **custom** Regression folder.

Click **Build**

**Figure 4-11 Regression Pipeline**



**Note**

The Regression pipeline does not have any sanity option. However, users must perform all the steps performed in the **BSF\_NewFeatures** pipeline. Ensure that the pipeline script is configured according to the environment variables.

## 4.1.7 BSF Regression Documentation

This section describes the documentation for **BSF-Regression** pipeline.

To view the documentation for any of the BSF features, on the ATS home page, click BSF-Regression. Then, click **Documentation** in the left navigation pane.

This page shows features of only those test cases that are released in previous releases.

## 4.2 Running NEF Test Cases using ATS

This section describes how to run Network Exposure Function (NEF) test cases using ATS. It includes:

- [Prerequisites](#)
- [Logging into ATS](#)
- [NEF-NewFeatures Pipeline](#)
- [NEF-NewFeatures Documentation](#)

### 4.2.1 Prerequisites

To run NEF test cases, ensure that the following prerequisites are met:

- The ATS version must be compatible with the NEF release.
- The NEF, ATS, Stub, and CNC Console must be deployed in the same namespace.
- The NEF must be deployed using the appropriate custom values.yaml file as per the configuration to be tested. The custom values.yaml file is available with the NEF documentation package. Ensure that you deploy the NEF for ATS using the correct custom value file based on the selected NEF configuration. For more information about the ATS configurations for different NEF feature test cases, see [NEF-NewFeatures Pipeline](#) and [NEF-Regression Pipeline](#).

**Note**

The following feature is added in **NEF-NewFeaturesPipeline** for the release 24.1.0:

- NEF Integration with the CNC Console
- Support for MSISDN-Less MO-SMS

NEF provides a set of eleven custom value files that can be used to customize the NEF deployment for ATS. The following table describes the mapping between the combination of the selected communication model and feature configurations for running NEF testcases and the custom value file to be used for NEF deployment for ATS:

## New Feature Pipeline

Table 4-2 Custom Value File for NewFeaturesPipeline for NEF deployment

| NEF Execute Suite | NEF Suite Configuration | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                          |
|-------------------|-------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nef Model-A       | Msisdnless-MO-SMS       | <p>oc-nef-custom-values-ats-udm-udr-modelA.yaml<br/>In the above file update the parameters as below:</p> <pre> msisdnless_mo_sms: true uncomment # -"ocnef-msisdnlessmosms" under appinfo: section  convergedScefNef: &amp;diamGatewayEnable true  <b>Simulator update:</b> In ocstub-nef-custom-values.yaml file, update the parameter as below:  ocstub-diam-nef: enabled: true </pre> |
| Nef Model-B       | Msisdnless-MO-SMS       | <p>oc-nef-custom-values-ats-udm-udr-modelB.yaml<br/>In the above file update the parameters as below:</p> <pre> msisdnless_mo_sms: true uncomment # -"ocnef-msisdnlessmosms" under appinfo: section  convergedScefNef: &amp;diamGatewayEnable true  <b>Simulator update:</b> In ocstub-nef-custom-values.yaml file, update the parameter as below:  ocstub-diam-nef: enabled: true </pre> |

Table 4-2 (Cont.) Custom Value File for NewFeaturesPipeline for NEF deployment

| NEF Execute Suite | NEF Suite Configuration | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------|-------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nef Model-D       | Msisdnless-MO-SMS       | <p>oc-nef-custom-values-ats-udm-udr-modelD.yaml<br/>In the above file update the parameters as below:</p> <pre>msisdnless_mo_sms: true uncomment # -"ocnef-msisdnlessmosms" under appinfo: section</pre> <p>convergedScefNef:<br/>&amp;diamGatewayEnable true</p> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <pre>ocstub-diam-nef: enabled: true</pre> |
| CNCC              | CNCC-NEF                | oc-nef-24.1.0-custom-values-ats-cncc.yaml                                                                                                                                                                                                                                                                                                                                                                             |

## Regression Pipeline

Table 4-3 Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration                 | Custom Values File to be used for NEF Deployment         |
|-------------------------------|-----------------------------------------|----------------------------------------------------------|
| CAPIF                         | CAPIF-Event Manager                     | oc-capif-custom-values-ats.yaml                          |
|                               | CAPIF-Functionality                     | oc-capif-custom-values-ats.yaml                          |
| Nef Model-A                   | ME-Location Reporting-GMLC              | oc-nef-custom-values-ats-gmlc-bsf-modelA.yaml            |
|                               | ME-Location Reporting-UDM               | oc-nef-custom-values-ats-udm-udr-modelA.yaml             |
|                               | ME-Location Reporting-GMLC-Failover-UDM | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelA.yaml |
|                               | AsQos+CHF_PCF                           | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelA.yaml |
|                               | AsQos+CHF_BSF                           | oc-nef-custom-values-ats-gmlc-bsf-modelA.yaml            |
|                               | TI_PCF                                  | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelA.yaml |
|                               | TI_BSF                                  | oc-nef-custom-values-ats-gmlc-bsf-modelA.yaml            |
|                               | TI_UDR                                  | oc-nef-custom-values-ats-udm-udr-modelA.yaml             |



Table 4-3 (Cont.) Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration   | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | ME - SCEF+NEF             | <p>oc-nef-custom-values-ats-udm-udr-modelA.yaml<br/>In oc-nef-custom-values-ats-udm-udr-modelA.yaml file update the parameter as below:</p> <p>convergedScefNef:<br/>&amp;diamGatewayEnable true</p> <p>convergedME: &amp;convergedME true</p> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <p>ocstub-diam-nef:<br/>enabled: true</p>                                        |
|                               | ME - PDU Session          | oc-nef-custom-values-ats-udm-udr-modelA.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | ME - Loss of Connectivity | oc-nef-custom-values-ats-udm-udr-modelA.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | ME - UeReachability       | oc-nef-custom-values-ats-udm-udr-modelA.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | Qos+CHF - SCEF+NEF        | <p>oc-nef-custom-values-ats-udm-udr-modelA.yaml<br/>In oc-nef-custom-values-ats-udm-udr-modelA.yaml file update the parameter as below:</p> <p>convergedScefNef:<br/>&amp;diamGatewayEnable true</p> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <p>ocstub-diam-nef:<br/>enabled: true</p> <p>pcf:<br/>    throwexception: true<br/>    switchToPCRFOnAuthFailure: true</p> |

Table 4-3 (Cont.) Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration                                                                                                                                                                                                       | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | Device Trigger                                                                                                                                                                                                                | <p>oc-nef-custom-values-ats-udm-udr-modelA.yaml<br/>In oc-nef-custom-values-ats-udm-udr-modelA.yaml file update the parameter as below:</p> <pre>convergedScefNef: &amp;diamGatewayEnable true deviceTrigger: true</pre> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <pre>ocstub-diam-nef: enabled: true</pre> <pre>pcf:   throwexception: true   switchToPCRFOnAuthFailure: true</pre> |
| Nef Model-B                   | ME-Location Reporting-GMLC                                                                                                                                                                                                    | oc-nef-custom-values-ats-gmlc-bsf-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | ME-Location Reporting-UDM                                                                                                                                                                                                     | oc-nef-custom-values-ats-udm-udr-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                                          |
|                               | ME-Location Reporting-GMLC-Failover-UDM<br><b>Note:</b> When this configuration is selected, the simulators must be re-installed by changing the gmlc fqdn to some random values under the nrf configurations as shown below: | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | <pre>nrf:   gmlcHost: "http:// ocnefsim-ocstub-svc-gmlc- wrong:1010"</pre> <p>Then, revert back to the value made under the nrf stub.</p>                                                                                     |                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|                               | AsQos+CHF_PCF                                                                                                                                                                                                                 | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | AsQos+CHF_BSF                                                                                                                                                                                                                 | oc-nef-custom-values-ats-gmlc-bsf-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | TI_PCF                                                                                                                                                                                                                        | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | TI_BSF                                                                                                                                                                                                                        | oc-nef-custom-values-ats-gmlc-bsf-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | TI_UDR                                                                                                                                                                                                                        | oc-nef-custom-values-ats-udm-udr-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                                                          |

Table 4-3 (Cont.) Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration   | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                 |
|-------------------------------|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | ME - SCEF+NEF             | oc-nef-custom-values-ats-udm-udr-modelB.yaml<br>In oc-nef-custom-values-ats-udm-udr-modelB.yaml file update the parameter as below:<br><br>convergedScefNef:<br>&diamGatewayEnable true<br><br><b>Simulator update:</b><br>In ocstub-nef-custom-values.yaml file, update the parameter as below:<br><br>ocstub-diam-nef:<br>enabled: true                                                                        |
|                               | ME - PDU Session          | oc-nef-custom-values-ats-udm-udr-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                     |
|                               | ME - Loss of Connectivity | oc-nef-custom-values-ats-udm-udr-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                     |
|                               | ME - UeReachability       | oc-nef-custom-values-ats-udm-udr-modelB.yaml                                                                                                                                                                                                                                                                                                                                                                     |
|                               | Qos+CHF - SCEF+NEF        | oc-nef-custom-values-ats-udm-udr-modelB.yaml<br>In oc-nef-custom-values-ats-udm-udr-modelB.yaml file update the parameter as below:<br><br>convergedScefNef:<br>&diamGatewayEnable true<br><br><b>Simulator update:</b><br>In ocstub-nef-custom-values.yaml file, update the parameter as below:<br><br>ocstub-diam-nef:<br>enabled: true<br><br>pcf:<br>throwexception: true<br>switchToPCRFOnAuthFailure: true |

Table 4-3 (Cont.) Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration                 | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------|-----------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | Device Trigger                          | <p>oc-nef-custom-values-ats-udm-udr-modelB.yaml<br/>In oc-nef-custom-values-ats-udm-udr-modelB.yaml file update the parameter as below:</p> <pre>convergedScefNef: &amp;diamGatewayEnable true deviceTrigger: true</pre> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <pre>ocstub-diam-nef: enabled: true</pre> <pre>pcf:   throwexception: true   switchToPCRFOnAuthFailure: true</pre> |
| Nef Model-D                   | ME-Location Reporting-GMLC              | oc-nef-custom-values-ats-gmlc-bsf-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | ME-Location Reporting-UDM               | oc-nef-custom-values-ats-udm-udr-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                          |
|                               | ME-Location Reporting-GMLC-Failover-UDM | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | AsQos+CHF_PCF                           | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | AsQos+CHF_BSF                           | oc-nef-custom-values-ats-gmlc-bsf-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | TI_PCF                                  | oc-nef-custom-values-ats-pcf-gmlcfailoverudm-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                              |
|                               | TI_BSF                                  | oc-nef-custom-values-ats-gmlc-bsf-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                         |
|                               | TI_UDR                                  | oc-nef-custom-values-ats-udm-udr-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                          |
|                               | ME - SCEF+NEF                           | <p>oc-nef-custom-values-ats-udm-udr-modelD.yaml<br/>In oc-nef-custom-values-ats-udm-udr-modelD.yaml file update the parameter as below:</p> <pre>convergedScefNef: &amp;diamGatewayEnable true</pre> <p><b>Simulator update:</b><br/>In ocstub-nef-custom-values.yaml file, update the parameter as below:</p> <pre>ocstub-diam-nef: enabled: true</pre>                                                                                              |
|                               | ME - PDU Session                        | oc-nef-custom-values-ats-udm-udr-modelD.yaml                                                                                                                                                                                                                                                                                                                                                                                                          |

Table 4-3 (Cont.) Custom Value File for Regression Pipeline for NEF deployment

| Supported Communication Model | NEF Suite Configuration   | Custom Values File to be used for NEF Deployment                                                                                                                                                                                                                                                                                                                                                                                          |
|-------------------------------|---------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                               | ME - Loss of Connectivity | oc-nef-custom-values-ats-udm-udr-modelID.yaml                                                                                                                                                                                                                                                                                                                                                                                             |
|                               | ME - UeReachability       | oc-nef-custom-values-ats-udm-udr-modelID.yaml                                                                                                                                                                                                                                                                                                                                                                                             |
|                               | Qos+CHF - SCEF+NEF        | oc-nef-custom-values-ats-udm-udr-modelID.yaml<br>In oc-nef-custom-values-ats-udm-udr-modelID.yaml file update the parameter as below:<br><br>convergedScefNef:<br>&diamGatewayEnable true<br><br><b>Simulator update:</b><br>In ocstub-nef-custom-values.yaml file, update the parameter as below:<br><br>ocstub-diam-nef:<br>enabled: true<br><br>pcf:<br>throwexception: true<br>switchToPCRFOnAuthFailure: true                        |
|                               | Device Trigger            | oc-nef-custom-values-ats-udm-udr-modelID.yaml<br>In oc-nef-custom-values-ats-udm-udr-modelID.yaml file update the parameter as below:<br><br>convergedScefNef:<br>&diamGatewayEnable true<br>deviceTrigger: true<br><br><b>Simulator update:</b><br>In ocstub-nef-custom-values.yaml file, update the parameter as below:<br><br>ocstub-diam-nef:<br>enabled: true<br><br>pcf:<br>throwexception: true<br>switchToPCRFOnAuthFailure: true |

For more information about the NEF installation, see *Oracle Communications Cloud Native Core, Network Exposure Function Installation, Upgrade, and Fault Recovery Guide*.

- ATS Prometheus metrics validation works only when the installation has a single pod for each microservice in the NEF deployment.

- You can customize test cases in the custom test case folders (cust\_newfeatures, cust\_regression and cust\_performance). You can add new test cases, remove unwanted test cases, and modify existing test cases. It does not impact the original product packaged test cases available in the newfeatures, regression, and performance folders respectively. For more details about custom test case folders, see [Custom Folder Implementation](#).
- Install Prometheus server in the cluster.
- Database cluster is in the running state with all required tables. Verify that there are no previous entries in the database before running test cases.
- Do not initiate a job in two different pipelines at the same time.

## 4.2.2 Logging into ATS

Before logging into ATS, you need to ensure that ATS is deployed successfully using HELM charts as shown below:

### Note

To modify default log in password, refer to [Modifying Login Password](#).

Figure 4-12 Verifying ATS Pod

|                                                 |     |         |             |       |
|-------------------------------------------------|-----|---------|-------------|-------|
| mysql-ocnef-6c6cf46765-xwtbq                    | 1/1 | Running | 1 (18d ago) | 40d   |
| nefats-ocats-78f58df796-nv68f                   | 1/1 | Running | 0           | 11d   |
| occapif-afmgr-cc5944c84-bt5d5                   | 1/1 | Running | 0           | 7d3h  |
| occapif-apimg-7df5d59887-c866p                  | 1/1 | Running | 0           | 7d8h  |
| occapif-eventmgr-6c5cc74949-728k6               | 1/1 | Running | 0           | 7d8h  |
| occapif-ext-egress-gateway-757fcb8f67-8grmc     | 1/1 | Running | 0           | 7d8h  |
| occapif-ext-ingress-gateway-6c5fbc8bdd-2g79v    | 1/1 | Running | 0           | 7d8h  |
| occapif-network-egressgateway-8688974b8d-qgwg6  | 1/1 | Running | 0           | 7d8h  |
| occapif-network-ingressgateway-5fbfdb5755-s76ck | 1/1 | Running | 0           | 7d8h  |
| occncc-acore-ingress-gateway-8c47c6685-kqrzn    | 1/1 | Running | 0           | 7d1h  |
| occncc-iam-ingress-gateway-5598c65fb6-g5tpl     | 1/1 | Running | 0           | 7d21h |
| occncc-iam-kc-0                                 | 2/2 | Running | 0           | 7d21h |
| occncc-mcore-cmservice-84dd4754c5-985gg         | 1/1 | Running | 0           | 7d1h  |
| occncc-mcore-ingress-gateway-59c7cb7478-whp89   | 1/1 | Running | 0           | 7d1h  |
| ocnef-aef-apirouter-579bc98bb4-v7wqv            | 1/1 | Running | 0           | 7d7h  |
| ocnef-apdmanager-56cbb758b4-stkhb               | 1/1 | Running | 0           | 6d17h |
| ocnef-appinfo-5f96955868-bh585                  | 1/1 | Running | 0           | 7d7h  |
| ocnef-config-server-64648d5d69-qppjt            | 1/1 | Running | 0           | 7d7h  |
| ocnef-consoledataservice-856b897875-55xwj       | 1/1 | Running | 0           | 19h   |
| ocnef-devicetrigger-5699f8994b-shh5v            | 1/1 | Running | 0           | 6d17h |
| ocnef-ext-nef-egress-gateway-547bccdc9-4rhs5    | 1/1 | Running | 0           | 7d7h  |
| ocnef-ext-nef-ingress-gateway-6cff9fd75d-4vvgm  | 1/1 | Running | 0           | 7d7h  |
| ocnef-fivegc-egress-gateway-79df4dff7-2z97h     | 1/1 | Running | 0           | 7d7h  |
| ocnef-fivegc-ingress-gateway-5f4ff45fb4-qt7fl   | 1/1 | Running | 0           | 7d7h  |
| ocnef-fivegcagent-686b49d77-xpxsr               | 1/1 | Running | 0           | 19h   |
| ocnef-monitoringevents-c47c769bf-mqr5f          | 1/1 | Running | 0           | 19h   |
| ocnef-msisdnlessmosms-58cd864998-lvdw4          | 1/1 | Running | 0           | 19h   |

Figure 4-13 Verifying ATS Pod for Stubs

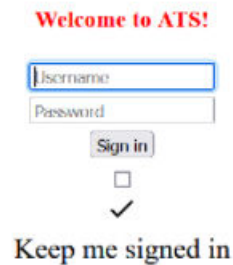
|                                           |     |         |   |     |
|-------------------------------------------|-----|---------|---|-----|
| ocnefsim-ocstub-diam-nef-7d8b7fd6c7-x8qql | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-af-5764f446fc-v45g7   | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-bsf-69d8c95d7f-jfhd5  | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-chf-5668c59456-kclqr  | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-gmlc-65865c5d6d-wfnx8 | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-nrf-78cfc96dd4-4tmx5  | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-pcf-68f7f89df6-djrf2  | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-udm-787dccfb78-2tth8  | 1/1 | Running | 0 | 61m |
| ocnefsim-ocstub-nef-udr-78589cc4b5-7crns  | 1/1 | Running | 0 | 61m |

For more information on verifying ATS deployment, see [Verifying ATS Deployment](#).

To log in to ATS, open a browser and provide the IP Address and port details as <Worker-Node-IP>:<Node-Port-of-ATS>.

**Figure 4-14 ATS Login**

### Oracle Communications Cloud Native Core - Automated Test Suite



Welcome to ATS!

Username

Password

Sign in

☐

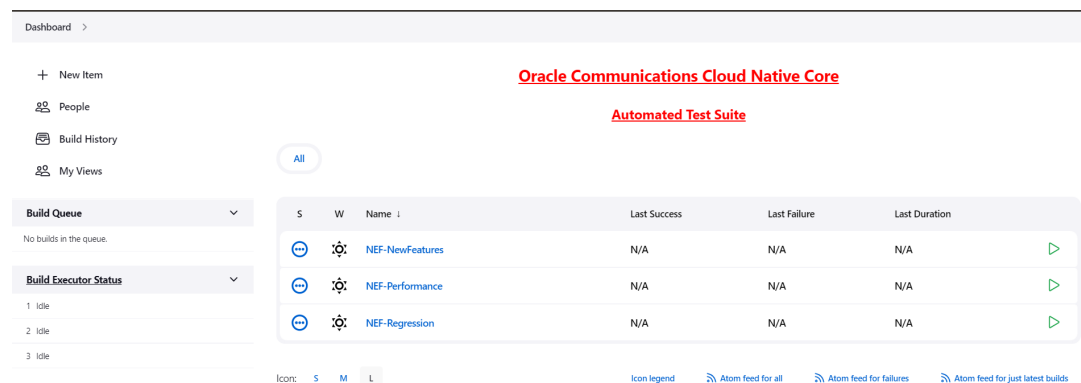
Keep me signed in

### Running ATS

To run ATS:

1. Enter the login credentials. Click **Sign in**. The following screen appears.

**Figure 4-15 NEF Pre-Configured Pipelines**



Dashboard >

+ New Item

People

Build History

My Views

Oracle Communications Cloud Native Core

Automated Test Suite

All

| Build Queue             | S   | W   | Name            | Last Success | Last Failure | Last Duration |
|-------------------------|-----|-----|-----------------|--------------|--------------|---------------|
| No builds in the queue. | --- | --- | NEF-NewFeatures | N/A          | N/A          | N/A           |
| Build Executor Status   | --- | --- | NEF-Performance | N/A          | N/A          | N/A           |
| 1 Idle                  | --- | --- | NEF-Regression  | N/A          | N/A          | N/A           |
| 2 Idle                  | --- | --- |                 |              |              |               |
| 3 Idle                  | --- | --- |                 |              |              |               |

Icon: S M L

Icon legend

Atom feed for all

Atom feed for failures

Atom feed for just latest builds

NEF ATS has three pre-configured pipelines.

- **NEF-NewFeatures:** This pipeline has all the test cases delivered as part of NEF ATS - 24.1.0.

#### Note

The following feature is added in **NEF-NewFeatures** for the release 24.1.0:

- NEF Integration with the CNC Console
- Support for MSISDN-Less MO-SMS

- **NEF-Performance:** This pipeline is not operational as of now. It is reserved for future releases of ATS.
- **NEF-Regression:** This pipeline has all the test cases delivered so far in the previous releases, along with Monitoring Event support for update operation added under UDM functionality.

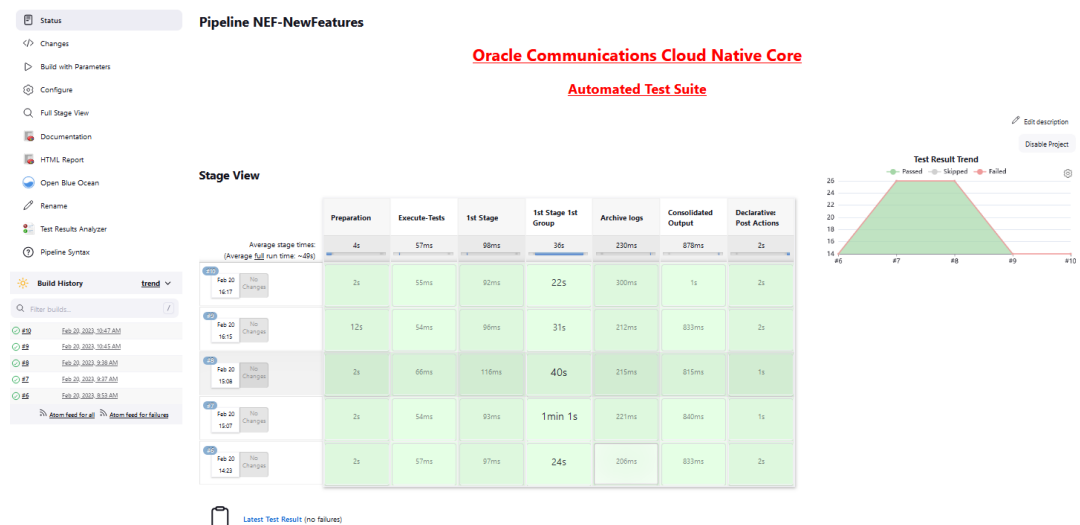
## 4.2.3 NEF-NewFeatures Pipeline

### NEF-NewFeatures Pipeline

This is a pre-configured pipeline where users can run all the NEF new test cases. To configure its parameters, which is a one time activity, perform the following steps:

1. Click **NEF-NewFeatures** in the **Name** column. The following screen appears:

Figure 4-16 Configuring NEF-New Features



In the above screen:

- Click **Configure** to configure NEF-New Features.
  - Click **Documentation** to view the documented test cases, which are part of this NEF release.
  - Click the blue dots inside **Build History** box to view the success console logs of the "Sanity", "All-NewFeatures" respectively.
  - The **Stage View** represents the already deployed pipeline for the customer reference.
2. Click **Configure**. User must wait for the page to load completely. Once the page loads completely, click the **Pipeline** tab:

#### Note

Make sure that the following page loads completely before you perform any action on it. Also, do not modify any configuration other than shown below.



Figure 4-17 Pipeline Tab

Configure

General

Advanced Project Options

Pipeline

General

Description

```

<html>
<head>
<h2 style="font-size:1.5vw;color:red;text-align:center;"><u>Oracle Communications Cloud Native Core</u></h2>
<h3 style="font-size:1.25vw;color:red;text-align: center;"><u>Automated Test Suite</u></h3>
</head>
<body>
</body>
</html>

```

[\[Safe HTML\] Preview](#)

☒ Enable project-based security

Inheritance Strategy

Do not inherit permission grants from other ACLs

This object will **not** inherit the [global security settings](#), or any permissions from its ancestors. Only permissions explicitly enabled here will be granted. To inadvertently locked out from Jenkins, an exception is made for the **Overall/Administer** permission: Administrators of Jenkins will still have access to this explicitly granted here.

The **Pipeline** section of the configuration page appears as follows:

The content of the pipeline script is as follows:

Figure 4-18 Pipeline Script

```

1 node ("built-in"){
2 //a = SELECTED_UIF b = NEF_namespace c = ingress_gateway_svc d = ingress_gateway_port
3 //e = prometheus_ip f = prometheus_port g = nef_name h = RERUI_COUNT
4 //i = AF_SDI_UIF j = AF_SDI_PORT k = AF_SDI_SVC
5 //l = UOR_SDI_UIF n = UOR_SDI_PORT m = UOR_SDI_SVC o = fivegc_ingress_gateway_svc p = fivegc_ingress_gateway_port
6 //q = gnlc_sdi_uiif r = gnlc_port s = gnlc_sdi_svc t = capif_ingress_svc u = capif_ingress_port v = EventNGR_capif_svc
7 //w = qos_svc x = pcf_sin_name y = pcf_port z = pcf_svc A = NE_SVC B = fivegcagent_svc C = API_NGR_SVC D = AF_NGR_SVC
8 //E = T2_SVC F = Diam_Gw_name G = Diam_Gw_port H = Diam_Gw_sin_svc I = Diam_Gw_svc J = DT_svc K = CHF_Name L = CHF_Port M = CHF_svc
9 //N = HSIIDLESS_NO_SVS_SVC O = Cncc_namespace P = ENABLE_CHKC_UT_INT Q = CHK_IN_USERNAME R = CHK_Console_USERNAME
10 //S = IP_Dropdown T = CHK_Release_Name U = CHK_Roles V = sleep_time W = max_entry X = wait_time Y = cncc_json_data_path
11
12
13 withEnv([
14
15]){
16
17 sh '''
18 sh /var/lib/jenkins/ocnef_tests/preTestConfig.sh \
19 -a NEF \
20 -b sanity \
21 -c ocnef-ext-nef-ingress-gateway \
22 -d 80 \
23 -e prometheus.one-infra \
24 -f 9090 \
25 -g ocnef \
26 -h 0 \
27 -i sft \
28 -j 1010 \
29 -k ocnefsm-ocstub-svc-sf \
30 -l udn1 \
31 -m 1010 \
32 -n ocnefsm-ocstub-svc-udm \
33 -o ocnef-fivegc-ingress-gateway \
34 -p 80 \
35 -q gnlc1 \
36 -r 1010 \
37 -s ocnefsm-ocstub-svc-gnlc \
38 -t occapif-ext-ingress-gateway \
39 -u 80 \
40 -v occapif-eventmgr.sanity \
41 -w ocnef-qualityofservice.sanity \
42 -x pcf1 \
43 -y 1010 \
44 -z ocnefsm-ocstub-svc-pcf \
45 -A ocnef-monitoringevents.sanity \
46 -B ocnef-fivegcagent.sanity \
47 -C occapif-espgr.sanity \
48 -D occapif-afmgr.sanity \
49 -E ocnef-trafficinfluence.sanity \
50 -F diamg1 \
51 -G 1010 \
52 -H ocnefsm-ocstub-diam-svc \
53 -I ocnef-oc-diam-gateway.sanity \
54 -J ocnef-devicetrigger.sanity \
55 -K chf1 \
56 -L 1010 \
57 -M ocnefsm-ocstub-svc-chf.sanity \
58 -N ocnef-hsiidlessmons.sanity \
59 -O sanity \
60 -P yes \
61 -Q admin \
62 -R nefuser \
63 -S Cluster1.NEF.nef \
64 -T 24.1.0-beta.43 \
65 -U ADMIN \
66 -V 2 \
67 -W 5 \
68 -X 150 \
69 -Y /var/lib/jenkins/ocnef_tests/cust_data/cncc \
70 ...

```

**Note**

As required by the setup details, the user can change any value except **line number 18**.

You can change the parameter values from "**a**" - to - "**z**" and "**A**" - to - "**Y**" as per user requirement. The parameter details are available as comments from line number **2** - to - **10**.

- **a:** Name of the NF to be tested in capital (NEF).
- **b:** Namespace in which the NEF is deployed.
- **c:** endPointIP (NEF ingress gateway service )value used while deploying NEF with the help of helm chart.
- **d:** endPointPort (NEF ingress gateway port) value used while deploying NEF with the help of helm chart.
- **e:** Name\_of\_Prometheus\_service.namespace (ocnef-prometheus-server.ocnef-infra).
- **f:** Port of Prometheus service (80).
- **g:** Helm\_release\_name used to deploy NEF (ocnef).
- **h:** Number of times the re-run of failed case is allowed (default as 2).
- **i:** AF Simulator Name used to install simulator (Af1)
- **j:** AF Simulator Port(1010).
- **k:** AF Simulator service used to install.
- **l:** UDM Simulator Name used to install simulator (udm1)
- **m:** UDM Simulator Port(1010).
- **n:** UDM Simulator service used to install.
- **o:** fivegc endPointIP (NEF fivegc ingress gateway service )value used while deploying NEF with the help of helm chart.
- **p:** five gc endPointPort (NEF fivegc ingress gateway port) value used while deploying NEF with the help of helm chart.
- **q:** GMLC Simulator Name used to install simulator (udm1)
- **r:** GMLC Simulator Port(1010).
- **s:** GMLC Simulator service used to install.
- **t:** OCCAPIF External Gateway
- **u:** endPointPort (OCCAPIF ingress gateway port) value used while deploying occapif with the help of helm chart.
- **v:** OCCAPIF Event manager service
- **w:** QoS service
- **x:** PCF simulator
- **y:**PCF simulator port
- **z:** PCF Simulator service used to install.
- **A:** OCNEF ME Service
- **B:** OCNEF FiveGCAgent Service

- **C:** OCCAPIF API Manager service
- **D:** OCCAPIF AF Manager service
- **E:** OCNEF Traffic Influence service
- **F:** Diameter Gateway name
- **G:** Diameter Gateway port number
- **H:** Diameter Gateway Simulator service
- **I:** Diameter Gateway service
- **J:** Device Trigger service
- **K:** Charging Function Name
- **L:** Charging Function SIM Port
- **M:** Charging Function SIM service
- **N:** MSISDNLess MO SMS service
- **O:** CNC Console Namespace
- **P:** Enable CNC Console UI
- **Q:** CNC User name
- **R:** CNC Console User
- **S:** NF Dropdown
- **T:** CNC Release version
- **U:** CNC Roles
- **V:** Sleep Time
- **W:** Maximum Entry
- **X:** Wait Time
- **Y:** CNCC JSON Data Path

**Note**

If user is using OCCNE1.6/OSO-1.6.1 or earlier, change the Prometheus service URI (-n) to "/api/" and Prometheus service name and port to appropriate values. For OCCNE 1.7.0/OSO 1.6.2 or later, retain the default value **"/prometheus/api/"**

3. Click **Save** after making necessary changes. The NEF-NewFeatures screen appears. Click the **Build with Parameters** link available in the left navigation pane of the **NEF-NewFeatures** Pipeline screen. The following screen appears.

Figure 4-19 Pipeline NEF\_NewFeatures

4. Select an option from the **Execute\_Suite** section:

**Note**

Based on the option selected in this section, the values under **Suite\_Configuration** sections get populated.

5. Select the applicable configuration under the **Suite\_Configuration** section based on the selected value for **Execute\_Suite** and the custom configuration file used for NEF deployment for ATS. For more information about the NEF custom\_values.yaml files for ATS, see "Download NEF Package" in the *Oracle Communications Cloud Native Core, Network Exposure Function Installation, Upgrade, and Fault Recovery Guide*.

**Important**

Ensure that the ATS configurations match with the custom\_values.yaml file used for NEF deployment. A mismatch between the NEF for ATS deployment configurations and the selected Suite configurations results in build failure.

The following table describes the mapping between the options selected for **Execute\_Suite** and **Suite\_Configuration** and the respective feature test cases that can be run:

Table 4-4 Execution Suite Configuration Mapping

| Execute_Suite | Suite_Configuration | Supported NEF Custom yaml file               | Features                  |
|---------------|---------------------|----------------------------------------------|---------------------------|
| Nef Model-A   | Msisdnless-MO-SMS   | oc-nef-custom-values-ats-udm-udr-modelA.yaml | Ocnef_MO-Sms_Notification |
| Nef Model-B   | Msisdnless-MO-SMS   | oc-nef-custom-values-ats-udm-udr-modelB.yaml | Ocnef_MO-Sms_Notification |
| Nef Model-D   | Msisdnless-MO-SMS   | oc-nef-custom-values-ats-udm-udr-modelD.yaml | Ocnef_MO-Sms_Notification |

Table 4-4 (Cont.) Execution Suite Configuration Mapping

| Execute_Suite | Suite_Configuration | Supported NEF Custom yaml file     | Features                                                                                                                                                                                                                                                                                                                                                                                                                                      |
|---------------|---------------------|------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CNCC          | CNCC NEF            | oc-nef-custom-values-ats-cncc.yaml | OCNEF_Console_Home_Screen_Validation<br>OCNEF_Global_Af_Config<br>OCNEF_Global_Comm_Profile_Mapping<br>OCNEF_Global_Model_D_Profile_Config<br>OCNEF_Global_NC_Afserviceid_Config<br>OCNEF_Global_NC_Qosprofile_Config<br>OCNEF_Global_NC_SCS_SME<br>OCNEF_Global_System_Configuration<br>OCNEF_Gmlc_Config<br>OCNEF_Logging_Config<br>OCNEF_Menultems<br>OCNEF_Msisdnless_Shortcode_Configuration<br>OCNEF_Qos_Config<br>OCNEF_Timeout_Config |

6. Select any of following valid values from **Configuration\_Type**:
  - **Product\_Config**: On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the key-value pair and yaml files defined or present in the "Product Config" folder.
  - **Custom\_Config**: On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the key-value pair and yaml files defined or present in the "Custom Config" folder.
7. Select any of the following options under the **Select\_Option** section:
  - **All**: This is the default option. It runs all the NEF test cases. Scroll down and click **Build** to execute all the test cases.
  - **Single/MultipleFeatures**: This option allows you to select any number of test cases that you want to run from the list of total test cases available for execution.
8. After selecting the test cases, click **Build** and select **Console Output** to view the test results. The following is a sample test result output:

**Figure 4-20 Sample Test Cases Result - New Feature**

```

3 features passed, 0 failed, 0 skipped
14 scenarios passed, 0 failed, 0 skipped
386 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m17.286s
+ set +x
[Pipeline] fileExists
[Pipeline] echo
File /var/lib/jenkins/ocnef_tests/features/cust_newfeatures/TI_BSF/stage1/group1/rerun-failing.features does not exist!
[Pipeline] sh (Stage: stage1 Group: group1 Report)
+ set +x
stage1 group1
Final Result:-
Total-Features RUN 3, PASS 3, FAIL 0

The report displayed here is same as the one displayed at the end of group execution
[Pipeline] echo
The below are the consolidated output of all the features that are run
[Pipeline] sh (Stage: stage1 Group: group1 Result)
+ set +x
*****Stage: stage1 Group: group1 Result*****
stage1 group1
Final Result:-
Total-Features RUN 3, PASS 3, FAIL 0

[Pipeline] sh (Overall Result)
+ set +x
*****Overall Result*****

```

**Note**

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

**Extracting Application Logs**

If any scenarios fail after running the pipeline job, perform the following steps:

1. Log in to the ATS pod:

```
kubectl exec -it pod/ocats-nef-6f6dfc76b5-jbgzt -n ocnef bash
```

2. Go to Jenkins build directory:

```
cd /var/lib/jenkins/.jenkins/jobs/NEF_NewFeatures/builds/2
```

For example:

```
cd $JENKINS_HOME/jobs/$JOB_NAME/builds/$BUILD_NUMBER/
```

3. Extract the applogs.zip file:

```
unzip applogs.zip
```

4. After successfully unzipping the file, open the applog folder to view the pod logs for failed scenarios:

```
(env) [jenkins@ocats-nef-6f6dfc76b5-jbgzt applog]$ pwd
/var/lib/jenkins/.jenkins/jobs/NEF_NewFeatures/builds/2/applog
(env) [jenkins@ocats-nef-6f6dfc76b5-jbgzt applog]$ ls -ltrh
total 760K
-rw-r--r--. 1 jenkins jenkins 250K Nov 19 11:08 ME_Subscription_case.log
-rw-r--r--. 1 jenkins jenkins 249K Nov 19 11:08 1st_Rerun-
ME_Subscription_case.log
-rw-r--r--. 1 jenkins jenkins 255K Nov 19 11:09 2nd_Rerun-
ME_Subscription_case.log
```

### Running NEF Test Cases

The following feature is added for the release 24.1.0:

- NEF Integration with the CNC Console
- Support for MSISDN-Less MO-SMS

For running this release feature, see [NEF-NewFeatures Pipeline](#). For running the previous release features, see [NEF-Regression Pipeline](#).

## 4.2.4 NEF-NewFeatures Documentation

To view NEF test cases, go to **NEF-NewFeatures** pipeline and click the **Documentation** link in the left navigation pane. The following feature is added for the release 24.1.0:

- NEF Integration with the CNC Console
- Support for MSISDN-Less MO-SMS

### Test Result Analyzer

Using the Test Result Analyzer plug-in available in ATS, user can view consolidated and detailed reports. For more information, see [Test Results Analyzer](#) section.

### Test Case Mapping to Features and Display Total Counts

With this feature, users can view Total Count of Features, TestCases/Scenarios and TestCase mapping to each Feature of NEF in ATS View. For more information, see [Support for Test Case Mapping and Count](#) section.

## 4.2.5 NEF-Regression Pipeline

This section describes how to run test cases for Network Exposure Function (NEF) Regression pipeline.

The NEF-Regression pipeline is a pre-configured pipeline where all the test cases of previous releases are available. For example, for NEF 24.1.0, this pipeline has all the test cases released till NEF 24.1.x.

To view NEF-Regression pipeline details, perform the following steps:

1. Click **NEF-Regression** in the **Name** column.
2. Click the **Build with Parameters** link available in the left navigation pane of the **NEF-Regression** Pipeline screen.

- Click **Build** and select **Console Output** to view the test results.

### Note

The Regression pipeline does not have any sanity option. However, you must perform all the steps performed in the **NEF-Regression** pipeline. Ensure that the pipeline script is configured according to the environment variables.

## Running Regression Test Cases

To run NEF test cases, perform the following steps:

- Click the **Build with Parameters** link available in the left navigation pane of the **NEF-Regression** Pipeline screen. The following page appears.

Figure 4-21 Pipeline NEF\_Regression

The screenshot shows the 'Pipeline NEF\_Regression' configuration interface. The top section, 'EXECUTION OPTIONS', contains four dropdown menus: 'TestSuite' (set to 'Regression'), 'Execute\_Suite' (with a dropdown menu open showing 'CAPIF', 'Nef Model-A', 'Nef Model-B', and 'Nef Model-D'), 'Suite\_Configuration' (set to 'ME-Location Reporting-GMLC'), and 'Configuration\_Type' (set to 'Product\_Config'). The bottom section, 'FEATURES AND TESTCASES', has a 'Features' dropdown set to 'All', displaying a list of test cases under the group 'stage1 Group: group1'. The list includes various test cases related to GMLC, such as 'OCNEF\_ME\_GMLC\_Additional', 'OCNEF\_ME\_GMLC\_Deferred\_Motion\_MaxReports\_Expiry', and 'OCNEF\_ME\_GMLC\_Deferred\_report\_area'.

- Select an option from the **Execute\_Suite** section:

### Note

Based on the option selected in this section, the values under **Suite\_Configuration** sections get populated.

- Select the applicable configuration under the **Suite\_Configuration** section based on the selected value for **Execute\_Suite** and the custom configuration file used for NEF deployment for ATS. For more information about the NEF `custom_values.yaml` files for ATS, see "Download NEF Package" in the *Oracle Communications Cloud Native Core, Network Exposure Function Installation, Upgrade, and Fault Recovery Guide*.

### Important

Ensure that the ATS configurations match with the `custom_values.yaml` file used for NEF deployment. A mismatch between the NEF for ATS deployment configurations and the selected Suite configurations results in build failure.



The following table describes the mapping between the options selected for **Execute\_Suite** and **Suite\_Configuration** and the respective feature test cases that can be run:

**Table 4-5 Execution Suite Configuration Mapping**

| Execute_Suite | Suite_Configuration        | Features                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------|----------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CAPIF         | CAPIF-Event Manager        | OCNEF_CAPIF_Event_Mgr<br>OCNEF_CAPIF_Event_Mgr_Creation_Failure<br>OCNEF_CAPIF_Event_Mgr_Duration_Based<br>OCNEF_CAPIF_Event_Mgr_Metrics<br>OCNEF_CAPIF_Event_Mgr_Notification<br>OCNEF_CAPIF_Event_Mgr_Publish<br>OCNEF_CAPIF_Event_Mgr_Unsubscribe<br>OCNEF_CAPIF_Event_Mgr_With_Filters                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
|               | CAPIF-Functionality        | OCNEF_CAPIF_Discovery_Group<br>OCNEF_CAPIF_OAuth<br>OCNEF_CAPIF_On_Off_boarding_Invoker<br>OCNEF_CAPIF_Preprovisioning_Invoker<br>OCNEF_CAPIF_SecurityContext                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | ME-Location Reporting-GMLC | OCNEF_ME_GMLC_Additional<br>OCNEF_ME_GMLC_Deferred_Area_MaxReports_Expiry<br>OCNEF_ME_GMLC_Deferred_Motion_MaxReports_Expiry<br>OCNEF_ME_GMLC_Deferred_Periodic_MaxReports_Expiry<br>OCNEF_ME_GMLC_Deferred_Periodic_error_exceeds_range<br>OCNEF_ME_GMLC_Deferred_report<br>OCNEF_ME_GMLC_Deferred_report_area<br>OCNEF_ME_GMLC_Deferred_report_area_onlyExpiry<br>OCNEF_ME_GMLC_Deferred_report_explicitCancellation_set<br>OCNEF_ME_GMLC_Deferred_report_inside_area<br>OCNEF_ME_GMLC_Deferred_report_leaving_area<br>OCNEF_ME_GMLC_Deferred_report_leaving_area_more_reports<br>OCNEF_ME_GMLC_Deferred_report_motion<br>OCNEF_ME_GMLC_Deferred_report_motion_more_reports<br>OCNEF_ME_GMLC_Deferred_report_motion_onlyExpiry<br>OCNEF_ME_GMLC_Deferred_report_negative_reports<br>OCNEF_ME_GMLC_Deferred_report_periodic<br>OCNEF_ME_GMLC_Deferred_report_periodic_more_reports<br>OCNEF_ME_GMLC_Deferred_report_periodic_onlyExpiry<br>OCNEF_ME_GMLC_Deferred_report_unknown<br>OCNEF_ME_GMLC_Immediate_report_currOrLastLocation<br>OCNEF_ME_GMLC_Immediate_report_currentLocation<br>OCNEF_ME_GMLC_Immediate_report_initialLocation<br>OCNEF_ME_GMLC_Immediate_report_lastKnownLocation<br>OCNEF_ME_GMLC_Immediate_report_unknown_monitorType<br>OCNEF_ME_GMLC_LocQos_Absent_destIfLocQosAbsent_gmlc<br>OCNEF_ME_GMLC_Notification_EnteringIntoArea_sendLocationInfo_false<br>OCNEF_ME_GMLC_Notification_EnteringIntoArea_sendLocationInfo_false_expiry<br>OCNEF_ME_GMLC_Notification_Motion_sendLocationInfo_false<br>OCNEF_ME_GMLC_Notification_Motion_sendLocationInfo_false_expiry<br>OCNEF_ME_GMLC_Notification_Periodic_sendLocationInfo_false<br>OCNEF_ME_GMLC_Notification_Periodic_sendLocationInfo_false_explicitCancel_false<br>OCNEF_ME_GMLC_Notification_Periodic_sendLocationInfo_false_explicitCancel_true<br>OCNEF_ME_GMLC_Notification_sendLocationInfo_set |
| Nef Model-B   |                            |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |

**Table 4-5 (Cont.) Execution Suite Configuration Mapping**

| Execute_Suite | Suite_Configuration                            | Features                                                                                                                                                                                                                                                                                                                              |
|---------------|------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nef Model-D   |                                                | OCNEF_ME_GMLC_PUT<br>OCNEF_ME_GMLC_Reports_Errors<br>OCNEF_ME_MODELD_gmlc_enabled                                                                                                                                                                                                                                                     |
| Nef Model-A   | ME-Location<br>Reporting-UDM                   | OCNEF_EFC_ME_AF_Unavailable<br>OCNEF_EFC_ME_Errors_Scenarios<br>OCNEF_EFC_ME_MultipleNotifications<br>OCNEF_EFC_ME_OneTimeNotification<br>OCNEF_GET_ME<br>OCNEF_GET_ME_Scenarios<br>OCNEF_ME_LocationReport_PUT<br>OCNEF_ME_Sub_UnSub                                                                                                 |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | ME-Location<br>Reporting-GMLC-<br>Failover-UDM | OCNEF_ME_GMLC_Not_Available_switchToUdmOnFailure_set<br>OCNEF_ME_MODELD_gmlc_enabled                                                                                                                                                                                                                                                  |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | AsQos+CHF_PCF                                  | OCNEF_Additional_Qos<br>OCNEF_CHF_QOS_Create<br>OCNEF_CHF_QOS_Delete<br>OCNEF_CHF_QOS_Metrics<br>OCNEF_CHF_QOS_Notification<br>OCNEF_CHF_QOS_Update<br>OCNEF_Qos_Create_Subscription<br>OCNEF_Qos_Delete_Subscription<br>OCNEF_Qos_Get_Subscription<br>OCNEF_Qos_Metrics<br>OCNEF_Qos_Notification<br>OCNEF_Qos_Update_with_PUT_PATCH |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | AsQos+CHF_BSF                                  | OCNEF_Additional_Qos<br>OCNEF_CHF_QOS_Create<br>OCNEF_CHF_QOS_Delete<br>OCNEF_CHF_QOS_Metrics<br>OCNEF_CHF_QOS_Notification<br>OCNEF_CHF_QOS_Update<br>OCNEF_Qos_Create_Subscription<br>OCNEF_Qos_Delete_Subscription<br>OCNEF_Qos_Get_Subscription<br>OCNEF_Qos_Metrics<br>OCNEF_Qos_Notification<br>OCNEF_Qos_Update_with_PUT_PATCH |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | TI_PCF                                         | OCNEF_TI_Create_Subscription<br>OCNEF_TI_Delete_Subscription<br>OCNEF_TI_Get_Subscription<br>OCNEF_TI_Notification<br>OCNEF_TI_Update_with_PUT_PATCH                                                                                                                                                                                  |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-A   | TI_BSF                                         | OCNEF_TI_Create_Subscription<br>OCNEF_TI_Delete_Subscription<br>OCNEF_TI_Get_Subscription                                                                                                                                                                                                                                             |
| Nef Model-B   |                                                |                                                                                                                                                                                                                                                                                                                                       |
| Nef Model-D   |                                                |                                                                                                                                                                                                                                                                                                                                       |

**Table 4-5 (Cont.) Execution Suite Configuration Mapping**

| Execute_Suite | Suite_Configuration       | Features                                                                                                                                                                                                                                                                                                                  |
|---------------|---------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nef Model-A   | TI_UDR                    | OCNEF_TI_Create_Subscription                                                                                                                                                                                                                                                                                              |
| Nef Model-B   |                           | OCNEF_TI_Get_Subscription                                                                                                                                                                                                                                                                                                 |
| Nef Model-D   |                           | OCNEF_TI_Metrics<br>OCNEF_TI_Update_with_PUT_PATCH                                                                                                                                                                                                                                                                        |
| Nef Model-A   | ME - SCEF+NEF             | OCNEF_SCEF_EPC_Notification                                                                                                                                                                                                                                                                                               |
| Nef Model-B   |                           | OCNEF_SCEF_NEF_GET_ME                                                                                                                                                                                                                                                                                                     |
| Nef Model-D   |                           | OCNEF_SCEF_NEF_GET_ME_Scenarios<br>OCNEF_SCEF_NEF_ME_AF_Unavailable<br>OCNEF_SCEF_NEF_ME_Errors_Scenarios<br>OCNEF_SCEF_NEF_ME_MultipleNotifications<br>OCNEF_SCEF_NEF_ME_OneTimeNotification<br>OCNEF_SCEF_NEF_ME_Sub_UnSub                                                                                              |
| Nef Model-A   | ME - PDU Session          | OCNEF_ME_PDU_Notification                                                                                                                                                                                                                                                                                                 |
| Nef Model-B   |                           | OCNEF_ME_PDU_PUT                                                                                                                                                                                                                                                                                                          |
| Nef Model-D   |                           | OCNEF_ME_PDU_SUBSCRIPTION                                                                                                                                                                                                                                                                                                 |
| Nef Model-A   | ME - Loss of Connectivity | OCNEF_ME_LossOfCon                                                                                                                                                                                                                                                                                                        |
| Nef Model-B   |                           | OCNEF_ME_LossOfCon_PUT                                                                                                                                                                                                                                                                                                    |
| Nef Model-D   |                           | OCNEF_ME_LossOfConn_MultipleNotifications<br>OCNEF_ME_LossOfConn_OneTimeNotification                                                                                                                                                                                                                                      |
| Nef Model-A   | ME - UeReachability       | OCNEF_ME_UE_REACHABILITY_MultipleNotifications                                                                                                                                                                                                                                                                            |
| Nef Model-B   |                           | OCNEF_ME_UE_REACHABILITY_OneTimeNotification                                                                                                                                                                                                                                                                              |
| Nef Model-D   |                           | OCNEF_ME_UE_Reachability<br>OCNEF_ME_UE_Reachability_PUT                                                                                                                                                                                                                                                                  |
| Nef Model-A   | Qos+CHF - SCEF+NEF        | OCNEF_CHF_SCEF_Qos_Create_Subscription                                                                                                                                                                                                                                                                                    |
| Nef Model-B   |                           | OCNEF_CHF_SCEF_Qos_Delete_Subscription                                                                                                                                                                                                                                                                                    |
| Nef Model-D   |                           | OCNEF_CHF_SCEF_Qos_Metrics<br>OCNEF_CHF_SCEF_Qos_Notification<br>OCNEF_CHF_SCEF_Qos_Update_with_PUT_PATCH<br>OCNEF_SCEF_Qos_Create_Subscription<br>OCNEF_SCEF_Qos_Delete_Subscription<br>OCNEF_SCEF_Qos_Get_Subscription<br>OCNEF_SCEF_Qos_Metrics<br>OCNEF_SCEF_Qos_Notification<br>OCNEF_SCEF_Qos_Update_with_PUT_PATCH |
| Nef Model-A   | Device Trigger            | Ocnef_DT_Notification                                                                                                                                                                                                                                                                                                     |
| Nef Model-B   |                           | Ocnef_DT_Sub_Unsub                                                                                                                                                                                                                                                                                                        |
| Nef Model-D   |                           |                                                                                                                                                                                                                                                                                                                           |

4. Select any of following valid values from **Configuration\_Type**:

- **Product\_Config**: On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the key-value pair and yaml files defined or present in the "Product Config" folder.
- **Custom\_Config**: On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the key-value pair and yaml files defined or present in the "Custom Config" folder.

5. Select any of the following options under the **Select\_Option** section:

- **All**: This is the default option. It runs all the NEF test cases. Scroll down and click **Build** to execute all the test cases.
- **Single/MultipleFeatures**: This option allows you to select any number of test cases that you want to run from the list of total test cases available for execution.

- After selecting the test cases, click **Build** and select **Console Output** to view the test results. The following is a sample test result output:

**Figure 4-22 Sample Test Cases Result - All-Regression**

```

9 features passed, 0 failed, 0 skipped
53 scenarios passed, 0 failed, 0 skipped
1456 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m42.237s
[Pipeline] sh
+ cd /var/lib/jenkins/ocnef_tests
++ cat /var/lib/jenkins/ocnef_tests/envIRON.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=0
+ export nef_data_path=/var/lib/jenkins/ocnef_tests/data/
+ nef_data_path=/var/lib/jenkins/ocnef_tests/data/
+ sh re-run.sh 0
1
Scenario passed
[Pipeline] sh
+ cd /var/lib/jenkins/ocnef_tests
++ cat /var/lib/jenkins/ocnef_tests/envIRON.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=0
+ export nef_data_path=/var/lib/jenkins/ocnef_tests/data/
+ nef_data_path=/var/lib/jenkins/ocnef_tests/data/
+ sh report.sh 0 file_name=outputresult
Final Result:-
Total-Features RUN 9, PASS 9, FAIL 0

```

#### ① Note

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

## 4.2.6 NEF-Regression Documentation

This section describes the documentation for NEF-Regression pipeline.

To view the documentation for any of the NEF features, click **NEF-Regression** on the ATS home page. Then, click **Documentation** in the left navigation pane.

This page shows features of only those test cases that are released in previous releases.

The NEF-Regression test cases are divided into multiple groups based on the functionality.

- ME Location Reporting-GMLC
- ME Location Reporting-UDM
- ME Location Reporting-GMLC-Failover-UDM
- AsQos+CHF\_PCF
- AsQos+CHF\_BSF
- TI\_PCF

- TI\_BSF
- TI\_UDR
- ME - SCEF+NEF
- ME - PDU Session
- ME - Loss of Connectivity
- ME - UeReachability
- Qos+CHF - SCEF+NEF
- Device Trigger

## 4.3 Running NRF Test Cases using ATS

This section describes how to run NRF test cases using ATS. It includes:

- [Prerequisites](#)
- [Logging into ATS](#)
- [NRF-NewFeatures Pipeline](#)
- [NRF-NewFeatures Documentation](#)
- [NRF-Regression Pipeline](#)
- [NRF-Regression Documentation](#)

### 4.3.1 Prerequisites

To run NRF test cases using NRF ATS 24.1.x, you need to ensure that following prerequisites are fulfilled.

- For running three-site georedundancy test cases, NRF 24.1.x should be deployed on three sites with DB replication enabled. Georedundancy feature **must** be enabled on all the NRF sites before running these test cases.
- For running two-site georedundancy test cases, NRF 24.1.x should be deployed on two sites with DB replication enabled. These test cases are executed separately as it requires two different NRFs. These can be executed on a three-site georedundancy setup as well. Georedundancy feature **must** be enabled on all the NRF sites before running these test cases.
- For running NRF-Growth test cases, two standalone NRF 24.1.x instances should be deployed. These test cases are executed separately as it requires two different NRFs.
- All the ATS pipelines other than georedundancy, three-site georedundancy, and NRF Growth must be executed with only one NRF deployment (standalone NRF).
- The following NRF database tables should not have any entries:
  - nrfApplicationDB.NfInstances
  - nrfApplicationDB.NfStatusMonitor
  - nrfApplicationDB.NfSubscriptions
- NRF/s should be deployed with the default helm and REST configurations, other than ocnrfHost and ocnrfPort.
- To run NF-FQDN-Authentication-Feature test cases, deploy NRF and NRF ATS, both with NF Authentication feature changes.

- All microservices of NRF must be up and running including **Alternate route** and **Artisan** microservice.
- For running Alerts test cases, configure NRF Alerts on the AlertManager on the Prometheus Server. For more details on configuring alerts, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
- Before deploying NRF, user should create RSA and ECDSA certificates and keys (public and private) for AccessToken microservice.
- Deploy NRF 24.1.x with default helm configurations using helm charts to run all test cases except **NF-FQDN-Authentication-Feature** cases.
- Ensure all microservices of NRF should be up and running including Accesstoken microservice. A key with RSA and another with ECDSA algorithm is mandatory.
- Copy the public keys (RSA and ECDSA) to the ATS pod at the `/var/lib/jenkins/ocnrf_tests/public_keys` location, with the names `ec_public.pem` and `rs_public.pem` respectively for ECDSA and RSA public keys. Create a copy of `rs_public.pem` and save it as `new_public.pem` in the same location.
- Create JSON files for key and certificate details and copy them to ATS at `/var/lib/jenkins/ocnrf_tests/` location. Refer to [KID feature specific Prerequisite](#).
- Deploy ATS using helm charts.
- For NRF 24.1.x, deploy five stub servers for running Roaming, SLF, Forwarding, and SLF via SCP functionality test cases. The service name for the stub servers must be **notify-stub-service**, **notify-stub-service02**, **notify-stub-service03**, **amf1-cluster1-net2-amf-5gc-mnc016-mcc310-3gppnetwork-org**, and **slf-stub-service01**.
- The **slf-stub-service01** stub must be deployed with **service port as 80**. For more information, see [Installing ATS for NRF](#).
- Replace the existing content with the following content in the alternate-route section of the `ocnrf_custom_values.yaml` file:  
The following configuration changes are required to accommodate alternate routing test cases based on the latest configuration.

```
staticVirtualFqdns:
 - name: https://abc.test.com
 alternateFqdns:
 - target: notify-stub-service03
 port: 8080
 priority: 10
 - target: notify-stub-service03
 port: 8080
 priority: 20
 - name: http://xyz.test.com
 alternateFqdns:
 - target: notify-stub-service03
 port: 8080
 priority: 10
 - target: notify-stub-service03
 port: 8080
 priority: 20

dnsSrvEnabled: false
```

- Ensure Prometheus service is up and running.

- Deploy ATS and Stubs in the same namespace as NRF, as default ATS deployment is with role binding. In addition, deploy test stubs in the same namespace as NRF.
- User must not initiate a job in two different pipelines at the same time.
- User must not abort a running job, this may lead to data corruption in database.
- For getting PVC support to retain NRF ATS environment variables and pipeline console histories, follow the [installation step](#). By default PVC support is disabled.
- If installation is done with Service mesh, the NRF must be deployed with below annotation in **lbDeployments** and **nonlbDeployments** section:

```
oracle.com/cnc: "true"
traffic.sidecar.istio.io/excludeInboundPorts: "9090,8095,8096,7,53"
traffic.sidecar.istio.io/excludeOutboundPorts: "9090,8095,8096,7,53"
```

where, first one is for connecting to Operations Services Overlay (OSO) for Alerts test cases. Second one is for connecting to ATS for fetching the metrics from pods.

- If Service Mesh is enabled, then create a destination rule to fetch the metrics from the Prometheus. In most of the deployments, Prometheus is kept outside the service mesh and a destination rule is required to communicate between TLS enabled entity (ATS) and non-TLS enabled entity (Prometheus).

To create a rule:

```
kubectl apply -f - <<EOF
apiVersion:networking.istio.io/v1alpha3
kind:DestinationRule
metadata:
 name:prometheus-dr
 namespace:ocnrf
spec:
 host:oso-prometheus-server.ocnrf.svc.cluster.local
 trafficPolicy:
 tls:
 mode:DISABLE
EOF
```

In the above rule,

- **name** indicates the name of destination rule.
- **namespace** indicates where the ATS is deployed.
- **host** indicates the hostname of the Prometheus server.
- In case of NRF is deployed with network policy enabled, then before NRF-ATS execution a new network policy must be created to access NRF-ATS Jenkins GUI. Following is the sample:

```
kubectl apply -f nodePort.yaml -n ocnrf
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
 name: allow-from-node-port
spec:
 podSelector:
 matchLabels:
```

```

 app: ocats-nrf
 policyTypes:
 - Ingress
 ingress:
 - ports:
 - protocol: TCP
 port: 8080

```

### Custom Folder Implementation

ATS provides custom test case folders (cust\_newfeatures, cust\_regression and cust\_performance) using which you can add new test cases, remove unwanted test cases, and modify existing test cases. It does not impact the original product packaged test cases available in the newfeatures, regression and performance folders. For more details, refer to [Custom Folder Implementation](#).

Apart from cust\_newfeatures and cust\_regression, NRF ATS has cust\_FQDNauthentication, cust\_GeoRedundancy, and cust\_3SiteGeo directories that contains test cases related to NF-FQDN and georedundancy respectively.

### Key Identifier (KID) feature specific Prerequisite

User should create the following files with exact names and update the file content as per the naming convention used for keys and certificate along with secret name and namespace. The file names are:

- set\_keyDetailsList.json
- set\_currentKeyId.json
- tokenSigningDetails.json

### Sample Files

```

$ cat set_keyDetailsList.json
{
 "tokenSigningDetails": {
 "defaultK8SecretDetails": {
 "k8SecretNameSpace": "ocnrf",
 "k8SecretName": "ocnrfaccesstoken-secret"
 },
 "keyDetailsList": [{
 "keyID": "EcdsaKid",
 "algorithm": "ES256",
 "privateKey": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "ecdsa_private_key.pem"
 },
 "certificate": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "ecdsa_certificate.crt"
 }
 }
],
 {

```



```

 "keyID": "RsaKid",
 "algorithm": "RS256",
 "privateKey": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "rsa_private_key.pem"
 },
 "certificate": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "rsa_certificate.crt"
 }
 },
 {
 "keyID": "newKey",
 "algorithm": "RS256",
 "privateKey": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "rsa_private_key.pem"
 },
 "certificate": {
 "k8SecretName": "ocnrfaccesstoken-
secret",
 "k8SecretNameSpace": "ocnrf",
 "fileName": "rsa_certificate.crt"
 }
 }
}
]
}
}

```

#### ① Note

Update the following parameters in the above files:

- k8SecretName
- k8SecretNameSpace
- fileName (name of the private keys and corresponding certificates)

For more information about parameter values, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

After updating the files, copy them under the **/var/lib/jenkins/ocnrf\_tests/** directory.

```

$ cat set_currentKeyId.json
{
 "tokenSigningDetails":
 {"currentKeyId": "EcdsaKid"

```

```
 }
}
```

#### **Note**

User need not update the currentKeyId name (for example: EcdsaKid), if the same file name as in keyID is used while configuring the set\_currentKeyId.json file.

```
$ cat tokenSigningDetails.json
{
 "presentCurrentKey" : {
 "algorithm" : "ES256",
 "privateKey" : {
 "k8sSecretName" : "ocnrfaccesstoken-secret",
 "fileName" : "ecdsa_private_key.pem"
 },
 "certificate" : {
 "k8sSecretName" : "ocnrfaccesstoken-secret",
 "fileName" : "ecdsa_certificate.crt"
 }
 },
 "newCurrentKey" : {
 "algorithm" : "RS256",
 "privateKey" : {
 "k8sSecretName" : "ocnrfaccesstoken-secret",
 "fileName" : "rsa_private_key.pem"
 },
 "certificate" : {
 "k8sSecretName" : "ocnrfaccesstoken-secret",
 "fileName" : "rsa_certificate.crt"
 }
 }
}
```

```
$ cat set_currentKeyId.json
{
 "tokenSigningDetails":
 {"currentKeyId":"EcdsaKid"}
}
```

#### **Note**

User doesn't want to update the name of the key if the same name was used while configuring.

```
$ cat tokenSigningDetails.json
{
 "presentCurrentKey" : {
 "algorithm" : "ES256",
```

```

 "privateKey" : {
 "k8sSecretName" : "ocnrfacesstoken-secret",
 "fileName" : "ecdsa_private_key.pem"
 },
 "certificate" : {
 "k8sSecretName" : "ocnrfacesstoken-secret",
 "fileName" : "ecdsa_certificate.crt"
 }
 },
 "newCurrentKey" : {
 "algorithm" : "RS256",
 "privateKey" : {
 "k8sSecretName" : "ocnrfacesstoken-secret",
 "fileName" : "rsa_private_key.pem"
 },
 "certificate" : {
 "k8sSecretName" : "ocnrfacesstoken-secret",
 "fileName" : "rsa_certificate.crt"
 }
 }
}
}
}

```

#### ① Note

Here, user should update k8SecretName fileName (name of private keys and the corresponding certificates).

#### ① Note

Update the following parameters in the above code snippet:

- k8SecretName
- fileName (name of the private keys and corresponding certificates)

For more information about parameter values, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

After updating all the files, copy them to the ATS pod using "**kubectl cp**" command. Refer to the following sample commands:

```

$ kubectl cp tokenSigningDetails.json ocats-ocats-nrf-58fc5dcbb9-
zjsks:/var/lib/jenkins/ocnrf_tests/tokenSigningDetails.json -n
ocnrf
$ kubectl cp set_keyDetailsList.json ocats-ocats-nrf-58fc5dcbb9-
zjsks:/var/lib/jenkins/ocnrf_tests/set_keyDetailsList.json -n ocnrf
$ kubectl cp set_currenKeyId.json ocats-ocats-nrf-58fc5dcbb9-zjsks:/var/lib/
jenkins/ocnrf_tests/set_currenKeyId.json -n ocnrf
$ kubectl cp ec_public.pem ocats-ocats-nrf-58fc5dcbb9-zjsks:/var/lib/jenkins/
ocnrf_tests/public_keys/ec_public.pem -n ocnrf
$ kubectl cp rs_public.pem ocats-ocats-nrf-58fc5dcbb9-zjsks:/var/lib/jenkins/
ocnrf_tests/public_keys/rs_public.pem -n ocnrf

```

```
$ kubectl cp rs_public.pem ocats-ocats-nrf-58fc5dcbb9-zjsks:/var/lib/jenkins/ocnrf_tests/public_keys/new_public.pem -n ocnrf
```

### Prerequisites to Run Testcases for the CCA-Header Feature

To run the pipeline for the CCA-Header feature, NRF must be deployed with the following changes:

1. During installation of NRF, the namespace, secret name and the caroot.cer file name must be provided in NRF's custom values, in ingress-gateway section, as follows:

```
ingress-gateway:
 ccaHeaderValidation:
 k8SecretName: ocingress-secret
 k8NameSpace: ocnrf
 fileName: caroot.cer
```

2. In NRF's custom values, metadata.ccaHeaderValidation.enabled should be set to true in the '**accesstoken\_mapping**' object from the ingress-gateway.routesConfig list.

#### Note

If ccaHeaderValidation is enabled in NRF, only **CCA-Header** pipeline should be run. The other pipelines (Newfeatures, Regression, 2SiteGeo, 3SiteGeo, or FQDN-Authentication) will not work with the feature enabled.

```
ingress-gateway:
 routesConfig:
 - id: accesstoken_mapping
 uri: http://{{ template "accesstoken.service.fullname" . }}:
{{ template "accesstoken.service.port" . }}
 path: /oauth2/token
 order: 4
 filters:
 controlledShutdownFilter:
 applicableShutdownStates:
 - COMPLETE_SHUTDOWN
 metadata:
 ccaHeaderValidation:
 enabled: true
```

3. Create a secret with the caroot.cer file provided in the **ocats\_ocnrf\_tests\_jenkinsjobs\_24.1.3.tgz**, using the following command. This same caroot.cer file is required to be configured in the secret to execute the CCA-Header cases.

```
kubectl create secret generic <secret name> --from-file=<ca root file name> -n <namespace>
```

For example:

```
kubectl create secret generic ocnrfgress-secret --from-file=caroot.cer -n
ocnrf
```

4. Update the secret name from step 3 in **global.yaml** file in either **/var/lib/jenkins/ocnrf\_tests/product\_config** or **/var/lib/jenkins/ocnrf\_tests/custom\_config** based on the whether the pipeline will be executed with **Product\_Config** option or **Custom\_Config** option, under the **ccaHeaderSecretName** parameter.

### Prerequisites to Run Testcases for the NRF Growth Features

To run the pipeline for the NRF Growth feature, the following prerequisites have to be done.

1. Install two NRFs that are not georedundant with each other.
2. The pipeline configuration has to be set to the below before running the pipeline. The values of options "u" to "x" have to be set to 0.1.

```
node ('master'){
 //a = SELECTED_NF b = NF_NAMESPACE c = FT_ENDPOINT d =
GATEWAY_IP e = GATEWAY_PORT
 //f = CONFIG_IP g = CONFIG_PORT h = STUB_IP i =
STUB_PORT j = NFINSTANCEID
 //k = PROMETHEUS_IP l = PROMETHEUS_PORT m = RERUN_COUNT n =
PROMETHEUS_URI o = HELM_RELEASE_NAME
 //p = MYSQL_HOST q = PRIVILEGED_USER_SECRET_NAME r =
DISCOVERY_WAIT_TIME
 //s = REPLICATION_STATUS_URI t = CLUSTER_DOMAIN u =
REGISTER_WRITE_WAIT_TIME
 //v = SUBSCRIPTION_WAIT_TIME w =
ACCESSTOKEN_WAIT_TIME x = REGISTER_READ_WAIT_TIME
 sh '''
 sh /var/lib/jenkins/ocnrf_tests/preTestConfig.sh \
 -a NRF \
 -b occne-cndbtierone,occne-cndbtiertwo \
 -c ocnrf-ingressgateway.ocnrf.svc.cluster.local:80,ocnrf-1-
ingressgateway.ocnrf.svc.cluster.local:80 \
 -d ocnrf-ingressgateway.occne-cndbtierone,ocnrf-1-
ingressgateway.occne-cndbtiertwo \
 -e 80,80 \
 -f ocnrf-nrfconfiguration.occne-cndbtierone,ocnrf-1-
nrfconfiguration.occne-cndbtiertwo \
 -g 8080,8080 \
 -h notify-stub-service.occne-cndbtierone,notify-stub-
service02.occne-cndbtierone \
 -i 8080,8080 \
 -j 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c,6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5d \
 -k oso-prom-svr.oso \
 -l 80 \
 -m 2 \
 -n /prometheus/api/ \
 -o ocnrf,ocnrf-1 \
 -p mysql-connectivity-service.occne-cndbtierone,mysql-connectivity-
service.occne-cndbtiertwo \
 -q privilegeduser-secret \
```

```

-r 0.5 \
-s http://ocnrf-ocnrf-app-info:5906/status/category/
replicationstatus,http://ocnrf-1-ocnrf-app-info:5906/status/category/
replicationstatus \
-t svc.cluster.local \
-u 0.1 \
-v 0.1 \
-w 0.1 \
-x 0.1
'''
if(env.Include_NewFeatures && "${Include_NewFeatures}" == "YES"){
sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
load "/var/lib/jenkins/ocnrf_tests/jenkinsData/Jenkinsfile-Merged"
}
else{
load "/var/lib/jenkins/ocnrf_tests/jenkinsData/Jenkinsfile-
Regression"
}
}
}
}

```

## 4.3.2 Logging into ATS

Before logging into ATS, you need to ensure that ATS is deployed successfully using HELM charts as shown below:

### Note

To modify default log in password, refer to [Modifying Login Password](#).

Figure 4-23 Verifying ATS Pod

```
(root@master ~)# kubectl get pods -n ocnrf -o wide
```

| NAME                                     | READY | STATUS  | RESTARTS | AGE  | IP              | NODE   | NOMINATED NODE | READINESS GATES |
|------------------------------------------|-------|---------|----------|------|-----------------|--------|----------------|-----------------|
| ocats-ocats-nrf-66f9cc5c64-xx1dt         | 2/2   | Running | 0        | 105m | 192.168.140.111 | slave2 | <none>         | <none>          |
| ocnrf-appinfo-86949930dc-1v2sp           | 2/2   | Running | 0        | 106m | 192.168.179.112 | slave5 | <none>         | <none>          |
| ocnrf-egressgateway-6c6bf9b6bf-9pf44     | 2/2   | Running | 0        | 166m | 192.168.179.111 | slave5 | <none>         | <none>          |
| ocnrf-ingressgateway-64574bbb85-w5rxb    | 2/2   | Running | 0        | 166m | 192.168.179.109 | slave5 | <none>         | <none>          |
| ocnrf-nfaccess-token-fb8dc566d-2cjzs     | 3/3   | Running | 0        | 166m | 192.168.140.124 | slave2 | <none>         | <none>          |
| ocnrf-nfdiscovery-bbfdc4c7b-m46cs        | 2/2   | Running | 0        | 166m | 192.168.77.44   | slave3 | <none>         | <none>          |
| ocnrf-nfregistration-6cdddf76dc-5vst8    | 2/2   | Running | 0        | 166m | 192.168.77.30   | slave3 | <none>         | <none>          |
| ocnrf-nfsubscription-76f5fcf4b7-mjq7r    | 2/2   | Running | 0        | 166m | 192.168.140.84  | slave2 | <none>         | <none>          |
| ocnrf-nrf-auditor-74d6b97ffc-2xdp6       | 2/2   | Running | 0        | 166m | 192.168.77.26   | slave3 | <none>         | <none>          |
| ocnrf-nrf-configuration-65fc5fffb-drt45f | 2/2   | Running | 0        | 166m | 192.168.77.46   | slave3 | <none>         | <none>          |
| ocstub-ocstub-python-7c7d4d865-plsc8     | 2/2   | Running | 0        | 103m | 192.168.140.112 | slave2 | <none>         | <none>          |
| ocstub1-ocstub-python-7b7cfdcc4-cng5g    | 2/2   | Running | 0        | 103m | 192.168.179.115 | slave5 | <none>         | <none>          |

```
(root@master ~)# kubectl get svc -n ocnrf
```

| NAME                                | TYPE         | CLUSTER-IP     | EXTERNAL-IP                                                 | PORT(S)                      | AGE   |
|-------------------------------------|--------------|----------------|-------------------------------------------------------------|------------------------------|-------|
| mysql-connectivity-service          | ExternalName | <none>         | mysql-connectivity-service-headless.ocnrf.svc.cluster.local | <none>                       | 3d19h |
| mysql-connectivity-service-headless | ClusterIP    | None           | <none>                                                      | 3306/TCP                     | 3d19h |
| notify-stub-service                 | ClusterIP    | 10.99.89.123   | <none>                                                      | 8080/TCP, 8091/TCP, 8443/TCP | 103m  |
| notify-stub-service02               | ClusterIP    | 10.110.95.204  | <none>                                                      | 8080/TCP, 8091/TCP, 8443/TCP | 103m  |
| ocats-ocats-nrf                     | LoadBalancer | 10.105.39.88   | <pending>                                                   | 8080:30348/TCP               | 107m  |
| ocnrf-egressgateway                 | ClusterIP    | 10.106.98.158  | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-ingressgateway                | ClusterIP    | 10.103.128.84  | <none>                                                      | 80/TCP                       | 166m  |
| ocnrf-nfaccess-token                | ClusterIP    | 10.99.116.111  | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-nfdiscovery                   | ClusterIP    | 10.97.51.196   | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-nfregistration                | ClusterIP    | 10.99.114.145  | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-nfsubscription                | ClusterIP    | 10.110.182.153 | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-nrf-auditor                   | ClusterIP    | 10.103.17.95   | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-nrf-configuration             | ClusterIP    | 10.110.125.184 | <none>                                                      | 8080/TCP                     | 166m  |
| ocnrf-ocnrf-app-info                | ClusterIP    | 10.111.114.179 | <none>                                                      | 5906/TCP                     | 166m  |

```
(root@master ~)# kubectl get nodes -o wide
```

| NAME   | STATUS | ROLES  | AGE | VERSION | INTERNAL-IP   | EXTERNAL-IP | OS-IMAGE                | KERNEL-VERSION                | CONTAINER-RUNTIME |
|--------|--------|--------|-----|---------|---------------|-------------|-------------------------|-------------------------------|-------------------|
| master | Ready  | master | 45d | v1.17.1 | 10.75.225.192 | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |
| slave1 | Ready  | <none> | 45d | v1.17.1 | 10.75.225.41  | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |
| slave2 | Ready  | <none> | 45d | v1.17.1 | 10.75.225.221 | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |
| slave3 | Ready  | <none> | 45d | v1.17.1 | 10.75.228.87  | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |
| slave4 | Ready  | <none> | 45d | v1.17.1 | 10.75.226.104 | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |
| slave5 | Ready  | <none> | 45d | v1.17.1 | 10.75.225.50  | <none>      | Oracle Linux Server 7.5 | 4.1.12-112.16.4.el7uek.x86_64 | docker://19.3.11  |

```
(root@master ~)#
```

For more information on verifying ATS deployment, see [Verifying ATS Deployment](#).

**Note**

In the **Verifying ATS Pod** screen, **slave2** is the node where ATS is deployed, **30348** is the ATS nodeport and **10.75.225.227** is the worker node IP, highlighted in red. For more details on ATS deployment, refer to [Installing ATS for NRF](#).

To log in to ATS, open a browser and provide the IP Address and port details as <Worker-Node-IP>:<Node-Port-of-ATS>. As per above screen, it is **10.75.225.227:30348**.

**Figure 4-24**    ATS Login

## Oracle Communications Cloud Native Core - Automated Test Suite

Welcome to ATS!

Username

Password

☐ Keep me signed in

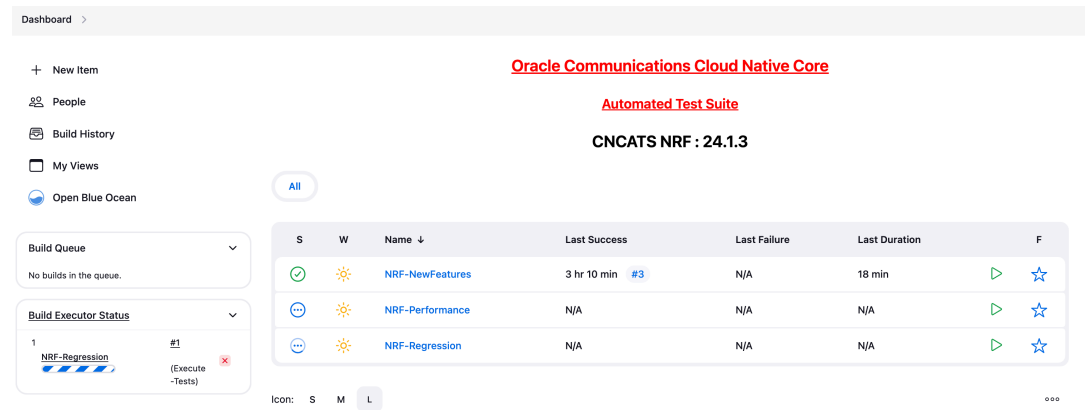
Sign in

### Running ATS

To run ATS:

1. Enter the login credentials. Click **Sign in**. The following screen appears.

Figure 4-25 NRF Pre-Configured Pipelines



NRF ATS has three pre-configured pipelines.

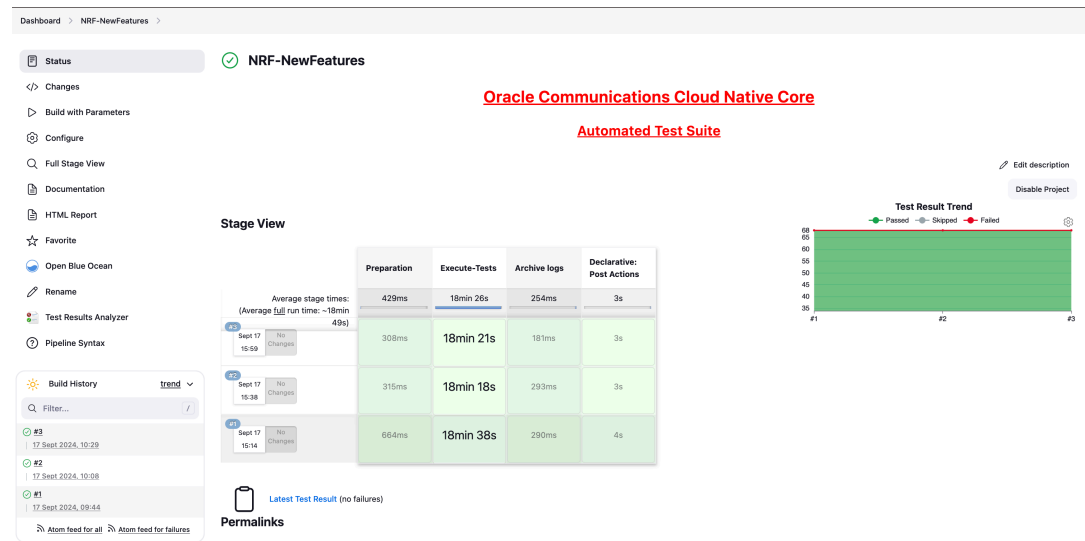
- **NRF-NewFeatures:** This pipeline has all the test cases delivered as part of NRF ATS - 24.1.3.
- **NRF-Performance:** This pipeline is not operational as of now. It is reserved for future releases of ATS.
- **NRF-Regression:** This pipeline has all the test cases delivered so far in the previous releases.

### 4.3.3 NRF-NewFeatures Pipeline

After identifying the NRF pipelines, configure ATS as one-time activity as per NRF deployment. In this pipeline, all the new testcases related to NRF are deployed. To configure its parameters:

1. Click **NRF-NewFeatures** in the **Name** column. The following screen appears:

Figure 4-26 Configuring NRF-New Features





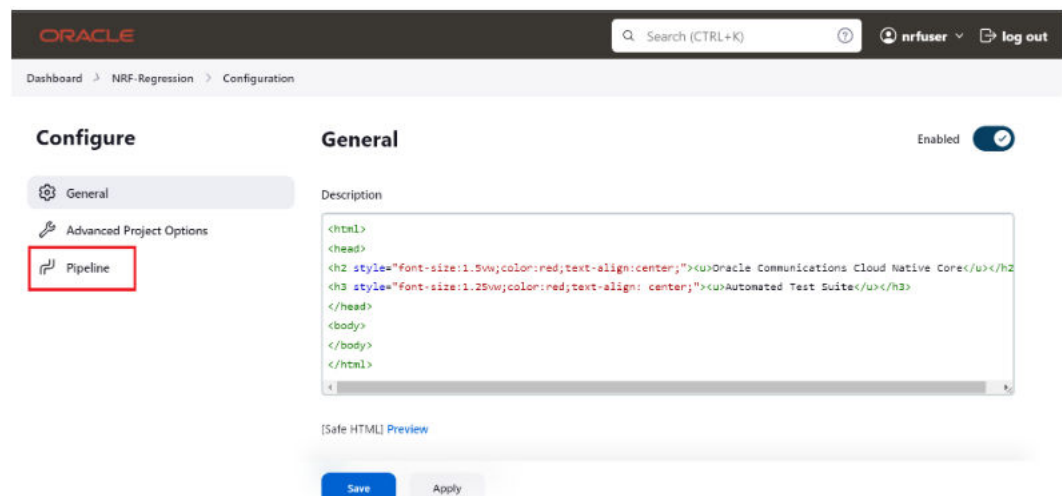
In the above screen:

- Click **Configure** to configure NRF-New Features.
  - Click **Documentation** to view the documented test cases, which are part of this NRF release.
  - Click the blue dots inside **Build History** box to view the success console logs of the "Sanity", "All-NewFeatures" respectively.
  - The **Stage View** represents the already deployed pipeline for the customer reference.
  - The **Test Results Analyzer** is the new plugin integrated in the NRF-ATS. This option can be used to display the previous history of all the execution build-wise. It will provide a graphical representation of the past execution together.
2. Click **Configure**. User must wait for the page to load completely. Once the page loads completely, click the **Pipeline** tab:

**Note**

Make sure that the following page loads completely before you perform any action on it. Also, do not modify any configuration other than shown below.

**Figure 4-27 Pipeline Tab**



The **Pipeline** section of the configuration page appears as follows:

**Figure 4-28 Pipeline Section**

```

node ('built-in'){
 //a = SELECTED_NF b = NF_NAMESPACE c = FT_ENDPOINT d = GATEWAY_IP e = GATEWAY_PORT
 //f = CONFIG_IP g = CONFIG_PORT h = STUB_IP i = STUB_PORT j = NFINSTANCEID
 //k = PROMETHEUS_IP l = PROMETHEUS_PORT m = RERUN_COUNT n = PROMETHEUS_URI o = HELM_RELEASE_NAME
 //p = MYSQL_HOST q = PRIVILEGED_USER_SECRET_NAME r = DISCOVERY_WAIT_TIME
 //s = REPLICATION_STATUS_URI t = CLUSTER_DOMAIN u = REGISTER_WRITE_WAIT_TIME
 //v = SUBSCRIPTION_WAIT_TIME w = ACCESSTOKEN_WAIT_TIME x = REGISTER_READ_WAIT_TIME
 withEnv([

]){

 sh '''
 sh /var/lib/jenkins/ocnrf_tests/preTestConfig.sh \
 -a NRF \
 -b ocnrf \
 -c ocnrf-ingressgateway.ocnrf.svc.cluster.local:80 \
 -d ocnrf-ingressgateway.ocnrf \
 -e 80 \
 -f ocnrf-nrfconfiguration.ocnrf \
 -g 8080 \
 -h notify-stub-service.ocnrf \
 -i 8080 \
 -j 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c \
 -k oso-prom-svr.oso \
 -l 80 \
 -m 2 \
 -n /prometheus/api/ \
 -o ocnrf \
 -p mysql-connectivity-service.ocnrf \
 -q privilegeduser-secret \
 -r 0.5 \
 -s http://ocnrf-ocnrf-app-info:5906/status/category/replicationstatus \
 -t svc.cluster.local \
 -u 0 \
 -v 0 \
 -w 0 \
 -x 0
 '''
 }
}

```

In the above screenshot, change the values of the **'Pipeline script'**. The content of the pipeline script is as follows:

**Figure 4-29 Pipeline Script**

```

1 node ('built-in'){
2 //a = SELECTED_NF b = NF_NAMESPACE c = FT_ENDPOINT d = GATEWAY_IP e = GATEWAY_PORT
3 //f = CONFIG_IP g = CONFIG_PORT h = STUB_IP i = STUB_PORT j = NFINSTANCEID
4 //k = PROMETHEUS_IP l = PROMETHEUS_PORT m = RERUN_COUNT n = PROMETHEUS_URI o = HELM_RELEASE_NAME
5 //p = MYSQL_HOST q = PRIVILEGED_USER_SECRET_NAME r = DISCOVERY_WAIT_TIME
6 //s = REPLICATION_STATUS_URI t = CLUSTER_DOMAIN u = REGISTER_WRITE_WAIT_TIME
7 //v = SUBSCRIPTION_WAIT_TIME w = ACCESSTOKEN_WAIT_TIME x = REGISTER_READ_WAIT_TIME
8 withEnv([
9
10]){
11
12 sh '''
13 sh /var/lib/jenkins/ocnrf_tests/preTestConfig.sh \
14 -a NRF \
15 -b ocats \
16 -c ocnrf-ingressgateway.ocats:80 \
17 -d ocnrf-ingressgateway.ocats \
18 -e 80 \
19 -f ocnrf-nrfconfiguration.ocats \
20 -g 8080 \
21 -h notify-stub-service.ocats \
22 -i 8080 \
23 -j 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c \
24 -k oso-prom-svr.oso \
25 -l 80 \
26 -m 0 \
27 -n /prometheus/api/ \
28 -o ocnrf \
29 -p database-mysql-cne \
30 -q privilegeduser-secret \
31 -r 0.5 \
32 -s http://ocnrf-ocnrf-app-info:5906/status/category/replicationstatus \
33 -t thrust6 \
34 -u 0 \
35 -v 0 \
36 -w 0 \
37 -x 0
38 '''
39
40 if(env.Include_Regression && "${Include_Regression}" == "YES"){
41 sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
42 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/Jenkinsfile-Merged"
43 }
44 else{
45 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/Jenkinsfile-NewFeatures"
46 }
47 }
48 }

```

**Note**

The user must not change any other value apart from **line number 15 to line 37**.

You can change the parameter values from "a" - to - "x" as per user requirement. The parameter details are available as comments from line number 2 - to - 7.

- **a:** Name of the NF to be tested in capital (NRF).
- **b:** Namespace in which the NRF is deployed.
- **c:** endPointIP: endPointPort value used while deploying NRF with the help of helm chart.
- **d:** Comma separated values of NRF1 and NRF2 ingress gateway service (For example: ocnrf-ingressgateway.ocnrf,1.1.1.1). It is also known as cluster\_domain.
- **e:** Comma separated values of NRF1 and NRF2 port of ingressgateway service (For example: 80,31000).
- **f:** Comma separated values of NRF1 and NRF2 configuration service (For example: ocnrf-nrfconfiguration.ocnrf,1.1.1.1). It is also known as cluster\_domain.
- **g:** Comma separated values of NRF1 and NRF2 port of configuration service (For example: 8080, 31001).
- **h:** Name\_of\_stub\_service.namespace (notify-stub-service.ocnrf).
- **i:** Port of stub service (8080).
- **j:** NRF\_Instance ID (6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c).
- **k:** Name\_of\_Prometheus\_service.namespace (occne-prometheus-server.occne-infra).
- **l:** Port of Prometheus service (80).
- **m:** Number of times the re-run of failed case is allowed (default as 2).
- **n:** URI of Prometheus service (/api) -
  - Use "/api/" for CNE version till 1.6.0 and "/prometheus/api/" for CNE version 1.7.0 and 1.8.0.
  - For, **CNE 1.9.x**, **CNE 1.10.x**, and **CNE 22.1.0**, cluster name is needed in the URI. The URI must be **<cluster\_name>/prometheus/api**. For example: **/bmw/prometheus/api** (For more details on the Prometheus URI, check corresponding CNE version's user guide.)
  - If OSO is being used, use **"/api/"** for OSO-1.6.1 or earlier and use **"/prometheus/api/"** for OSO-1.6.2 or later.
  - For OSO-1.10.0, the URI must be **<cluster\_name>/prometheus/api**. For example: **/bmw/prometheus/api** (For more details on the Prometheus URI, check corresponding OSO version's user guide)
- **o:** Helm\_release\_name used to deploy NRF
- **p:** Host name of MySQL (For example: mysql-connectivity-service)
- **q:** Privileged User secret name (For example: privilegeduser-secret)
- **r:** wait time before sending discovery request (0.5)
- **s:** comma-separated values of replicationStatusUri(s) of the appinfo(s).
- **t:** the cluster domain

- **u**: wait time before sending registration requests for write operations
- **v**: wait time before sending subscription requests
- **w**: wait time before sending accesstoken requests
- **x**: wait time before sending registration requests for read operation

**Note**

- If the **2-Site-Geo** cases are run, the user has to provide values for NRF2 in corresponding environment variables in the above script as per the deployment. A sample configuration is given as below:

```
node ('master'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
FT_ENDPOINT d = GATEWAY_IP e = GATEWAY_PORT
 //f = CONFIG_IP g = CONFIG_PORT h =
STUB_IP i = STUB_PORT j = NFINSTANCEID
 //k = PROMETHEUS_IP l = PROMETHEUS_PORT m =
RERUN_COUNT n = PROMETHEUS_URI o = HELM_RELEASE_NAME
 //p = MYSQL_HOST q =
PRIVILEGED_USER_SECRET_NAME r = DISCOVERY_WAIT_TIME
 //s = REPLICATION_STATUS_URI t =
CLUSTER_DOMAIN u = REGISTER_WRITE_WAIT_TIME
 //v = SUBSCRIPTION_WAIT_TIME w =
ACCESSTOKEN_WAIT_TIME x =
REGISTER_READ_WAIT_TIME
 sh '''
 sh /var/lib/jenkins/ocnrf_tests/preTestConfig.sh \
 -a NRF \
 -b occne-cndbtierone,occne-cndbtiertwo \
 -c ocnrf-
ingressgateway.ocnrf.svc.cluster.local:80,ocnrf-1-
ingressgateway.ocnrf.svc.cluster.local:80 \
 -d ocnrf-ingressgateway.occne-cndbtierone,ocnrf-1-
ingressgateway.occne-cndbtiertwo \
 -e 80,80 \
 -f ocnrf-nrfconfiguration.occne-cndbtierone,ocnrf-1-
nrfconfiguration.occne-cndbtiertwo \
 -g 8080,8080 \
 -h notify-stub-service.occne-cndbtierone,notify-stub-
service02.occne-cndbtierone \
 -i 8080,8080 \
 -j 6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5c,6faf1bbc-6e4a-4454-a507-a14ef8e1bc5d \
 -k oso-prom-svr.oso \
 -l 80 \
 -m 2 \
 -n /prometheus/api/ \
 -o ocnrf,ocnrf-1 \
 -p mysql-connectivity-service \
 -q privilegeduser-secret \

 -r 0.5 \
 -s http://ocnrf-ocnrf-app-info:5906/status/category/
replicationstatus,http://ocnrf-1-ocnrf-app-info:5906/status/
category/replicationstatus \
 -t svc.cluster.local \
 -u 0 \
 -v 0 \
 -w 0 \
 -x 0
```

```

'''
 if(env.Include_NewFeatures && "${Include_NewFeatures}" ==
"YES"){
 sh '''sh /var/lib/jenkins/common_scripts/
merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/
Jenkinsfile-Merged"
 }
 else{
 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/
Jenkinsfile-Regression"
 }
}
}
}

```

- If **3SiteGeo** and georedundancy suite in Regression pipeline cases are run, the user has to provide values for NRF2, NRF3 in corresponding environment variables in the above script as per the deployment. A sample configuration is given as below:

```

node ('master'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
FT_ENDPOINT d = GATEWAY_IP e = GATEWAY_PORT
 //f = CONFIG_IP g = CONFIG_PORT h =
STUB_IP i = STUB_PORT j = NFINSTANCEID
 //k = PROMETHEUS_IP l = PROMETHEUS_PORT m =
RERUN_COUNT n = PROMETHEUS_URI o = HELM_RELEASE_NAME
 //p = MYSQL_HOST q =
PRIVILEGED_USER_SECRET_NAME r = DISCOVERY_WAIT_TIME
 //s = REPLICATION_STATUS_URI t =
CLUSTER_DOMAIN u = REGISTER_WRITE_WAIT_TIME
 //v = SUBSCRIPTION_WAIT_TIME w =
ACCESSTOKEN_WAIT_TIME x =
REGISTER_READ_WAIT_TIME
 sh '''
 sh /var/lib/jenkins/ocnrf_tests/preTestConfig.sh \
 -a NRF \
 -b occne-cndbtierone,occne-cndbtiertwo,occne-
cndbtierthree \
 -c ocnrf-
ingressgateway.ocnrf.svc.cluster.local:80,ocnrf-1-
ingressgateway.ocnrf.svc.cluster.local:80,ocnrf-2-
ingressgateway.ocnrf.svc.cluster.local:80 \
 -d ocnrf-ingressgateway.occne-cndbtierone,ocnrf-1-
ingressgateway.occne-cndbtiertwo,ocnrf-2-ingressgateway.occne-
cndbtierthree \
 -e 80,80,80 \
 -f ocnrf-nrfconfiguration.occne-cndbtierone,ocnrf-1-
nrfconfiguration.occne-cndbtiertwo,ocnrf-2-
nrfconfiguration.occne-cndbtierthree \
 -g 8080,8080,8080 \
 -h notify-stub-service.occne-cndbtierone,notify-stub-
service02.occne-cndbtierone,notify-stub-service03.occne-
cndbtierone \
 -i 8080,8080,8080 \
 '''
}

```

```

 -j 6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5c,6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5d,6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e \
 -k oso-prom-svr.oso \
 -l 80 \
 -m 2 \
 -n /prometheus/api/ \
 -o ocnrf,ocnrf-1,ocnrf-2 \
 -p mysql-connectivity-service \
 -q privilegeduser-secret \

 -r 0.5 \
 -s http://ocnrf-ocnrf-app-info:5906/status/category/
replicationstatus,http://ocnrf-1-ocnrf-app-info:5906/status/
category/replicationstatus,http://ocnrf-2-ocnrf-app-info:5906/
status/category/replicationstatus \
 -t svc.cluster.local \
 -u 0 \
 -v 0 \
 -w 0 \
 -x 0
 ...
 if(env.Include_NewFeatures && "${Include_NewFeatures}" ==
"YES"){
 sh '''sh /var/lib/jenkins/common_scripts/
merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/
Jenkinsfile-Merged"
 }
 else{
 load "/var/lib/jenkins/ocnrf_tests/jenkinsData/
Jenkinsfile-Regression"
 }
}
}
}

```

### Note

If user is using CNE1.6 or OSO 1.6.1 or earlier, change the Prometheus service URI (-n) to "/api/" and Prometheus service name and port to appropriate values. For CNE 1.7.0 or OSO 1.6.2 or later, retain the default value **"/prometheus/api/"**. If OSO 1.10.0 is used, use the URI along with the cluster\_name (for example: **/bmw/prometheus/api**).

3. Click **Save** after making necessary changes. The NRF-NewFeatures screen appears. Click the **Build with Parameters** link available in the left navigation pane of the **NRF-NewFeatures** Pipeline screen.

In the Pipeline screen, there are three **Select\_Option(s)**, which are:

- **All:** This is the default option. It runs all the NRF test cases. Scroll down and click **Build** to execute all the test cases.

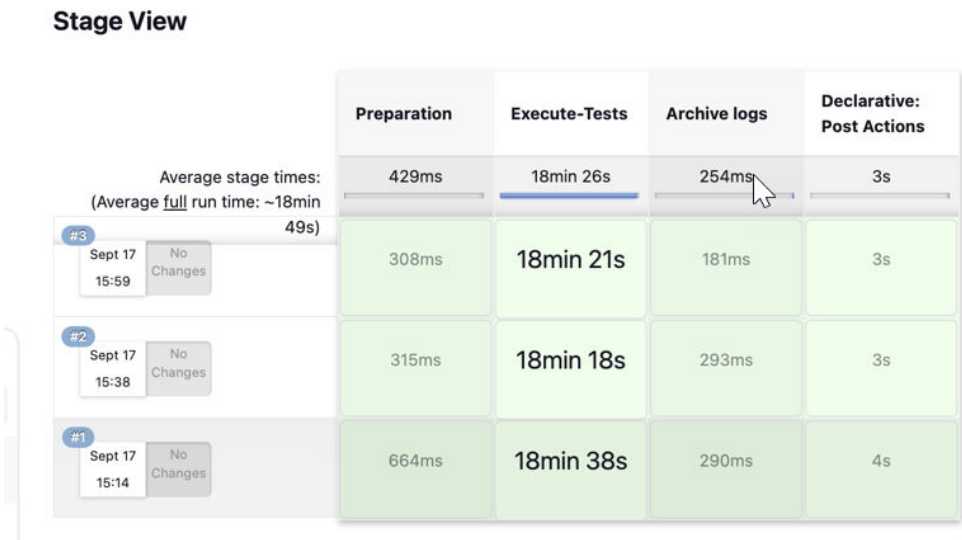
- **Sanity:** It is recommended to run sanity before running any test case. It ensures all the deployments are done properly.
  - **Single/MultipleFeatures:** This option allows you to select any number of test cases that you want to run from the list of total test cases available for execution. After selecting the test cases, scroll down and click **Build** to run the selected NRF test cases.
4. Select one of the following configuration types:
- **Product\_Config:** On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the key-value pair and yaml files defined or present in the "Product Config" folder.
  - **Custom\_Config:** On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the key-value pair and yaml files defined or present in the "Custom Config" folder. To use the [Parameterization](#) feature, always select the Custom\_Config option. User can copy, add, or delete the required test cases that are available for the NRF and place them appropriately within the custom folder for NRF-NewFeatures. Reload the page to view the test cases available in the custom NewFeatures folder. For more information, see [Parameterized approach for SUT custom configuration](#).

Based on the requirement, user can select any of the above option and click on **"Build"**. The NRF testcases are divided into NRF Service operations. The classification is as follows:

- **NRF Sanity** - This feature file contains all the basic sanity cases for NRF ATS to validate the deployment is correct or not. It is advisable for user to run these cases before starting a complete suite.
- **Discovery** - These feature files are listed with a prefix as **"Disc"**. All the discovery microservice related cases will be listed once this option is selected.
- **NRF Functional** - These feature files are listed with a prefix as **"Feat"**. All the functional cases will be listed once this option is selected.
- **Registration** - These feature files are listed with a prefix as **"Upd"**. These are related to update operation of registered profiles. All the registration cases will be listed once this option is selected.
- **Subscription** - These feature files are listed with a prefix as **"Subs"**. All the subscription microservice related cases will be listed once this option is selected.
- **Roaming** - These feature files are listed with a prefix as **"Roaming"**. All the roaming cases will be listed once this option is selected.



Figure 4-30 Sample Screen: NRF-ATS Full Execution



A sample consolidated test report is shown below, when rerun is set to 2:

Figure 4-31 Test Cases Result - Sanity

```
1 feature passed, 0 failed, 0 skipped
11 scenarios passed, 0 failed, 0 skipped
198 steps passed, 0 failed, 0 skipped, 0 undefined
Took 0m55.758s
[Pipeline] sh
+ cd /var/lib/jenkins/ocnrf_tests
++ cat /var/lib/jenkins/ocnrf_tests/environ.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=2
+ sh re-run.sh 2 /var/lib/jenkins/.jenkins/workspace/NRF-NewFeatures
/var/lib/jenkins/ocnrf_tests/rerun_failing.features does not exist
/var/lib/jenkins/ocnrf_tests/rerun_failing.features does not exist
1
Scenario passed
[Pipeline] sh
+ cd /var/lib/jenkins/ocnrf_tests
++ cat /var/lib/jenkins/ocnrf_tests/environ.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=2
+ sh report.sh 2
Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0
```

**Note**

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

**Figure 4-32 Test Cases Result - All-NewFeatures**

```

2024-09-17 10:26:32,329 [32m INFO LOG.AllHooks:223 [0m| [32mInside after_all [0m
2024-09-17 10:26:32,329 [32m INFO LOG.AllHooks:275 [0m| [32mExiting after_all [0m
15 features passed, 0 failed, 0 skipped
68 scenarios passed, 0 failed, 0 skipped
1231 steps passed, 0 failed, 0 skipped, 0 undefined
Took 9m51.111s
[Pipeline] sh
+ cd /var/lib/jenkins/ocnrf_tests
++ cat /var/lib/jenkins/ocnrf_tests/envIRON.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=1
+ sh re-run.sh 1 /var/lib/jenkins/.jenkins/workspace/NRF-NewFeatures
/var/lib/jenkins/ocnrf_tests/rerun_failing.features does not exist
1
Scenario passed

```

### Parameterized approach for SUT custom configuration

Using this feature, user can make the following customizations to custom folders

- Add new test cases by adding datafiles
- Remove test cases
- Modify the parameters and their values in the key-value pair or <feature>.yaml files

To run ATS test cases, user can maintain as many versions of **Custom\_Config** folder by using the following naming convention:

Cust ConfigN

where N can be any number

At the time of execution, ensure to rename the required folder to **Cust Config** folder as Jenkins always retrieves data from this folder when user selects **Custom\_Config**.

### Updating Global Parameters

To use Custom\_Config, it is required to change the value of **cust\_folder** from **data** to **cust\_data** in global.yaml file. In addition, you can customize the parameters and their respective values in the global.yaml as per the requirements.

### Updating Feature Parameters

Consider the following points when customizing <feature>.yaml files for parameterized feature:

- In addition to global.yaml parameters, feature files may also contain parameters for which user can update values at the time of running pipelines.
- Changing the values of parameters tagged as "Feature Specific Value" may cause failures at the time of running pipelines.
- Values for parameters tagged with #START\_GLOBAL and #END\_GLOBAL tags take values from global.yaml.

**Note**

For NRF-ATS release 24.1.3, parameterization is supported for NRF NewFeatures pipeline only.

## 4.3.4 NRF-NewFeatures Documentation

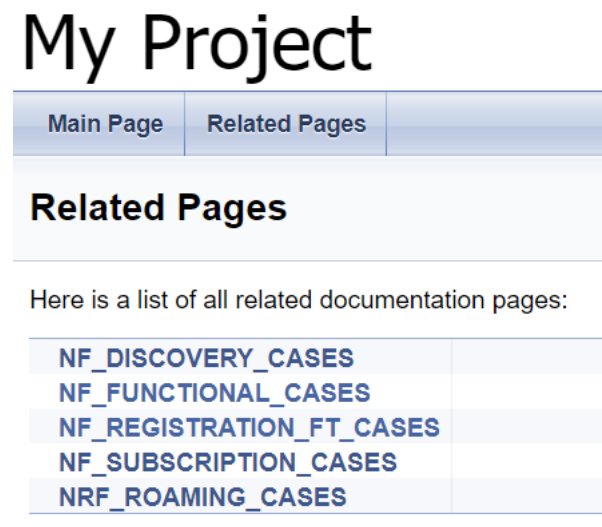
To view NRF test cases, go to **NRF-NewFeatures** pipeline and click the **Documentation** link in the left navigation pane. It lists all the test cases provided as part of NRF ATS -24.1.x along with sanity cases.

The NRF test cases are divided into multiple groups based on the functionality.

- **NF\_DISCOVERY\_CASES** - All the discovery microservice related cases will be listed once this option is selected.
- **NF\_FUNCTIONAL\_CASES** - All the functional cases will be listed once this option is selected.
- **NF\_REGISTRATION\_FT\_CASES** - All the registration cases will be listed once this option is selected.
- **NF\_SUBSCRIPTION\_CASES** - All the subscription microservice related cases will be listed once this option is selected.
- **NF\_ROAMING\_CASES** - All the roaming cases will be listed once this option is selected.

The following screenshot shows all the NRF features:

**Figure 4-33 NRF-NewFeatures Documentation**



Click any functionality to view its test cases and scenarios of each test case as shown below:

Figure 4-34 Sample Feature: NF\_BASIC\_SANITY\_CASES

**NF\_BASIC\_SANITY\_CASES****Sanity.feature**

**Description :** This feature file validates successful scenarios for all the basic operations performed by NRF

**Scenario-1 : Plmn Configuration**

**Objective :** Validate the successful configuration of nrfPlmnList

**Pre-requisite :** NRF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                   | Expected Result                                        |
|-----------------------------------------------------------------------------|--------------------------------------------------------|
| 1.) Send a configuration request to NRF to set nrfPlmnList with valid value | 1.) Configuration is successful with response code 200 |
| 2.) Validate the response body.                                             | 2.) Response body should have the configured plmn.     |

**Scenario-2 : Registration**

**Objective :** Validate the successful registration of an NF with mandatory and conditional parameters

**Pre-requisite :** NRF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                                              | Expected Result                                                         |
|--------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------|
| 1.) Send registration request for an NF to NRF with mandatory and conditional parameters in nrfProfile | 1.) Registration should be successful with response code 201            |
| 2.) Perform a GET operation to fetch the registered NF information                                     | 2.) Verify that GET operation provides the registered profile of the NF |

Based on the functionalities covered under **Documentation**, the **Build Requires Parameters** screen displays test cases. To navigate back to the Pipeline **NRF-NewFeatures** screen, click the **Back to NRF-NewFeatures** link available on top left corner of the screen.

## 4.3.5 NRF-Regression Pipeline

This pre-configured pipeline contains all the test cases delivered till NRF ATS 24.1.x. However, some test cases are updated as per new implementation of NRF.

The configuration method and parameters are same as the **NewFeatures** pipeline.

Only difference in this pipeline is, while executing **2-Site-Geo** and **3-Site-Geo** test cases, user has to provide appropriate values for NRF2 and NRF 3 in the Pipeline script. In this pipeline, user will not get option to run **Sanity**.

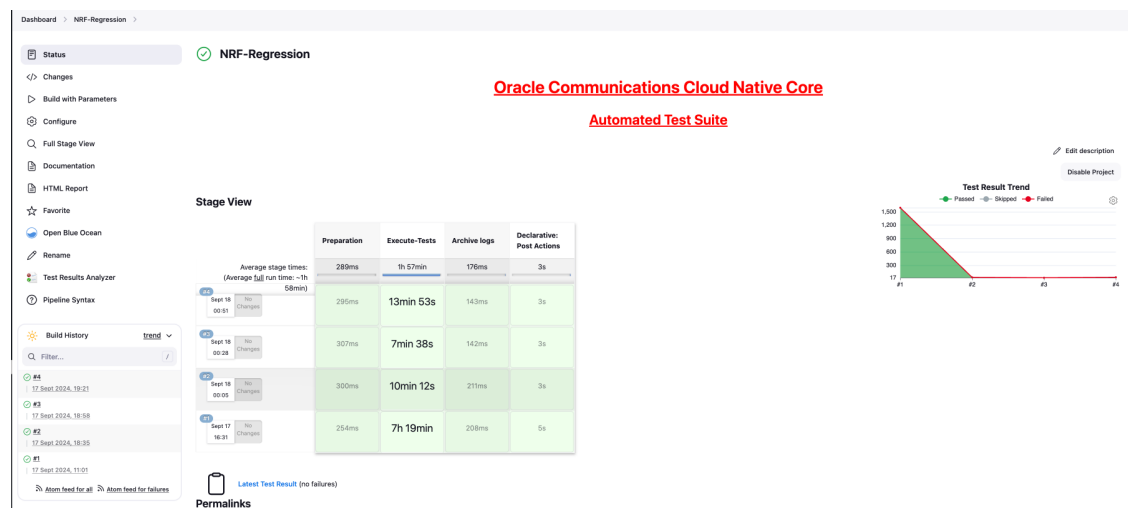
The NRF-Regression test cases are divided into following service operations:

- **2-Site-Geo** - These feature files are listed with a prefix as **"Geo\_Redundancy"**.
- **3-Site-Geo** - These feature files contain all the cases which are executed on 3 Site GEO NRF.
- **CCA-Header** - These feature files contain all cases related to CCA Header feature.
- **NF-FQDN-Authentication** - These feature files are listed with a prefix as **"NfAuthentication"**.
- **AccessToken** - These feature files are listed with a prefix as **"oAuth"** and **"AccessToken"**.
- **Alerts** - These feature files are list with a prefix as **"Alert"**.
- **Configuration** - These feature files are listed with a prefix as **"Config"** and **"SystemOptions"**.
- **Discovery** - These feature files are listed with a prefix as **"Disc"**.
- **NRF Forwarding** - These feature files are listed with a prefix as **"Forwarding"**.
- **NRF Functional** - These feature files are listed with a prefix as **"Feat"**.

- **Key-ID** - These feature files are listed with a prefix as "**Kid**".
- **NRF State Data** - These feature files are listed with a prefix as "**State**".
- **Preferred Locality** - These feature files are listed with a prefix as "**Pref**".
- **Registration** - These feature files are listed with a prefix as "**HBTimerEnhancement**", "**Reg**" and "**Upd**". These are related to update operation of registered profiles.
- **NRF SLF** - These feature files are listed with a prefix as "**SLF**".
- **Subscription** - These feature files are listed with a prefix as "**Subs**".
- **NRF Growth** - These feature files contain all cases related to NRF Growth feature.

The following screenshot shows the NRF-Regression.

**Figure 4-35 NRF-Regression**



The following screenshot shows successful execution as part of ATS image.

**Figure 4-36 NRF-Regression - All-default-Regression**

```

2024-09-17 18:21:14,342 [32m INFO LOG.AllHooks:223 [0m| [32mInside after_all [0m
2024-09-17 18:21:14,342 [32m INFO LOG.AllHooks:275 [0m| [32mExiting after_all [0m
572 features passed, 0 failed, 0 skipped
1589 scenarios passed, 0 failed, 0 skipped
25301 steps passed, 0 failed, 0 skipped, 0 undefined
Took 245m0.812s
[Pipeline] sh
+ cd /var/lib/jenkins/ocnrf_tests
++ cat /var/lib/jenkins/ocnrf_tests/envIRON.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=1
+ sh re-run.sh 1 /var/lib/jenkins/.jenkins/workspace/NRF-Regression
/var/lib/jenkins/ocnrf_tests/rerun_failing.features does not exist
1
Scenario passed

```

## 4.3.6 NRF-Regression Documentation

Click **Documentation** in the left navigation pane of the **NRF-Regression** pipeline to view all the test cases provided till NRF ATS 23.4.x.

The NRF test cases are divided into multiple groups based on following functionalities:

- **NF\_ALERT\_CASES** - Test case for validating alerts will be listed once this option is selected.
- **NF\_CCA\_HEADER\_CASES** - Cases related to CCAHeader feature will be listed once this option is selected.
- **NF\_CHFINFO\_CASES** - All the ChfInfo enhancement cases will be listed once this option is selected.
- **NF\_CONFIGURATION\_CASES** - Cases related to NRF configuration will be listed once this option is selected.
- **NF\_DISCOVERY\_CASES** - All the discovery microservice related cases will be listed once this option is selected.
- **NRF\_FORWARDING\_CASES** - All the forwarding related cases will be listed once this option is selected.
- **NF\_FQDN\_AUTHENTICATION\_CASES** - All the NF FQDN authentication cases will be listed once this option is selected.
- **NF\_FUNCTIONAL\_CASES** - All the functional cases will be listed once this option is selected.
- **NF\_GEO\_REDUNDANCY\_CASES** - All the Geo-Redundancy related cases (2-Site-GEO and 3-Site-Geo both) will be listed once this option is selected.
- **NF\_KEY\_ID\_FEATURE\_CASES** - All the K-ID feature related cases will be listed once this option is selected.

- **NF\_OAUTH\_CASES** - All the accesstoken related cases will be listed once this option is selected.
- **NRF\_PREFERRED\_LOCALITY\_CASES** - All the preferred locality feature related cases will be listed once this option is selected.
- **NF\_REGISTRATION\_FT\_CASES** - All the registration related cases will be listed once this option is selected.
- **NRF\_ROAMING\_CASES** - All the roaming related cases will be listed once this option is selected.
- **NF\_SLF\_FEATURE\_CASES** - All the SLF related cases will be listed once this option is selected.
- **NF\_STATE\_DATA\_CASES** - All the nfStateData related cases will be listed once this option is selected.
- **NF\_SUBSCRIPTION\_CASES** - All subscription related cases will be listed once this option is selected.
- **NF\_NRF\_GROWTH** - All growth related cases will be listed once this option is selected.

Figure 4-37    NRF-Regression Documentation

My Project

|           |               |
|-----------|---------------|
| Main Page | Related Pages |
|-----------|---------------|

Related Pages

Here is a list of all related documentation pages:

|                              |  |
|------------------------------|--|
| NF_ALERT_CASES               |  |
| NF_BASIC_SANITY_CASES        |  |
| NF_CCA_HEADER_CASES          |  |
| NF_CHFINFO_CASES             |  |
| NF_CONFIGURATION_CASES       |  |
| NF_DISCOVERY_CASES           |  |
| NF_FQDN_AUTHENTICATION_CASES |  |
| NF_FUNCTIONAL_CASES          |  |
| NF_GEO_REDUNDANCY_CASES      |  |
| NF_KEY_ID_FEATURE_CASES      |  |
| NF_NRF_GROWTH                |  |
| NF_OAUTH_CASES               |  |
| NF_REGISTRATION_FT_CASES     |  |
| NF_SLF_FEATURE_CASES         |  |
| NF_STATE_DATA_CASES          |  |
| NF_SUBSCRIPTION_CASES        |  |
| NRF_FORWARDING_CASES         |  |
| NRF_PREFERRED_LOCALITY_CASES |  |
| NRF_ROAMING_CASES            |  |



The following sample screen shows documentation of Regression pipeline for NRF ATS.

**Figure 4-38 Sample Screen: NRF-Regression Documentation**

#### NF\_CONFIGURATION\_CASES

##### Config01\_NfCallbackUri.feature

**Description :** This feature file validates the NRF configurations with CALLBACK\_URI

##### Scenario-1 : With correct fqdn

**Objective :** Validate the configuration request for CALLBACK\_URI with correct fqdn

**Pre-requisite :** NRF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                         | Expected Result                                                              |
|-----------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 1.) Send a request to NRF to GET existing configuration                           | 1.) Configuration GET request should be successful                           |
| 2.) Send a request to NRF to set configuration with correct fqdn for callback uri | 2.) Configuration request should be successful with response code 200        |
| 3.) Send a request to NRF to GET configuration                                    | 3.) Configuration GET request should successfully return updated information |

##### Scenario-2 : With valid pattern

**Objective :** Validate the configuration request for CALLBACK\_URI with valid pattern

**Pre-requisite :** NRF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                          | Expected Result                                                              |
|------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| 1.) Send a request to NRF to GET existing configuration                            | 1.) Configuration GET request should be successful                           |
| 2.) Send a request to NRF to set configuration with valid pattern for callback uri | 2.) Configuration request should be successful with response code 200        |
| 3.) Send a request to NRF to GET configuration                                     | 3.) Configuration GET request should successfully return updated information |

## 4.4 Running NSSF Test Cases using ATS

### 4.4.1 Prerequisites

The prerequisites to run NSSF Test Cases using NSSF ATS 24.1.1 are:

- Deploy NSSF 24.1.1 with default helm configurations using helm charts.
- All NSSF microservices must be up and running.
- For NSSF ATS Helm Release should be "ocats"
- For NSSF ATS 24.1.1, deploy one stub server and the service name should be "**amf-stubserver**". It is required to run AMF-subscription Notification functionality test cases.
- For NSSF ATS 24.1.1, deploy one stub server and the service name should be "**nrf-stubserver**". It is required to run NRF-subscription Notification functionality test cases.

### 4.4.2 Logging into ATS

Before logging into ATS, [deploy ATS](#) using HELM charts as shown below:

Verify ATS deployment



```
[opc@ocnssf-oci-phx-einstein-bastion-01]$ helm status ocats
NAME: ocats
LAST DEPLOYED: Tue Mar 5 02:22:08 2024
NAMESPACE: nssf-release
STATUS: deployed
REVISION: 1
TEST SUITE: None
```

There are two ways to log in to ATS GUI.

- When an external load balancer (metalLB in case of OCCNE) is available and you provide an external IP to the ATS service, the user can log in to ATS GUI using <External-IP>:8080.
- When you do not provide an external IP to the ATS service, open the browser and enter the external IP of the worker node and nodeport of the ATS service to log in to the ATS GUI.  
<Worker-Node-IP>:<Node-Port-of-ATS>

**Note**

In the **Verifying ATS Deployment** screenshot, the ATS nodeport is highlighted in red as 32013. For more details on ATS deployment, refer to **NSSF ATS Installation Procedure**.

Open a browser and enter the IP Address and port details as <Worker-Node-IP>:<NodePort-of-ATS> (In the above example, the Worker-Node-IP and NodePort-of-ATS are 10.98.101.177:32013, which are shown as highlighted in the screenshot above).

The ATS login screen appears.

Oracle Communications Cloud Native Core - Automated Test Suite

Welcome to ATS!

Username

Password

☐ Keep me signed in

Sign in

To run ATS:

1. Enter the username as 'nssfuser' and password as 'nssfpaswd'. Click **Sign in**. A page with preconfigured pipelines appears.

**Note**

To modify the default login password, refer to [Modifying Login Password](#).

A page with preconfigured pipelines appears.

**Oracle Communications Cloud Native Core**

**Automated Test Suite**

**CNCATS NSSF : 24.1.1**

| S | W | Name ↓           | Last Success   | Last Failure | Last Duration | F   |
|---|---|------------------|----------------|--------------|---------------|-----|
| ☑ | ☀ | NSSF-HealthCheck | N/A            | N/A          | N/A           | ▶ ☆ |
| ☑ | ☀ | NSSF-NewFeatures | 2 hr 27 min #1 | N/A          | 6 min 37 sec  | ▶ ☆ |
| ☑ | ☀ | NSSF-Performance | N/A            | N/A          | N/A           | ▶ ☆ |
| ☑ | ☀ | NSSF-Regression  | 2 hr 17 min #1 | N/A          | 49 min        | ▶ ☆ |

Icon: S M L

- **NSSF-Healthcheck:** This pipeline does a test probe with SUT- trigger helm test and provides the results in Console Logs.
- **NSSF-New-Features:** This pipeline has all the test cases delivered as part of NSSF ATS - 24.1.1.
- **NSSF-Performance:** This pipeline is not operational as of now. It is reserved for future releases of ATS.
- **NSSF-Regression:** This pipeline has the test cases of all the previous releases.

### 4.4.3 NSSF-NewFeatures Pipeline

In this pipeline, you can configure ATS, which is a one-time activity as per System Under Test (SUT) deployment. You can also run all the new NSSF test cases using the pipeline. To configure its parameters:

1. Click **NSSF-NewFeatures** in the **Name** column. The following screen appears:

**Oracle Communications Cloud Native Core**  
**Automated Test Suite**

**Stage View**

**Build History**

| Build | Time                 | Status  |
|-------|----------------------|---------|
| #1    | 05-Mar-2024, 2:50 am | Success |

**Test Result Trend**

Legend: ● Passed ● Skipped ● Failed

| Stage                     | Time     |
|---------------------------|----------|
| Preparation               | 2s       |
| Execute-Tests             | 33ms     |
| 1st Stage                 | 59ms     |
| 1st Stage 1st Group       | 7min 33s |
| Logging / Rerun           | 88ms     |
| Archive logs              | 93ms     |
| Consolidated Output       | 648ms    |
| Declarative: Post Actions | 3s       |

**Permalinks**

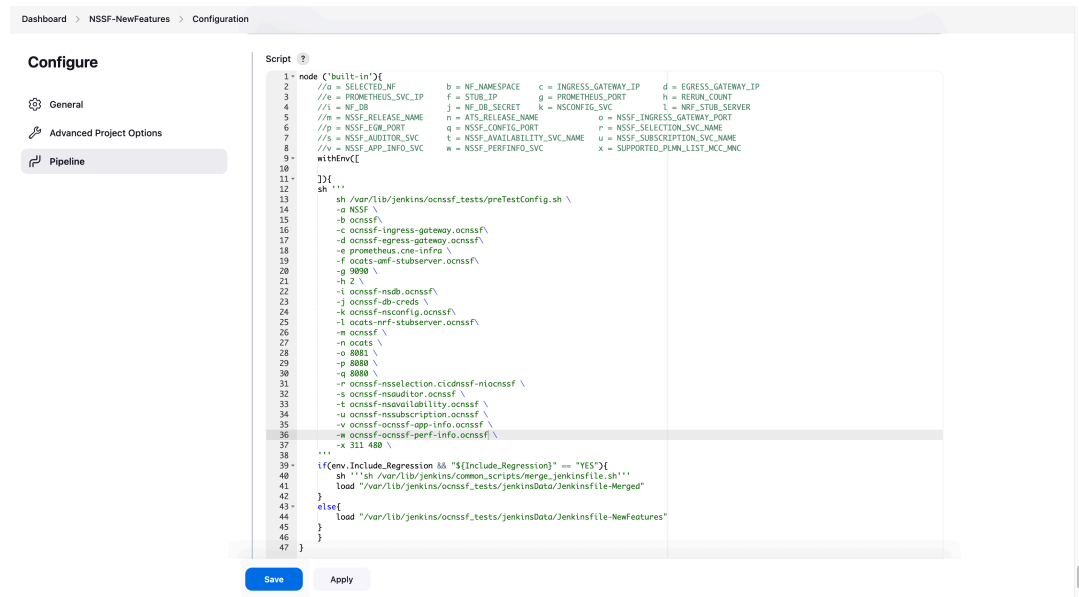
- Last build (#1), 5 hr 10 min ago
- Last stable build (#1), 5 hr 10 min ago
- Last successful build (#1), 5 hr 10 min ago
- Last completed build (#1), 5 hr 10 min ago

In the above screen:

- Click **Configure** to access the configuration screen.
  - Click **Documentation** to view the documented test cases.
  - Click blue dots inside the **Build History** box to view the success console logs of the "All" and "Sanity" respectively.
  - The **Stage View** represents the already run pipeline for customer reference.
2. Click **Configure**. Users MUST wait for the page to load completely. Once the page loads completely, click the **Pipeline** tab to reach the Pipeline configuration as shown below:

### Warning

Make sure that the screen shown above loads completely before you perform any action on it. Also, do not modify any configuration other than that discussed below.



3. You can modify script pipeline parameters from "a" to "n" on the basis of your deployment environment and click Save. The content of the pipeline script is as follows:

```
node ('built-in'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
 INGRESS_GATEWAY_IP d = EGRESS_GATEWAY_IP
 //e = PROMETHEUS_SVC_IP f = STUB_IP g =
 PROMETHEUS_PORT h = RERUN_COUNT
 //i = NF_DB j = NF_DB_SECRET k =
 NSCONFIG_SVC l = NRF_STUB_SERVER
 //m = NSSF_RELEASE_NAME n = ATS_RELEASE_NAME o =
 NSSF_INGRESS_GATEWAY_PORT
 //p = NSSF_EGW_PORT q = NSSF_CONFIG_PORT r =
 NSSF_SELECTION_SVC_NAME
 //s = NSSF_AUDITOR_SVC t = NSSF_AVAILABILITY_SVC_NAME u =
 NSSF_SUBSCRIPTION_SVC_NAME
 //v = NSSF_APP_INFO_SVC w = NSSF_PERFINFO_SVC x =
 SUPPORTED_PLMN_LIST_MCC_MNC
 withEnv([

]){
 sh '''
 sh /var/lib/jenkins/ocnssf_tests/preTestConfig.sh \
 -a NSSF \
 -b ocnssf\
 -c ocnssf-ingress-gateway.ocnssf\
 -d ocnssf-egress-gateway.ocnssf\
 -e prometheus.cne-infra \
 -f ocats-amf-stubserver.ocnssf\
 -g 9090 \
 -h 2 \
 -i ocnssf-nsdb.ocnssf\
 -j ocnssf-db-creds \
 -k ocnssf-nsconfig.ocnssf\
 -l ocats-nrf-stubserver.ocnssf\
 -m ocnssf \
```

```

 -n ocats \
 -o 8081 \
 -p 8080 \
 -q 8080 \
 -r ocnssf-nsselection.ocnssf \
 -s ocnssf-nsauditor.ocnssf \
 -t ocnssf-nsavailability.ocnssf \
 -u ocnssf-nssubscription.ocnssf \
 -v ocnssf-ocnssf-app-info.ocnssf \
 -w ocnssf-ocnssf-perf-info.ocnssf \
 -x 311 480 \
 ...
 if(env.Include_Regression && "${Include_Regression}" == "YES"){
 sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-Merged"
 }
 else{
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-
NewFeatures"
 }
}
}

```

**Note**

The User MUST NOT change any other value apart from these parameters.

The description of these parameters is as follows:

- a: Name of the NF to be tested in capital (NSSF).
- b: Namespace in which the NSSF is deployed (default is ocnssf)
- c: Ingress Gateway IP address (default is ocnssf-ingress-gateway.ocnssf)
- d: Egress Gateway IP address (default is ocnssf-egress-gateway.ocnssf)
- e: Prometheus service IP address (default is prometheus.cne-infra)
- f: Stub service IP address (default is ocats-amf-stubserver.ocnssf)
- g: Port of Prometheus service (default is 80)
- h: Number of times the re-run of failed case is allowed (default is 2).
- i: Database name (default is ocnssf-nsdb.ocnssf)
- j Database secrets (default ocnssf-db-creds)
- k: NSSF config ip address (ocnssf-nsconfig.ocnssf)
- l: NRF stub server IP address (ocats-nrf-stubserver.ocnssf)
- m: NSSF release name (ocnssf)
- n: ATS release name (ocats)
- o: NSSF Ingress Gateway Port
- p: NSSF Egress Gateway Port
- q: NSSF Config Port

- r: NSSF Selection Service Name
- s: NSSF Auditor Service Name
- t: NSSF Availability Service Name
- u: NSSF Subscription Service Name
- v: APP Info Service Name
- w: Perf info Service Name
- x: NSSF Supported PLMN

**Note**

- Do not change any value if the OCCNE cluster is used and NSSF, ATS, and STUB are deployed in the ocnssf namespace.
- In the above image NSSF Helm release name is "ocnssf", ATS Helm Release Name is "ocats" and namespace is "ocnssf". If any change in helm release name of NSSF, ATS and namespace needs to be updated accordingly. For example, NSSF Helm release name is "ocnssfats" and ATS Helm release name is "ocatsnssf" and namespace is "ocnssf2410", then above Pipeline Configuration should be edited as shown below.
- NSSF Helm release name should be updated in "c d i k m r s t u v w" if any change in NSSF helm release name apart from "ocnssf".
- ATS Helm release name Should be updated in "f l n", if any change in ATS Helm release name apart from "ocats".

```
node ('built-in'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
 INGRESS_GATEWAY_IP d = EGRESS_GATEWAY_IP
 //e = PROMETHEUS_SVC_IP f = STUB_IP g =
 PROMETHEUS_PORT h = RERUN_COUNT
 //i = NF_DB j = NF_DB_SECRET k =
 NSCONFIG_SVC l = NRF_STUB_SERVER
 //m = NSSF_RELEASE_NAME n = ATS_RELEASE_NAME
 o = NSSF_INGRESS_GATEWAY_PORT
 //p = NSSF_EGW_PORT q = NSSF_CONFIG_PORT
 r = NSSF_SELECTION_SVC_NAME
 //s = NSSF_AUDITOR_SVC t = NSSF_AVAILABILITY_SVC_NAME
 u = NSSF_SUBSCRIPTION_SVC_NAME
 //v = NSSF_APP_INFO_SVC w = NSSF_PERFINFO_SVC
 x = SUPPORTED_PLMN_LIST_MCC_MNC
 withEnv([

]){
 sh '''
 sh /var/lib/jenkins/ocnssf_tests/preTestConfig.sh \
 -a NSSF \
 -b ocnssf2410\
 -c ocnssfats-ingress-gateway.ocnssf2410\
 -d ocnssfats-egress-gateway.ocnssf2410\
 -e prometheus.cne-infra \
 -f ocatsnssf-amf-stubserver.ocnssf2410\
 -g 9090 \
 -h 2 \
 -i ocnssfats-nsdb.ocnssf\
 -j ocnssf-db-creds \
 -k ocnssfats-nsconfig.ocnssf2410\
 -l ocatsnssf-nrf-stubserver.ocnssf2410\
 -m ocnssfats \
 -n ocatsnssf \
 -o 8081 \
 -p 8080 \
 -q 8080 \
 -r ocnssfats-nsselection.ocnssf2410 \
 -s ocnssfats-nsauditor.ocnssf2410\
```

```

-t ocnssfats-nsavailability.ocnssf2410\
-u ocnssfats-nssubscription.ocnssf2410\
-v ocnssfats-ocnssf-app-info.ocnssf2410\
-w ocnssfats-ocnssf-perf-info.ocnssf2410\
-x 311 480 \

...
if(env.Include_Regression && "${Include_Regression}" ==
"YES"){
 sh '''sh /var/lib/jenkins/common_scripts/
merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/
Jenkinsfile-Merged"
}
else{
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/
Jenkinsfile-NewFeatures"
}
}
}

```

- Click **Save** after making necessary changes. The **Pipeline NSSF-NewFeatures** screen appears.

## Running NSSF New Features Test Cases

To run NSSF New Features test cases:

- Click on **Build with Parameters** present on the NSSF-NewFeatures screen in the extreme left column corresponding to NSSF-NewFeatures row as shown below:

The screenshot displays the Oracle Cloud Native Core Automated Test Suite (ATS) interface. The top navigation bar includes the Oracle logo, a search bar, and user information (nssfuser, log out). The main header shows the path: Dashboard > NSSF-NewFeatures > NSSF-NewFeatures.

On the left, a sidebar lists various actions: Status (selected), Changes, Build with Parameters, Configure, Full Stage View, Documentation, HTML Report, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax.

The main content area is titled "NSSF-NewFeatures" and includes a "Test Result Trend" graph showing a single data point for build #1. Below the graph, a "Stage View" table provides a detailed breakdown of the build stages and their durations.

| Preparation | Execute-Tests | 1st Stage | 1st Stage 1st Group | Logging / Rerun | Archive logs | Consolidated Output | Declarative: Post Actions |
|-------------|---------------|-----------|---------------------|-----------------|--------------|---------------------|---------------------------|
| 2s          | 33ms          | 69ms      | 7min 33s            | 88ms            | 93ms         | 648ms               | 3s                        |

Below the stage view, a "Latest Test Result" section indicates "no failures" for build #1. A "Permalink" section lists recent builds with their status and timestamps.

- Last build (#1), 5 hr 10 min ago
- Last stable build (#1), 5 hr 10 min ago
- Last successful build (#1), 5 hr 10 min ago
- Last completed build (#1), 5 hr 10 min ago

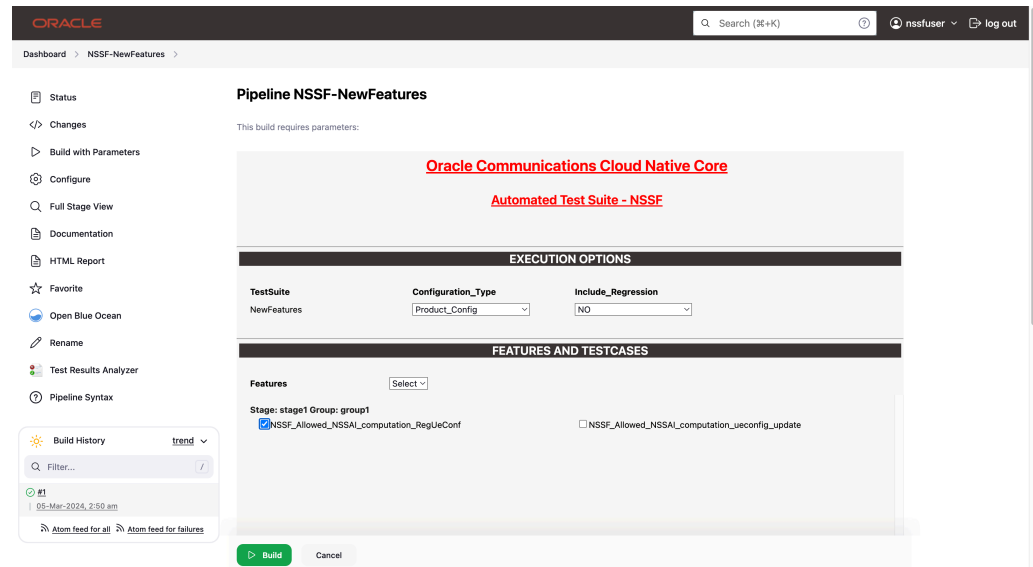
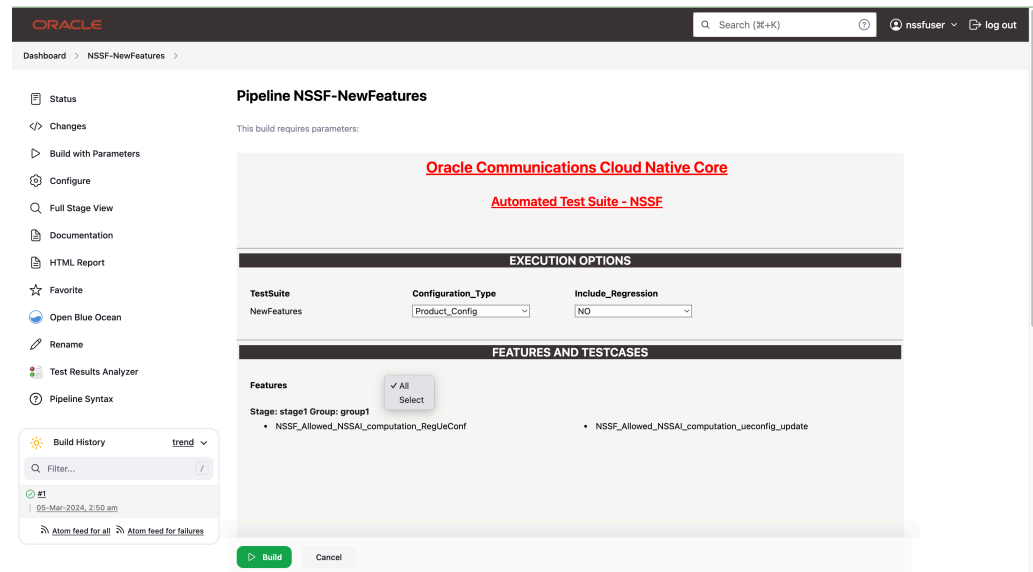


## 2. The following screen appears:

The screenshot displays the Oracle Communications Cloud Native Core Automated Test Suite - NSSF Pipeline configuration interface. The top navigation bar includes the Oracle logo, a search bar, and user information (nssfuser, log out). The left sidebar contains a list of navigation items: Status, Changes, Build with Parameters, Configure, Full Stage View, Documentation, HTML Report, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area is titled 'Pipeline NSSF-NewFeatures' and includes a 'This build requires parameters:' section. Below this, there are three main sections: 'EXECUTION OPTIONS', 'FEATURES AND TESTCASES', and 'TestCases'. The 'EXECUTION OPTIONS' section shows 'TestSuite' as 'NewFeatures', 'Configuration\_Type' as 'Product\_Config', and 'Include\_Regression' as 'NO'. The 'FEATURES AND TESTCASES' section shows 'Features' as 'All' and 'TestCases' as 'All'. The 'TestCases' section lists 39 test cases, including 'NSSF\_Allowed\_NSSAI\_computation\_RegUeConf' and 'NSSF\_Allowed\_NSSAI\_computation\_ueconfig\_update'. At the bottom, there are 'Build' and 'Cancel' buttons.

In the above screen, there are three **Select\_Option(s)**, which are:

- By default, features 'ALL', configuration type 'Product\_config', include regression 'NO' will be selected, and all test cases will be executed once clicked on BUILD, as shown above.
- Features: ALL. Once you click on ALL, a dropdown 'Select' option will appear. After selecting 'Select,' options will appear to choose feature files. To run the selected feature files, click on 'Build,' as shown below.



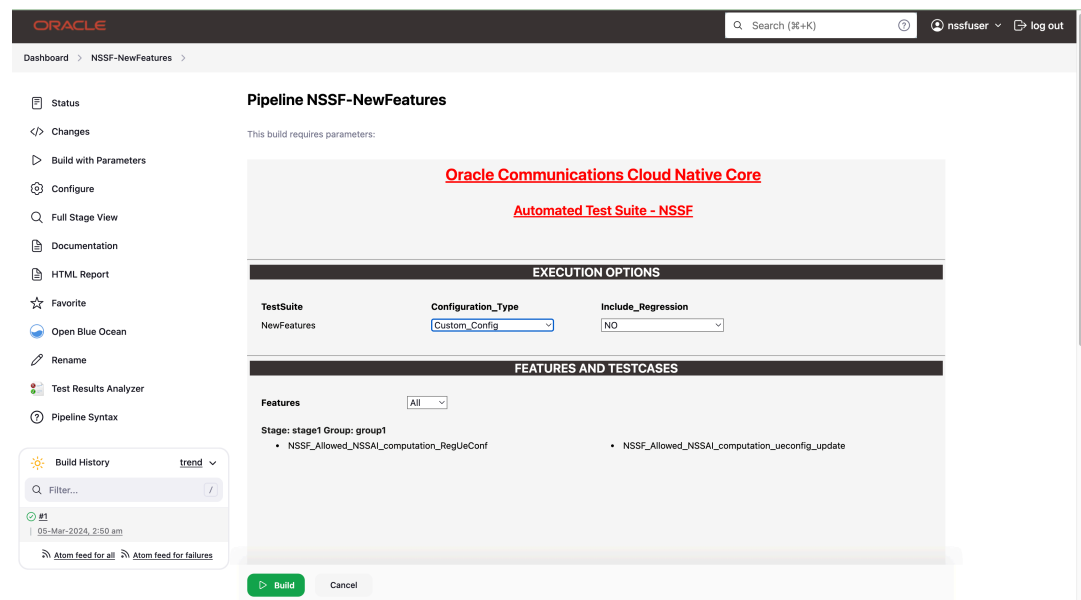
### 3. Select one of the following configuration types:

- **Product\_Config:** On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the keyvalue pair and yaml files defined or present in the "Product Config" folder.
- **Custom\_Config:** On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the keyvalue pair and yaml files defined or present in the "Custom Config" folder. To use the [Parameterization](#) feature, always select the Custom\_Config option. User can copy, add, or delete the required test cases that are available for the NSSF and place them appropriately within the custom folder for NSSF-NewFeatures. Reload the page to view the test cases available in the custom NewFeatures folder. For more information, see Parameterized approach for SUT custom configuration.

The NSSF test cases are divided into NSSF Service operations as follows:

- `NSSF_Allowed_NSSAI_computation_RegUeConf`- This feature file contains test cases related to NSSF Allowed SNSSAI Computation based on enabling and disabling of System Options PLMN based.
- `NSSF_Allowed_NSSAI_computation_ueconfig_update` - This feature file contains test cases related to NSSF Allowed SNSSAI Computation based on enabling and disabling of System Options PLMN based.

If you want to run `custom_config`, select below parameter as `custom_config`, where the framework automatically points out to `custom_config` and `cust_data`.



```
2 features passed, 0 failed, 0 skipped
39 scenarios passed, 0 failed, 0 skipped
586 steps passed, 0 failed, 0 skipped, 0 undefined
Took 7m27.291s
```

### Parameterized approach for SUT custom configuration

Using this feature, user can make the following customizations to custom folders:

- Add new test cases by adding datafiles
- Remove test cases
- Modify the parameters and their values in the key-value pair or `<feature>.yaml` files

To run ATS test cases, user can maintain as many versions of `Custom_Config` folder by using the following naming convention:

`Cust ConfigN`

where N can be any number

At the time of execution, ensure to rename the required folder to `Custom Config` folder as Jenkins always retrieves data from this folder when user selects `Custom_Config`.

## Updating Global Parameters

To use Custom\_Config, it is required to change the value of cust\_folder from data to cust\_data in global.yaml file. In addition, you can customize the parameters and their respective values in the global.yaml as per the requirements.

## Updating Feature Parameters

Consider the following points when customizing <feature>.yaml files for parameterized feature:

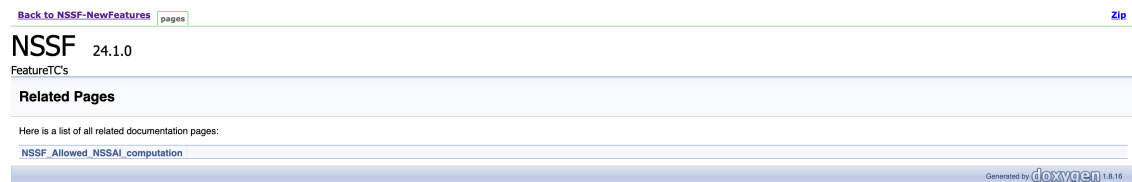
- In addition to global.yaml parameters, feature files may also contain parameters for which user can update values at the time of running pipelines.
- Changing the values of parameters tagged as "Feature Specific Value" may cause failures at the time of running pipelines.
- Values for parameters tagged with #START\_GLOBAL and #END\_GLOBAL tags take values from global.yaml.

### Note

For NSSF-ATS release 24.1.1, parameterization is supported for NSSF NewFeatures pipeline only.

## 4.4.4 NSSF-NewFeatures Documentation

To view NSSF functionalities, go to **NSSF-NewFeatures** pipeline latest build and click the **Documentation** link in the left navigation pane. The following screen appears. Click any functionality to view its test cases and scenarios of each test case as shown below:



A sample of a few documentation features are as follows:

[Back to NSSF-NewFeatures](#) [pages](#) [Zip](#)

# NSSF

## NSSF\_Allowed\_NSSAI\_computation

### NSSF\_Allowed\_NSSAI\_computation\_RegUeConf.feature

Description : This feature file is about NSSF Allowed SNSAI Computation Based on enabling and disabling of System Options PLMN based.

**Scenario-1 : When EnableEnhancedAllowedNSSAIComputation is set to true,Validation that PUT request is successfully enabled the feature and Validate using GET request**

Objective : When EnableEnhancedAllowedNSSAIComputation is set to true,Validation that PUT request is successfully enabled the feature and Validate using GET request

Pre-requisite : NSSF is already deployed with latest images and ATS client is up and running.

| Procedure                                                 | Expected Result                              |
|-----------------------------------------------------------|----------------------------------------------|
| 1.) Send PUT System options configuration request to nssf | 1.) Verify successful response 200           |
| 2.) Send GET System options configuration request to nssf | 2.) Verify successful response 200 with body |

**Scenario-2 : When EnableEnhancedAllowedNSSAIComputation is set to wrong plmn,Validation that PUT request is getting failed and validate that GET request als getting failed**

Objective : When EnableEnhancedAllowedNSSAIComputation is set to wrong plmn,Validation that PUT request is getting failed and validate that GET request als getting failed

Pre-requisite : NSSF is already deployed with latest images and ATS client is up and running. | Procedure | Expected Result || :----- | :----- | || 1.) Send PUT System options configuration request to nssf | 1.) Verify response code with 400 bad request ||

**Scenario-3 : When EnableEnhancedAllowedNSSAIComputation is set to false,Validation that PUT request is successfully disabled the feature and Validate using GET request**

Objective : When EnableEnhancedAllowedNSSAIComputation is set to false,Validation that PUT request is successfully disabled the feature and Validate using GET request

Pre-requisite : NSSF is already deployed with latest images and ATS client is up and running.

| Procedure                                                 | Expected Result                              |
|-----------------------------------------------------------|----------------------------------------------|
| 1.) Send PUT System options configuration request to nssf | 1.) Verify successful response 200           |
| 2.) Send GET System options configuration request to nssf | 2.) Verify successful response 200 with body |

**Scenario-4 : When "S0" is set to true, validate that NSSF\_NSSelection receives a reg request with R-Snssai and S-Snssai being identical, and an accurate computed A-Snssai.**

## 4.4.5 NSSF-Regression Pipeline

- This pipeline contains test cases from the previous versions.
- Some of the test cases are updated as per the new implementation of NSSF.
- The configuration method and parameters are the same as the **NewFeatures** pipeline.
- The NSSF test cases are divided into NSSF Service operations. The classification is as follows:
- **Availability Update:** These feature files are listed with a prefix as "Update".
  - **Configuration:** These feature files are listed with a prefix as "failure".
  - **Registration:** These feature files are listed with a prefix as "NsSelection\_Registration".
  - **PDU Session:** These feature files are listed with a prefix as "NsSelection\_PDU".
  - **SuportedFeatures:** This feature file has Feature Negotiation test cases.
  - **subscription\_patch:** This feature file has SUBMOD test cases.
  - **eanan:** This feature file has EANAN test cases.
  - **TAI Range:** This feature file has TAI Range supported NsAvailability and NsSelection test cases.
  - **NSSF\_Sending\_Notification\_via\_ARS\_SCP:** This feature file has test cases related to NRF discovery and subscription.
  - **NSSF Sanity:** This feature file contains all the basic sanity cases for NSSF ATS 24.1.1.

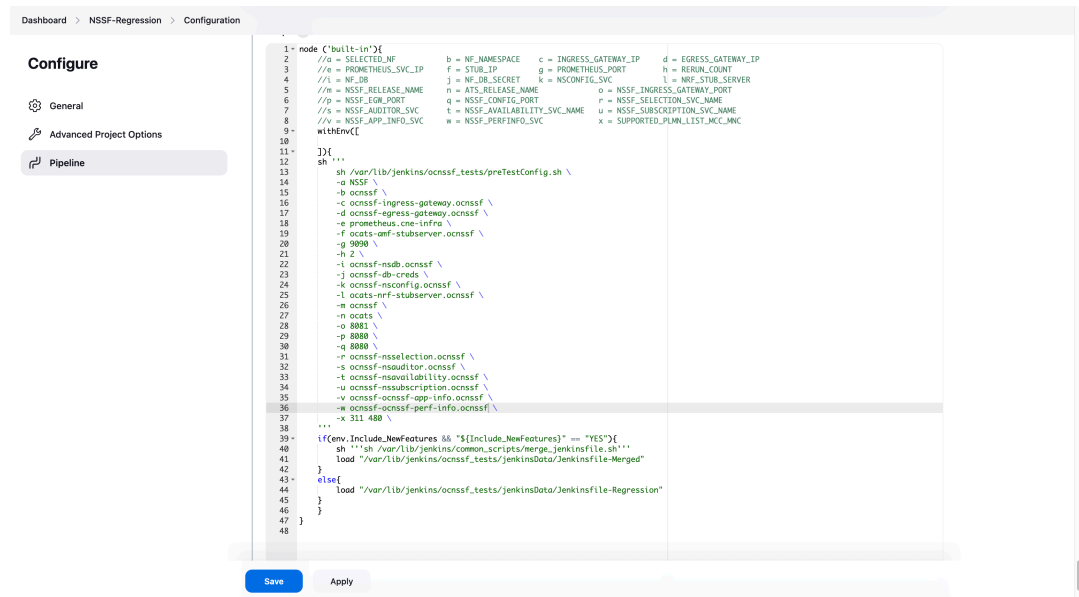
The screenshot displays the Oracle NSSF-Regression dashboard. The top navigation bar includes the Oracle logo, a search bar, and user information (nssfuser). The main content area is titled "NSSF-Regression" and features a sidebar with navigation options: Status, Changes, Build with Parameters, Configure, Full Stage View, Documentation, HTML Report, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main view is titled "Oracle Communications Cloud Native Core Automated Test Suite" and includes a "Test Result Trend" chart showing a green line for "Passed" and a red line for "Failed". Below the chart is a "Stage View" table showing the execution of the test suite across four stages: Preparation, Execute-Tests, 1st Stage, and 1st Stage 1st Group. The table includes columns for stage duration, test results, and declarative post actions. A "Build History" panel on the left shows a list of builds with their status and timestamps. A "Latest Test Result" section at the bottom indicates that the latest test result has no failures.

| Preparation | Execute-Tests | 1st Stage | 1st Stage 1st Group | Logging / Rerun | Archive logs | Consolidated Output | Declarative: Post Actions |
|-------------|---------------|-----------|---------------------|-----------------|--------------|---------------------|---------------------------|
| 2s          | 22ms          | 37ms      | 53min 8s            | 78ms            | 78ms         | 624ms               | 3s                        |
| 2s          | 21ms          | 35ms      | 52min 57s           | 76ms            | 78ms         | 615ms               | 3s                        |
| 2s          | 22ms          | 42ms      | 53min 8s            | 76ms            | 77ms         | 619ms               | 3s                        |
| 2s          | 22ms          | 36ms      | 53min 6s            | 79ms            | 78ms         | 630ms               | 3s                        |
| 2s          | 25ms          | 37ms      | 53min 20s           | 84ms            | 81ms         | 633ms               | 3s                        |

- In the above screen:
  - Click **Configure** to access the configuration screen.
  - Click **Documentation** to view the documented test cases.
  - Click blue dots inside the **Build History** box to view the success console logs of the "All" and "Sanity" respectively.
  - The **Stage View** represents the already run pipeline for customer reference.
- Click **Configure**. Users MUST wait for the page to load completely. Once the page loads completely, click the **Pipeline** tab to reach the Pipeline configuration as shown below:

### Warning

Make sure that the screen shown above loads completely before you perform any action on it. Also, do not modify any configuration other than that discussed below.



3. You can modify script pipeline parameters from "a" to "m" on the basis of your deployment environment and click Save. The content of the pipeline script is as follows:

```
node ('built-in'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
 INGRESS_GATEWAY_IP d = EGRESS_GATEWAY_IP
 //e = PROMETHEUS_SVC_IP f = STUB_IP g =
 PROMETHEUS_PORT h = RERUN_COUNT
 //i = NF_DB j = NF_DB_SECRET k =
 NSCONFIG_SVC l = NRF_STUB_SERVER
 //m = NSSF_RELEASE_NAME n = ATS_RELEASE_NAME o =
 NSSF_INGRESS_GATEWAY_PORT
 //p = NSSF_EGW_PORT q = NSSF_CONFIG_PORT r =
 NSSF_SELECTION_SVC_NAME
 //s = NSSF_AUDITOR_SVC t = NSSF_AVAILABILITY_SVC_NAME u =
 NSSF_SUBSCRIPTION_SVC_NAME
 //v = NSSF_APP_INFO_SVC w = NSSF_PERFINFO_SVC x =
 SUPPORTED_PLMN_LIST_MCC_MNC
 withEnv([

]){
 sh '''
 sh /var/lib/jenkins/ocnssf_tests/preTestConfig.sh \
 -a NSSF \
 -b ocnssf \
 -c ocnssf-ingress-gateway.ocnssf \
 -d ocnssf-egress-gateway.ocnssf \
 -e prometheus.cne-infra \
 -f ocats-amf-stubserver.ocnssf \
 -g 9090 \
 -h 2 \
 -i ocnssf-nsdb.ocnssf \
 -j ocnssf-db-creds \
 -k ocnssf-nsconfig.ocnssf \
 -l ocats-nrf-stubserver.ocnssf \
```

```

 -m ocnssf \
 -n ocats \
 -o 8081 \
 -p 8080 \
 -q 8080 \
 -r ocnssf-nsselection.ocnssf \
 -s ocnssf-nsauditor.ocnssf \
 -t ocnssf-nsavailability.ocnssf \
 -u ocnssf-nssubscription.ocnssf \
 -v ocnssf-ocnssf-app-info.ocnssf \
 -w ocnssf-ocnssf-perf-info.ocnssf \
 -x 311 480 \
 ...
 if(env.Include_NewFeatures && "${Include_NewFeatures}" == "YES"){
 sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-Merged"
 }
 else{
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-
Regression"
 }
}
}

```

#### Note

The User MUST NOT change any other value apart from these parameters.

The description of these parameters is as follows:

- a: Name of the NF to be tested in capital (NSSF).
- b: Namespace in which the NSSF is deployed (default is ocnssf)
- c: Ingress Gateway IP address (default is ocnssf-ingress-gateway.ocnssf)
- d: Egress Gateway IP address (default is ocnssf-egress-gateway.ocnssf)
- e: Prometheus service IP address (default is prometheus.cne-infra)
- f: Stub service IP address (default is ocats-amf-stubserver.ocnssf)
- g: Port of Prometheus service (default is 80)
- h: Number of times the re-run of failed case is allowed (default is 2).
- i: Database name (default is ocnssf-nsdb.ocnssf)
- j Database secrets (default ocnssf-db-creds)
- k: NSSF config ip address (ocnssf-nsconfig.ocnssf)
- l: NRF stub server IP address (ocats-nrf-stubserver.ocnssf)
- m: NSSF release name (ocnssf)
- n: ATS release name (ocats)
- o: NSSF Ingress Gateway Port
- p: NSSF Egress Gateway Port



- q: NSSF Config Port
- r: NSSF Selection Service Name
- s: NSSF Auditor Service Name
- t: NSSF Availability Service Name
- u: NSSF Subscription Service Name
- v: APP Info Service Name
- w: Perf info Service Name
- x: NSSF Supported PLMN

**Note**

- Do not change any value if the OCCNE cluster is used, and NSSF, ATS, and STUB are deployed in the ocnssf namespace.
- In the above image, the NSSF Helm Release Name is "ocnssf", ATS Helm Release Name is "ocats", and Namespace is "ocnssf". If any change in the helm release name of NSSF, ATS, and namespace needs to be updated accordingly.
- For example, if the NSSF Helm Release Name is "ocnssfats" and ATS Helm Release Name is "ocatsnssf", and Namespace is "ocnssf2410", then the above Pipeline Configuration should be edited as shown below:
  - NSSF Helm Release Name should be updated in "c d i k m r s t u v w" if any change in NSSF helm release name apart from "ocnssf".
  - ATS Helm Release Name should be updated in "f l n" if any change in ATS Helm Release Name apart from "ats".

```
node ('built-in'){
 //a = SELECTED_NF b = NF_NAMESPACE c =
INGRESS_GATEWAY_IP d = EGRESS_GATEWAY_IP
 //e = PROMETHEUS_SVC_IP f = STUB_IP g =
PROMETHEUS_PORT h = RERUN_COUNT
 //i = NF_DB j = NF_DB_SECRET k =
NSCONFIG_SVC l = NRF_STUB_SERVER
 //m = NSSF_RELEASE_NAME n = ATS_RELEASE_NAME o
= NSSF_INGRESS_GATEWAY_PORT
 //p = NSSF_EGW_PORT q = NSSF_CONFIG_PORT r
= NSSF_SELECTION_SVC_NAME
 //s = NSSF_AUDITOR_SVC t = NSSF_AVAILABILITY_SVC_NAME u
= NSSF_SUBSCRIPTION_SVC_NAME
 //v = NSSF_APP_INFO_SVC w = NSSF_PERFINFO_SVC x
= SUPPORTED_PLMN_LIST_MCC_MNC
 withEnv([

]){
 sh '''
 sh /var/lib/jenkins/ocnssf_tests/preTestConfig.sh \
 -a NSSF \
 -b ocnssf2410\
 -c ocnssfats-ingress-gateway.ocnssf2410\
 -d ocnssfats-egress-gateway.ocnssf2410\
 -e prometheus.cne-infra \
 -f ocatsnssf-amf-stubserver.ocnssf2410\
 -g 9090 \
 -h 2 \
 -i ocnssfats-nsdb.ocnssf\
 -j ocnssf-db-creds \
 -k ocnssfats-nsconfig.ocnssf2410\
 -l ocatsnssf-nrf-stubserver.ocnssf2410\
 -m ocnssfats \
 -n ocatsnssf \
 -o 8081 \
 -p 8080 \
 -q 8080 \
 '''
 }
}
```

```

-r ocnssfats-nsselection.ocnssf2410 \
-s ocnssfats-nsauditor.ocnssf2410\
-t ocnssfats-nsavailability.ocnssf2410\
-u ocnssfats-nssubscription.ocnssf2410\
-v ocnssfats-ocnssf-app-info.ocnssf2410\
-w ocnssfats-ocnssf-perf-info.ocnssf2410\
-x 311 480 \

...
if(env.Include_Regression && "${Include_Regression}" == "YES"){
 sh '''sh /var/lib/jenkins/common_scripts/
merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-
Merged"
}
else{
 load "/var/lib/jenkins/ocnssf_tests/jenkinsData/Jenkinsfile-
Regression"
}
}
}

```

- Click **Save** after making necessary changes. The **Pipeline NSSF-Regression** screen appears.

### NSSF-Regression - Build with Parameters

The screenshot displays the 'Pipeline NSSF-Regression' configuration interface. At the top, it says 'This build requires parameters:'. Below this, there are two main sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'.

**EXECUTION OPTIONS:**

- TestSuite:** Regression
- Configuration\_Type:** Product\_Config (dropdown)
- Include\_NewFeatures:** NO (dropdown)

**FEATURES AND TESTCASES:**

**TestCases:** Select (dropdown)

**Stage: stage1 Group: group1**

**Selected Test Cases (checked):**

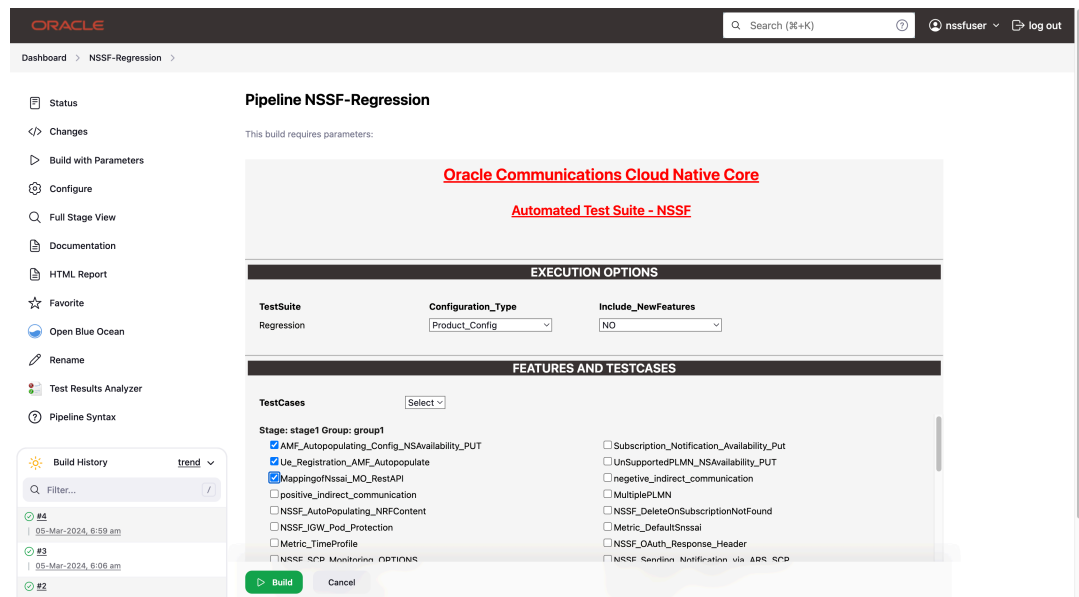
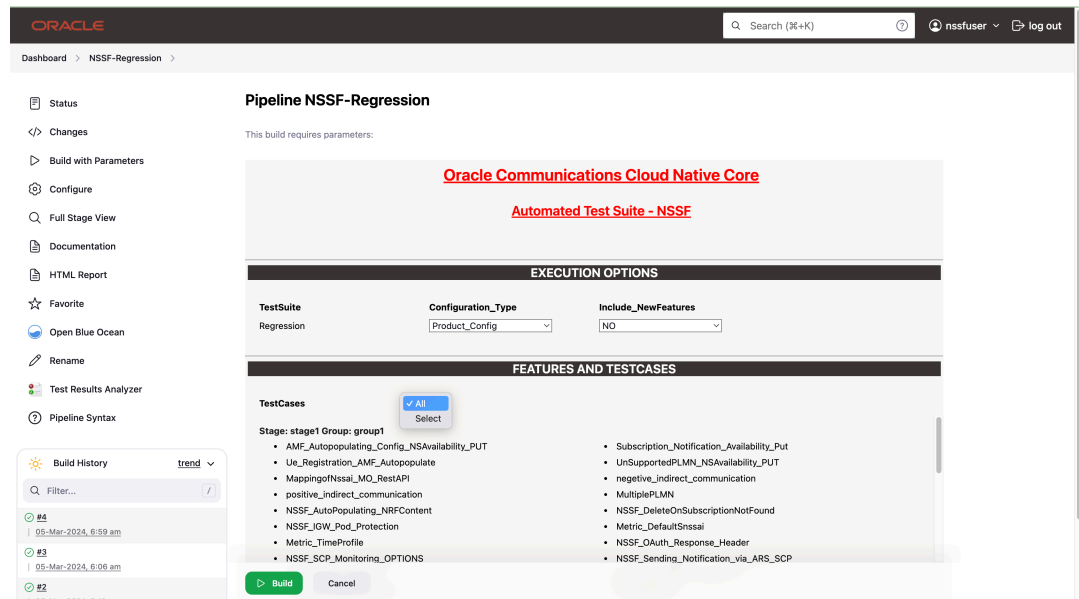
- AMF\_AutoPopulating\_Config\_NSAvailability\_PUT
- Ue\_Registration\_AMF\_AutoPopulate
- MappingOfNssai\_MO\_RestAPI

**Other Test Cases (unchecked):**

- Subscription\_Notification\_Availability\_Put
- UnsupportedPLMN\_NSAvailability\_PUT
- negative\_indirect\_communication
- MultiplePLMN
- NSSF\_DeleteOnSubscriptionNotFound
- Metric\_DefaultNssai
- NSSF\_OAuth\_Response\_Header
- NSSF\_SCP\_Monitoring\_OPTIONS
- NSSF\_Sentinel\_Notification\_via\_APS\_SCP

At the bottom, there are 'Build' and 'Cancel' buttons.

- By default, features "ALL," configuration type "Product\_config," and include new features "NO" will be selected. All test cases will be executed once you click on BUILD, as shown above.
- Features: ALL. Once you click on ALL, a dropdown "Select" option will appear. After selecting the "Select" option, as shown below, options will appear to select feature files. To run the selected feature files, click on "Build," as shown below.



74 features passed, 0 failed, 0 skipped  
 692 scenarios passed, 0 failed, 0 skipped  
 7049 steps passed, 0 failed, 0 skipped, 0 undefined  
 Took 49m35.126s

## 4.4.6 NSSF-Regression Documentation

To view NSSF Regression cases, go to **NSSF-Regression** latest pipeline build and click the **Documentation** link in the left navigation pane. The following screen appears. Click any functionality to view its test cases and scenarios of each test case as shown below:

[Back to NSSF-Regression](#) | [pages](#)[Zip](#)

## NSSF

### Related Pages

Here is a list of all related documentation pages:

MappingOfNssai\_MO\_RestAPI  
 NaSelection.Registration\_EPS\_to\_5G\_Handover  
 PimnLevelNSIPProfile  
 Ue.Registration\_AMF\_Autopopulate  
 AMF\_Autopopulation\_Configuration\_NSAvailability\_PUT  
 NSSF\_AutoPopulating\_NRFContent  
 nssf\_OAuth\_INSTANCEIDtoken  
 NSSF\_AVAILABILITY\_PATCH\_AND\_NEGATIVE\_CASES  
 NSSF\_BASIC\_SANITY\_CASES  
 NSSF\_BASIC\_SUBSCRIBE\_CASES  
 NSSF\_BASIC\_UPDATE\_CASES  
 NSSF\_CONFIG\_CASES  
 NSSF\_NaSelection\_Failure\_CASES  
 NSSF\_NaSelection\_PDU\_CASES  
 NSSF\_NaSelection.Registration\_CASES  
 NSSF\_IndirectComm\_Negative\_CASES  
 NSSF\_IndirectComm\_Positive\_CASES  
 NSSF\_MultiplePLMN\_CASES  
 NSSF\_EANAN\_CASES  
 NSSF\_SUBS\_PATCH\_CASES  
 NaSelection.Registration\_TaiRange  
 NaSubscription\_Notification\_TaiRange  
 NaAvailability\_Put\_TacRange  
 NaAvailability\_Subscription\_TacRange  
 NaSelection\_PDU\_TaiRange  
 NSSF\_Server\_Header  
 NSSF\_User\_Agent\_Header

Generated by [dewygen](#) 1.8.15

A sample of a few documentation features are as follows:

[Back to NSSF-Regression](#) | [pages](#)[Zip](#)

## NSSF

### NSSF\_User\_Agent\_Header

#### NSSF\_User\_Agent\_Header.feature

**Description :** This feature file validates the User-Agent headers based on the feature Enabled/Disabled

**Scenario-1 : If User-Agent feature is enabled, and overwrite is enabled, oc\_egressgateway\_user\_agent\_consumer metric with consumerInstancelid as a dimension should be pegged**

**Objective :** Validate oc\_egressgateway\_user\_agent\_consumer\_total metric when User-Agent feature is enabled, and overwrite is enabled.

**Pre-requisite :** NSSF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                                            | Expected Result                           |
|------------------------------------------------------------------------------------------------------|-------------------------------------------|
| 1.) Perform Agent Support Configuration by Configuration PUT request to NSSF                         | 1.) Verify successful response            |
| 2.) Perform Configuration by setting http response to 201 for notification in amf-stub server        | 2.) Verify successful response            |
| 3.) Perform subscription Configuration by sending create subscription request to NSSF                | 3.) Verify successful response            |
| 4.) Perform Snsai Auth Configuration by sending create Auth configuration request to NSSF            | 4.) Verify successful response 201        |
| 5.) Perform Fetch HTTP messages from amf stub server                                                 | 5.) Verify successful response            |
| 6.) Perform Fetch User-agent header from amf stub server                                             | 6.) Verify successful response            |
| 7.) Perform Post Fetch metrics for oc_egressgateway_user_agent_consumer_total                        | 7.) Validate with successful metric count |
| 8.) Perform Agent Support Configuration for disable the feature by Configuration PUT request to NSSF | 8.) Verify successful response            |

**Scenario-2 : If User-Agent feature is disabled, oc\_egressgateway\_user\_agent\_consumer metric shouldn't be pegged.**

**Objective :** Validate oc\_egressgateway\_user\_agent\_consumer\_total metric when User-Agent feature is disabled, and overwrite is enabled.

**Pre-requisite :** NSSF is already deployed with latest images and ATS client is up and running.

| Procedure                                                                                     | Expected Result                    |
|-----------------------------------------------------------------------------------------------|------------------------------------|
| 1.) Perform Agent Support Configuration by Configuration PUT request to NSSF                  | 1.) Verify successful response     |
| 2.) Perform Configuration by setting http response to 201 for notification in amf-stub server | 2.) Verify successful response     |
| 3.) Perform subscrition Configuration by sending create subscription request to NSSF          | 3.) Verify successful response     |
| 4.) Perform Snsai Auth Configuration by sending create Auth configuration request to NSSF     | 4.) Verify successful response 201 |
| 5.) Perform delete http response for notification in amf-stub server                          | 5.) Verify successful response     |

## 4.5 Running Policy Test Cases using ATS

This section describes how to run CNC Policy test cases using ATS.

For this ATS-Policy release, scenarios or test cases are included for the following three deployment modes:

- Converged Policy
- PCF only for both TLS enabled (server side) and disabled mode
- PCRF only

**Note**

Restart the NRF-client pod of CNC Policy for UDR and CHF discovery as part of each test case.

## 4.5.1 Prerequisites

### PCF only deployment

To run CNC Policy test cases in PCF mode, ensure that the following prerequisites are met:

- Deploy CNC Policy 24.1.0 with default helm configurations using helm charts to run all test cases. The ATS version must be compatible with CNC Policy 24.1.0. For more information on how to install CNC Policy, see *Oracle Communications Cloud Native Core Policy Installation, Upgrade and Fault Recovery Guide*.
- For running ATS features for optional services such as LDAP, Notifier, and Usage Monitoring, ensure that the respective service is enabled on CNC Policy. If this service is not enabled, the test case for this feature will fail.
- For running SOAP notification service, ensure that the SOAP connector service is enabled on CNC Policy. If this service is not enabled, the test case for this feature will fail.
- For using Controlled Shutdown feature, ensure that enableControlledShutdown parameter is enabled on CNC Policy. If this parameter is not enabled, the test case for this feature will fail.
- Install py-stub, ocamf-stub, ocdns-bind stub, and oc-ldap stub for Converged Policy and PCF deployment modes.
- Ensure there is a single pod for each microservice in the CNC Policy deployment for ATS Prometheus metrics validation to work.
- Users can add new test cases, remove unwanted test cases, and modify existing test cases in the custom test case folders (**cust\_newfeatures**, **cust\_regression**, and **cust\_performance**). It does not impact the original test cases packaged with the product under newfeatures, regression, and performance folders. For more details about custom test case folders, see [Custom Folder Implementation](#).
- The ATS prometheus metrics validation works only when:
  - the metrics suffixes are not configured
  - installation has a single pod for each microservice in the BSF deployment
- For PRE and Policy\_addition timer, Update PRE deployment to reduce the value of POLLING\_INTERVAL from 10s to 5s and POLLING\_INTERVAL\_POLICYTABLE from 30s to 10s.

**Figure 4-39 PRE and Policy\_addition timer**

```
[cloud-user@platform-bastion-1 ~]$ kubectl get deploy -n ocpcf | grep 'pre'
ocpcf-ocpm-pre 1/1 1 1 10h
```

Run the following command to reduce the value of polling deployment:

```
[cloud-user@platform-bastion-1 ~]$ kubectl edit deploy ocpcf-ocpm-pre -n ocpcf
```

- The following deployment is displayed:

**Figure 4-40 POLLING\_INTERVAL and POLLING\_INTERVAL\_POLICYTABLE Deployment**

```
- name: POLLING_INTERVAL
 value: "5000"
- name: POLLING_INTERVAL_POLICYTABLE
 value: "10000"
```

#### Note

The value of POLLING\_INTERVAL can be updated to 5000 and POLLING\_INTERVAL\_POLICYTABLE to 10000 if it is not updated.

For the policy\_additon timer, the following changes have been made in the config.xml file for NewFeatures, Performance, and Regression.

**Figure 4-41 Policy Addition Timer**

```
//a = SELECTED_NF b = PCF_NAMESPACE c = PROMSVC_NAME d = GOSTUB_NAMESPACE
//e = SECURITY f = PCF_NFINSTANCE_ID g = POD_RESTART_TIME h = POLICY_TIME
//i = NF_NOTIF_TIME j = RERUN_COUNT k = INITIALIZE_TEST_SUITE l = STUB_RESPONSE_TO_BE_SET
//m = POLICY_CONFIGURATION_ADDITION n = POLICY_ADDITION o = NEXT_MESSAGE
//p = PROMSVCIP q = PROMSVCPORT r = TIME_INT_POD_DOWN s = POD_DOWN_RETRIES
//t = TIME_INT_POD_UP u = POD_UP_RETRIES v = ELK_WAIT_TIME w = ELK_HOST
//x = ELK_PORT y = STUB_LOG_COLLECTION z = LOG_METHOD A = enable_snapshot B = svc_cfg_to_be_read C = PCF_API_ROOT
```

For every SUT, change the of value of -n from 15 to 7, as follows:

**Figure 4-42 Change in SUT Value**

```
-k 0 \
-l 1 \
-m 1 \
-n 7 \
-o 1 \
-p occone-prometheus-
-q 80\
-r 30\
--
```

- In the application-config configmap, configure the following parameters with the respective values:

- Set the value of `primaryNrfApiRoot` as `primaryNrfApiRoot=nf1stub.<namespace_pystubs_are_deployed_in>.svc:8080`.  
For example:  
`primaryNrfApiRoot=nf1stub.ocats.svc:8080`
- Remove `secondaryNrfApiRoot`.  
For example: `secondaryNrfApiRoot=nf1stub.ocats.svc:8080`
- Set the value of `nrfClientSubscribeTypes` to `UDR, CHF, NWDAF`.
- Remove `supportedDataSetId`.  
For example: `supportedDataSetId=POLICY`

**Note**

These values can also be configured during Policy deployment.  
Run the following command to get all configmaps in your namespace:  
`kubectl get configmaps -n <Policy_namespace>`

- Before running Policy test cases using ATS, restart `nrf-client-nfdiscovery` and `nrf-client-nfmanagement` pods.
  - Edit the Alternate Route Service deployment pointing towards DNS Stub.
1. Run the following command to get searches information from `dns-bind` pod to enable communication between Alternate Route and `dns-bind` service:

```
kubectl exec -it <dns-bind pod> -n <NAMESPACE> -- /bin/bash -c
'cat /etc/resolv.conf' | grep search | tr ' ' '\n' | grep -v 'search'
```

The following output is displayed after running the command:

**Figure 4-43 Sample Output**

```
[flanders@bastion-1 ~]$ kubectl exec -it ocdns-stub-ocdns-bind-7467cf56c8-5d9q2 -n flanders-ns -- /bin/bash -c 'cat /etc/resolv.conf' | grep search | tr
'\n' | grep -v 'search'
flanders-ns.svc.inferno.lab.us.oracle.com
svc.inferno.lab.us.oracle.com
inferno.lab.us.oracle.com
```

By default alternate service will point to CoreDNS and you will see following settings in deployment file:

**Figure 4-44 Alternate Route Service Deployment File**

```
terminationMessagePolicy: File
dnsPolicy: ClusterFirst
restartPolicy: Always
```

2. Run the following command to edit the deployment file and add the following content in alternate service to query DNS stub:

```
$kubectl edit deployment ocpcf-occpn-alternate-route -n ocpcf
```



- a. Add the IP Address of the nameserver that you have recorded after installing the DNS stub (cluster IP Address of DNS Stub).
- b. Add the search information one by one which you recorded earlier.
- c. Set dnsPolicy to "None".

```
dnsConfig:
 nameservers:
 - 10.233.33.169 // cluster IP of DNS Stub
 searches:
 - ocpcf.svc.occne15-ocpcf-ats
 - svc.occne15-ocpcf-ats
 - occne15-ocpcf-ats
 dnsPolicy: None
```

For example:

**Figure 4-45 Example**

```
terminationMessagePath: /dev/termination-log
terminationMessagePolicy: File
dnsConfig:
 nameservers:
 - 10.233.31.245
 searches:
 - flanders-ns.svc.inferno.lab.us.oracle.com
 - svc.inferno.lab.us.oracle.com
 - inferno.lab.us.oracle.com
 dnsPolicy: None
restartPolicy: Always
schedulerName: default-scheduler
```

### PCRF only deployment

To run CNC Policy test cases in PCRF mode, ensure that the following prerequisites are met:

- Run the following command to set the log level to Debug in Diam-GW statefulset:

```
kubectl edit statefulset <diam-gw statefulset name> -n <namespace>
```

```
name: LOGGING_LEVEL_APP
```

```
value: DEBUG
```

- Set the peer Configuration using configuration map for backend peers. Edit and set the default configuration of Diameter peer in the diam configuration map and set responseOnly flag to true.

```
kubectl edit cm oc-diam-gateway-config-peers -n <namespace>
```

```
nodes:
```

```
- name: 'ocpcf-occpn-pcrf-core'
 type: 'pcrf'
 responseOnly: true
 host: ocpcf-occpn-pcrf-core-headless
 port: 3868
```

```
realm: ''
identity: ''
```

### Converged mode deployment

To run CNC Policy test cases in Converged mode, ensure that the following prerequisites are also met:

- Deploy CNC Policy 24.1.0 with default helm configurations using helm charts to run all test cases. The ATS version must be compatible with CNC Policy 24.1.0. For more information on how to install CNC Policy, see *Oracle Communications Cloud Native Core Policy Installation and Upgrade Guide*.
- For running ATS features for optional services such as LDAP, Notifier, and Usage Monitoring, ensure that the respective service is enabled on CNC Policy.
- Install Prometheus server in cluster.
- Database cluster should be in a running state with all the required tables. Ensure there are no previous entries in database before executing test cases.
- Deploy ATS in the same namespace as Policy using Helm Charts.
- User must not initiate a job in two different pipelines at the same time.
- The installation should have only one pod for each microservice related to ATS Prometheus Metrics validation to work in the CNC Policy deployment.
- If Service Mesh check is enabled, apply the below settings:  
For fetching the metrics from the Prometheus, create a destination rule. In most deployments, Prometheus is kept outside of the service mesh. Hence, a destination rule is required to communicate between TLS enabled entity (ATS) and non-TLS entity (Prometheus). Created the rule as follows:

```
kubectl apply -f - <<EOF

apiVersion:networking.istio.io/v1alpha3
kind:DestinationRule
metadata:
 name:prometheus-dr
 namespace:ocats
spec:
 host:oso-prometheus-server.pcf.svc.cluster.local
 trafficPolicy:
 tls:
 mode:DISABLE
EOF
```

Here,

- 1.) name, indicates the name of destination rule.
- 2.) namespace, indicates where the ATS is deployed.
- 3.) host, this indicates the hostname of the prometheus server.

- For running ATS features for Model D Indirect Communication Model, ensure that the custom values yaml of CNC Policy has the following parameters with the specified values:

```
sbiRoutingConfigMode: REST
routeConfigMode: REST
configureDefaultRoute: true
```

- Update the `occpn-servicemesh-config-custom-values-24.1.0.yaml` of CNC Policy with the following `virtualService` configuration for Egress Gateway:

```
virtualService:
- name: nrftvirtual1
host: ocpcf-occpn-egress-gateway
destinationhost: ocpcf-occpn-egress-gateway
port: 8000
exportTo: |-
["."]
attempts: "0"
```

- Restart the `ocpm-ldap-gateway` pod manually before running the LDAP-specific ATS test cases.
- To ensure the consistent functioning of ATS related to the audit service, modify the audit deployment by reducing the value of `AUDIT_NOTIFY_SCHEDULER_POLLING_INTERVAL_MILLISEC` from 30000 to 1000 (changing from 30 seconds to 1 second).

```
$ kubectl get deploy -n ocpcf | grep 'audit'
ocpcf-ocpm-audit-service 1/1 1
1 10h
```

```
$ kubectl edit deploy ocpcf-ocpm-audit-service -n ocpcf
```

The deployment will open. Scroll down until below fields are seen.

```
- name: AUDIT_NOTIFY_SCHEDULER_POLLING_INTERVAL_MILLISEC
value: "30000"
```

Update the value of `AUDIT_NOTIFY_SCHEDULER_POLLING_INTERVAL_MILLISEC` to 1000 if it isn't already.

#### Note

In order to run stage2 or stage3 cases using `configuration_type` of `custom_config`, replace the "data" folder with "cust\_data" folder in `stage<x>.txt` under `/var/lib/jenkins/ocpcf_tests/cust_data/common/stage_hooks/stage<x>` directory.

## 4.5.2 Logging into ATS

Before logging into ATS GUI, it is important to get the nodeport of the service, 'ocats-Policy'.

To get the nodeport, perform the following steps:

```
kubectl get svc -n <Policy_namespace>
```

**Example:**

```
kubectl get svc -n ocpf
```

The output of the command is:

|                 |              |              |              |
|-----------------|--------------|--------------|--------------|
| ocats-ocats-pcf | LoadBalancer | 10.233.56.56 | 10.75.225.49 |
| 8080:31944/TCP  | 19h          |              |              |

To log in to ATS, open the web browser and type the following URL:

`http://<Worker-Node-IP>:<Node-Port-of-ATS>`

If the ocats-policy Service has an external IP available, <SVC external IP> can also be used to log in to ATS.

`http://<SVC external IP>:8080`

**Example:** `http://10.75.225.49:8080`

### Running ATS

To run ATS test cases, perform the following steps:

1. Enter the **username** as `policyuser` and **password** as `polycypasswd`.
2. Click **Sign in**.

**Note**

To modify default login password, see [Modifying Login Password](#).

The following page appears showing policy pre-configured pipelines:

Figure 4-46 Pre-configured Pipelines

**Oracle Communications Cloud Native Core**

**Automated Test Suite**

**CNCATS POLICY : 24.1.0**

All

| S | W | Name ↓             | Last Success       | Last Failure | Last Duration |  | F |
|---|---|--------------------|--------------------|--------------|---------------|--|---|
|   |   | Policy-HealthCheck | N/A                | N/A          | N/A           |  |   |
|   |   | Policy-NewFeatures | 2 days 12 hr<br>#1 | N/A          | 1 hr 36 min   |  |   |
|   |   | Policy-Regression  | N/A                | N/A          | N/A           |  |   |

The description of each pipeline is as follows:

- **Policy-HealthCheck:** This pipeline checks if CNC Policy and ATS are deployed correctly. This shows only when the user has enabled this feature at the time of installing CNC Policy ATS.
- **Policy-NewFeatures:** This pipeline has all the new test cases delivered for CNC Policy 24.1.0.
- **Policy-Regression:** This pipeline has all the test cases delivered in Policy ATS - 24.1.0.
- **Policy-Performance:** This pipeline is not operational as of now. It is reserved for future releases of ATS.

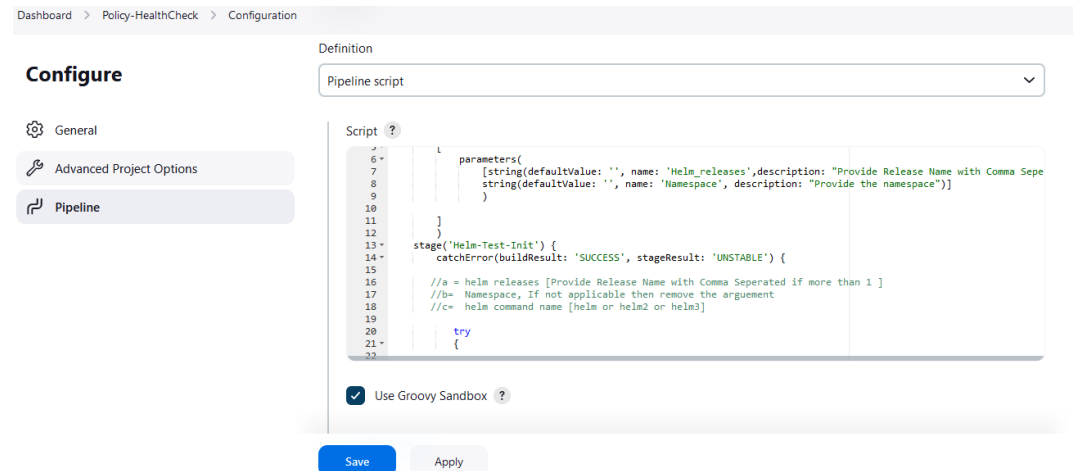
### 4.5.3 Running Policy-HealthCheck Pipeline

This is a pre-configured pipeline where ATS performs a test probe with SUT. It triggers helm test & provides the results in Jenkins Console Logs.

You can run Policy-HealthCheck pipeline to check if all CNC Policy pods are up and running. If yes, it provides the status as successful. If any pod is down due to any reason, then the pipeline fails.

To configure the parameters for **Policy-HealthCheck** pipeline, perform the following steps:

1. Click **Policy-HealthCheck** in the **Name** column.
2. Click **Configure** in the left navigation pane.
3. When you scroll-down, the **General** tab becomes active. Be sure that the screen loads completely.
4. Continue scrolling down until the **Pipeline** tab becomes active. The following is a screen capture that shows the Pipeline script:

**Figure 4-47 Helm Test Script**

- In the Script area under the Pipeline section, users may customize the values for the following parameters:

**NOT\_SUPPORTED**

Do not modify values of parameters other than the ones described in this section.

- a:** Change this parameter to update the helm release name where Policy is deployed in your bastion.
- b:** Change this parameter to update the namespace where Policy is deployed in your bastion.

- Click **Save** to update the values.

**Running Helm Test**

To run CNC Policy test cases, click **Build Now**.

## 4.5.4 Running Policy-NewFeatures Pipeline

**Policy-New Features Pipeline**

This is a pre-configured pipeline where all the Policy new test cases are executed.

To configure the parameters for **Policy-NewFeatures** pipeline, perform the following steps:

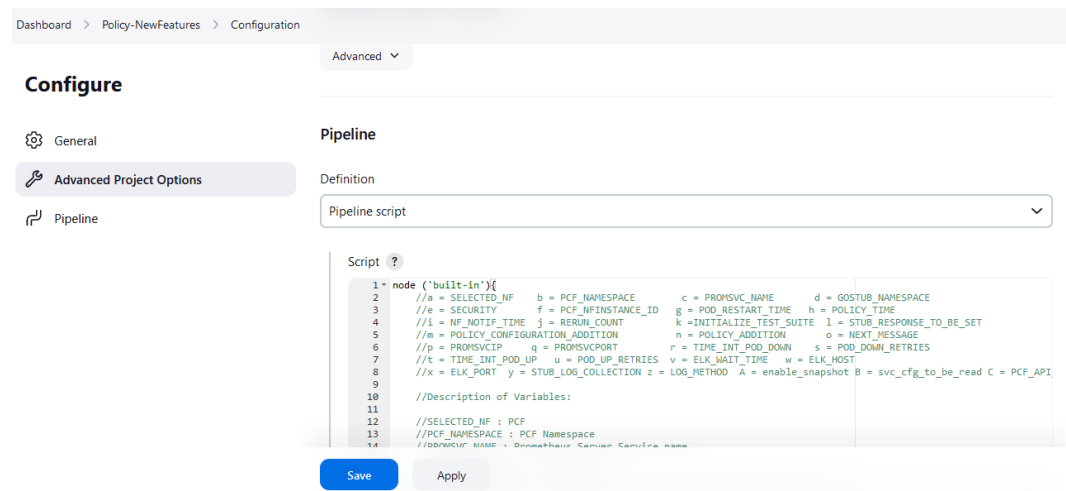
**Note**

Configuring the parameters is a one-time activity.

- Click **Policy-NewFeatures** in the **Name** column.
- Click **Configure** in the left navigation pane.
- When you scroll-down, the **General** tab becomes active. Be sure that the screen loads completely.

4. (Optional) If you want to retain a specified number of builds in the persistent volume, select the check box for **Discard old Builds**. It is recommended to configure this option; otherwise, Persistent Volume may fail when the builds are exceedingly high. Enter values for the following parameters:
  - **Days to keep builds**: Specifies the number of days for which build records are stored.
  - **Max # of builds to keep**: Specifies the number of build records to store.
5. Continue scrolling down until the **Pipeline** tab becomes active. The following is a screen capture that shows the Pipeline script:

Figure 4-48 Policy Pipeline Script



6. In the Script area under the Pipeline section, users may customize the values for the following parameters:

**NOT\_SUPPORTED**

Do not modify values of parameters other than the ones described in this section.

- **b**: Change this parameter to update the namespace where Policy is deployed in your bastion.
- **d**: Change this parameter to update the namespace where gostubs are deployed in your bastion.
- **e**: Set this parameter as 'unsecure', if you intend to run ATS in TLS disabled mode. Else, set this parameter as 'secure'.
- **g**: Set this parameter to more than 45 secs. The default wait time for the pod is 45 secs. Every TC requires restart of the nrf-client-management pod.
- **h**: Set this parameter to more than 60 secs. The default wait time to add a configured policy to the database is 60 secs.
- **i**: Set this parameter to more than 140 secs. The default wait time for Nf\_Notification Test Cases is given as 140 secs.
- **k**: Use this parameter to set the waiting time to initialize Test Suite.
- **l**: Use this parameter to set the waiting time to get response from Stub.

- **m:** Use this parameter to set the waiting time after adding Policy Configuration.
- **n:** Use this parameter to set the waiting time after adding Policy.
- **o:** Use this parameter to set the waiting time before sending next message.
- **p:** Use this parameter to set Prometheus Server IP.
- **q:** Use this parameter to set Prometheus Server Port.
- **r:** Use this parameter to set the interval after which the POD status is checked when it is down.
- **s:** Use this parameter to set number of retry attempt in which will check the pod down status
- **t:** Use this parameter to set the interval after which we check the POD status if its UP.
- **u:** Use this parameter to set number of retry attempt in which will check the pod up status.
- **v:** Use this parameter to set Wait time to connect to Elastic Search
- **w:** Use this parameter to set Elastic Search HostName
- **x:** Use this parameter to set Elastic Search Port
- **y:** Use this parameter to set To Enable/Disable Stub logs collection
- **z:** Use this parameter to set Log collection endpoint either Elasticsearch or Kubernetes
- **A:** Use this parameter to enable or disable snapshots that are created at the start and restored at the end of each test run
- **B:** Use this parameter to for timer to wait for importing service configurations
- **C:** Use this parameter to set PCF\_API\_ROOT information for Ingress gateway service name and port.
- **D:** Use this parameter to set bulk\_import\_to\_complete to add custom time in Jenkins post bulk imports.

(Optional) To collect application logs per failed scenario, user can configure the values for the following parameter:

- **z:** If you want log collection to happen through Elastic search, set the value for this parameter as `Elasticsearch`. If not, specify the value as `Kubernetes`.  
If you want to collect logs Elastic search, it is required to configure the values for the following parameters:
    - **v:** Specifies the wait time to connect to Elastic search (`ELK_WAIT_TIME`).
    - **w:** Specifies the host name of Elastic search (`ELK_HOST`). For example, `occne-elastic-elasticsearch-master.occne-infra/`
    - **x:** Specifies the port for Elastic search (`ELK_PORT`). For example, `9200`.
  - **y:** If you want to collect stub logs, set the value for this parameter as `yes`. If not, specify the value as `no`.
7. **A:** To enable or disable snapshots that are created at the start and restored at the end of each test run.
  8. Click **Save** to update the values. The Policy\_NewFeatures Pipeline page appears.



**Note**

It is recommended to save the pipeline script in your local machine as it is needed at the time of ATS pod restart.

**Extracting Application Logs**

If any scenarios failed after running the pipeline job, perform the following steps:

1. Log in to the ATS pod:

```
kubectl exec -it pod/ocnp1-ocats-ocats-policy-xxxxxxx -n ocpcf bash
```

2. Go to build directory:

```
cd /var/lib/jenkins/.jenkins/jobs/Policy-Regression/builds/1
```

For example:

```
cd $JENKINS_HOME/jobs/$JOB_NAME/builds/$BUILD_NUMBER/
```

3. Extract the applogs.zip file:

```
unzip applogs.zip
```

4. After successfully unzipping the file, open the applog folder to view pod logs for failed scenarios:

```
(env) [jenkins@gapcf-ocats-ocats-policy-6df5cd84c-c6cm4 applog]$ pwd
/var/lib/jenkins/.jenkins/jobs/Policy-Regression/builds/3/applog
(env) [jenkins@gapcf-ocats-ocats-policy-6df5cd84c-c6cm4 applog]$ ls -ltrh
total 1.4M
-rw-r--r--. 1 jenkins jenkins 461K Sep 1 14:49 Initial_Run-
Send_Npcf_SMPolicyControl_Create_request_message_to_PCF_and_verify_the_praI
nfos_structure_is_downloaded_in_the_response_message_to_SMF_also_check_requ
ests_total_metric_incremented_in_the_PCF.log
-rw-r--r--. 1 jenkins jenkins 459K Sep 1 14:49 1st_Rerun-
Send_Npcf_SMPolicyControl_Create_request_message_to_PCF_and_verify_the_praI
nfos_structure_is_downloaded_in_the_response_message_to_SMF_also_check_requ
ests_total_metric_incremented_in_the_PCF.log
-rw-r--r--. 1 jenkins jenkins 459K Sep 1 14:49 2nd_Rerun-
Send_Npcf_SMPolicyControl_Create_request_message_to_PCF_and_verify_the_praI
nfos_structure_is_downloaded_in_the_response_message_to_SMF_also_check_requ
ests_total_metric_incremented_in_the_PCF.log
```

**Running Policy Test Cases**

To run CNC Policy test cases, perform the following steps:

1. Click the **Build with Parameters** link available in the left navigation pane of the **Policy-NewFeatures** Pipeline screen. The following screen appears.

**Figure 4-49 Pipeline Policy-NewFeatures**

The screenshot displays the Oracle Communications Cloud Native Core Automated Test Suite (ATS) configuration page for the 'Policy-NewFeatures' test suite. The page is divided into two main sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'.

**EXECUTION OPTIONS:**

- TestSuite:** NewFeatures
- SUT:** PCF (dropdown menu)
- Configuration\_Type:** Product\_Config (dropdown menu)
- FilterWithTags:** No (dropdown menu)
- Fetch\_Log\_Upon\_Failure:** YES (dropdown menu)

**FEATURES AND TESTCASES:**

- Features:** All (dropdown menu)

At the bottom of the page, there is a 'Build History' section with a 'trend' dropdown and a 'Build' button.

2. Select any of following valid values from the drop-down for **SUT**:
  - PCF
  - cnPCRF
  - Converged Policy
3. Select any of following valid values from the drop-down for **Configuration\_Type**:
  - **Product\_Config**: On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the key-value pair and yaml files defined or present in the "Product Config" folder.
  - **Custom\_Config**: On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the key-value pair and yaml files defined or present in the "Custom Config" folder.  
To use the parameterization feature, always select the **Custom\_Config** option. User can copy, add, or delete the required test cases that are available for the Converged Policy, PCF or PCRF and place them appropriately within the **custom** folder for **Policy-NewFeatures**. Reload the page to view the test cases available in the **custom** NewFeatures folder.

For more information, see [Parameterized approach for SUT custom configuration](#).
4. For the **Select\_Option** field, select any of the following drop-down values:
  - **All**: By default, all the Policy test cases are selected for execution.
  - **Single/MultipleFeatures**: This option allows you to select any number of test cases you want to run from the list of total test cases. Select the check-box for each feature you want to run.

Based on your selection, related **Test Cases** appear on the page.

5. To collect logs for any given build, select YES from the drop-down menu of **Fetch\_Log\_Upon\_Failure**.
6. Click **Build** and select **Console Output** to view the test results. The following is a sample test result output:

**Figure 4-50 Test Result Output in Console**

```
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnftest/common_steps.py:194
2023-06-22 05:43:21,589[32m INFO LOG.stage1.group1.STEP:331 [0m] [32mocng[0m
2023-06-22 05:43:21,596[31m ERROR LOG.stage1.group1.Prometheus_Client:71 [0m] [31mCould not fetch scrapeInterval from prometheus[0m
2023-06-22 05:43:21,599[31m ERROR LOG.stage1.group1.Prometheus_Client:91 [0m] [31mCould not fetch target_namespace from prometheus[0m
Then Waiting for test_suite_to_initialize
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnftest/pcf_steps.py:2515
2023-06-22 05:43:21,606[32m INFO LOG.stage1.group1.STEP:2532 [0m] [32mWait for 0s[0m
Then Unset Variable
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnftest/pcf_steps.py:286
Given Check And Set Policy Namespace
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnftest/pcf_steps.py:105
2023-06-22 05:43:21,608[32m INFO LOG.stage1.group1.STEP:110 [0m] [32mPCF Namespace set to vega-ns5[0m
Given Create Service Client
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnftest/pcf_steps.py:115
2023-06-22 05:43:21,883[32m INFO LOG.stage1.group1.STEP:133 [0m] [32mATS Pod IP 10.233.102.218[0m
2023-06-22 05:43:21,883[32m INFO LOG.stage1.group1.STEP:128 [0m] [32mDiam pod name vega-ns5-oc-diam-gateway-0[0m
2023-06-22 05:43:21,884[32m INFO LOG.stage1.group1.STEP:130 [0m] [32mvega-ns5-oc-diam-gateway-0[0m
2023-06-22 05:43:21,884[32m INFO LOG.stage1.group1.STEP:134 [0m] [32mDiam gw pod info - {'vega-ns5-oc-diam-gateway-0': '10.233.70.70'}[0m
2023-06-22 05:43:21,886[32m INFO LOG.stage1.group1.STEP:140 [0m] [32mATS svc name ocats-policy[0m
2023-06-22 05:43:21,888[32m INFO LOG.stage1.group1.STEP:212 [0m] [32mLdap org1 service name is: vega-ns5-oc-ldap-org1-service[0m
2023-06-22 05:43:21,889[32m INFO LOG.stage1.group1.STEP:214 [0m] [32mLdap org1 service cluster IP is: 10.233.56.158[0m
2023-06-22 05:43:21,889[32m INFO LOG.stage1.group1.STEP:218 [0m] [32mLdap org1 service port is: 8080[0m
2023-06-22 05:43:21,889[32m INFO LOG.stage1.group1.STEP:221 [0m] [32mLdap org1 service secure port is: 8081[0m
```

**Note**

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

**Parameterized approach for SUT custom configuration**

Using this feature, user can make the following customizations to custom folders

- Add new test cases by adding datafiles
- Remove test cases
- Modify the parameters and their values in the key-value pair or <feature>.yaml files

To run ATS test cases, user can maintain as many versions of Custom Config folder by using the following naming convention:

Cust ConfigN

where N can be any number

At the time of execution, ensure to rename the required folder to **Cust Config** folder as Jenkins always retrieves data from this folder when user selects **Custom\_Config**.

**Updating Global Parameters**

To use Custom\_Config, it is required to change the value of **cust\_folder** from **data** to **cust\_data** in global.yaml file. In addition, you can customize the parameters and their respective values in the global.yaml as per the requirements.

To use custom diameter identity, update the value of parameter "diam\_gw\_id" under global.yaml file to the one configured for "DIAMETER\_Identity" under diam-gw's statefulset.

**Updating Feature Parameters**

Consider the following points when customizing <feature>.yaml files for parameterized feature:

- In addition to global.yaml parameters, feature files may also contain parameters for which user can update values at the time of running pipelines.

- Changing the values of parameters tagged as "Feature Specific Value" may cause failures at the time of running pipelines.
- Values for parameters tagged with #START\_GLOBAL and #END\_GLOBAL tags take values from global.yaml.

#### **Note**

For CNC Policy-ATS release 24.1.0, parameterization is supported for PCF features only.

### **Incorrect Values**

If the value to be parameterized is incorrect or not updated correctly, the following error is generated:

```
2021-12-17 09:02:43,908[32m INFO LOG.featureHooks:14 [0m] [32mInside
before_feature[0m
2021-12-17 09:02:43,908[32m INFO LOG.goldenConfig:305 [0m] [32mInside
goldenConfig.py[0m
2021-12-17 09:02:43,912[31m ERROR LOG.goldenConfig:93 [0m] [31mPlease
check if the key metricsEnabled is valid or if it is present under correct
section[0m
2021-12-17 09:02:43,913[32m INFO LOG.goldenConfig:324 [0m] [32mExiting
goldenConfig.py[0m
2021-12-17 09:02:43,913[32m INFO LOG.featureHooks:27 [0m] [32mExiting
before_feature[0m
```

The execution continues to happen without any changes to the data files.

### **Config-Rollback (Snapshot Backup and Restore)**

Using this feature, user can take a snapshot of initial configuration of the system, and then restore it at a later stage. On enabling this feature, a snapshot of system configurations is taken before ATS run starts and is reverted on completion of ATS job.

### **Queuing Jenkins Jobs**

Using this feature, you can queue a second job even when current job is still running. The second job can be triggered either from the same or a different pipeline.

The following table lists various scenarios on how queuing works:

**Table 4-6 Queuing Jenkins Jobs**

| Concurrent Builds | New Features Current Build | New Features New Build | Regression Current Build | Regression New Build Build | Result                                       |
|-------------------|----------------------------|------------------------|--------------------------|----------------------------|----------------------------------------------|
| Enabled           | Running                    | Triggered              | NA                       | NA                         | New-Build of New-Features is added to queue. |
| Enabled           | Running                    | NA                     | NA                       | Triggered                  | New-Build of Regression is added to queue.   |

Table 4-6 (Cont.) Queuing Jenkins Jobs

| Concurrent Builds | New Features Current Build | New Features New Build | Regression Current Build | Regression New Build Build | Result                                       |
|-------------------|----------------------------|------------------------|--------------------------|----------------------------|----------------------------------------------|
| Disabled          | NA                         | NA                     | Running                  | Triggered                  | New-Build of Regression is added to queue.   |
| Disabled          | NA                         | Triggered              | Running                  | NA                         | New-Build of New-Features is added to queue. |

### Test Result Analyzer

Using the Test Result Analyzer plug-in, user can view consolidated and detailed reports. For more information, see [Test Results Analyzer](#) section.

### Test Case Mapping to Features and Display Total Counts

With this feature, users can view total count of features, test cases, and TestCase mapping for each feature of CNC Policy on ATS GUI. For more information, see [Support for Test Case Mapping and Count](#).

## 4.5.5 Policy New Features Documentation

This section describes the documentation for **Policy-NewFeatures** pipeline.

To view the documentation for any of the CNC Policy features, on the ATS home page, click Policy-NewFeatures. Then, click **Documentation** in the left navigation pane.

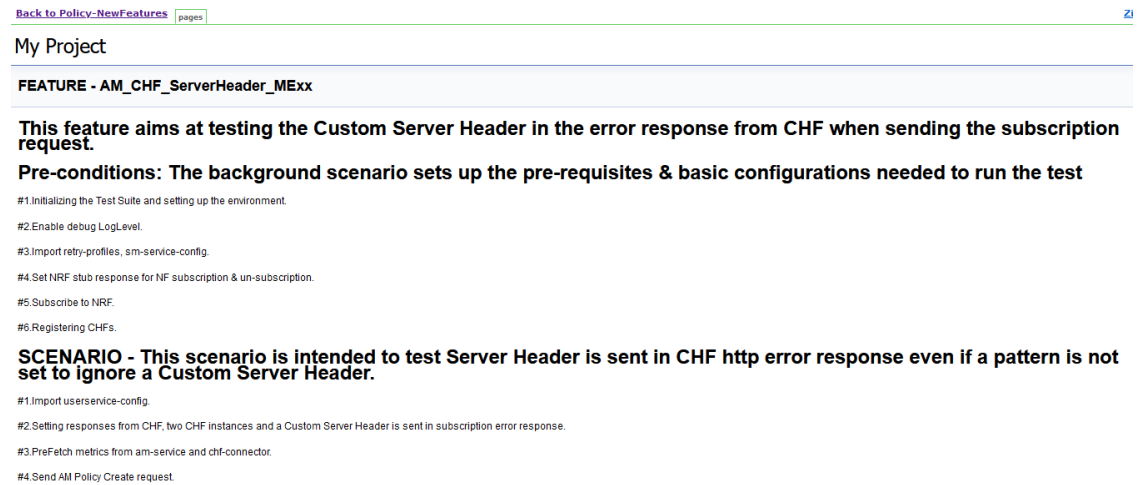
On clicking Documentation, the following page opens with the list of new features:

Figure 4-51 Policy-NewFeatures Feature List

|                                                                                                               |  |
|---------------------------------------------------------------------------------------------------------------|--|
| <a href="#">Back to Policy-NewFeatures</a> <span>pages</span>                                                 |  |
| <b>My Project</b>                                                                                             |  |
| <b>Related Pages</b>                                                                                          |  |
| Here is a list of all related documentation pages:                                                            |  |
| <a href="#">FEATURE - AM_CHF_ServerHeader_MExx</a>                                                            |  |
| <a href="#">FEATURE - AM_Concurrency_Extended_ID</a>                                                          |  |
| <a href="#">FEATURE - Bulwark_Support_SM_Create_Delete_UpdateNotify_PDSNotification_RedCap_ocLogId_verify</a> |  |
| <a href="#">FEATURE - Collision_Detection_Terminate_Notify</a>                                                |  |
| <a href="#">FEATURE - Gx_NUDR_Sd_Sy_Lookup_timeout</a>                                                        |  |
| <a href="#">FEATURE - Gx_NUDR_SySLR_Enhancements</a>                                                          |  |
| <a href="#">FEATURE - Pending_Transaction_AM_Retry_NoRetry_Extended_ID</a>                                    |  |
| <a href="#">FEATURE - Pending_Transaction_N7_Fast_Lock_Bulwark</a>                                            |  |
| <a href="#">FEATURE - Pending_Transaction_UE_Extended_ID</a>                                                  |  |
| <a href="#">FEATURE - SM_CHF_ServerHeader_MExx</a>                                                            |  |
| <a href="#">FEATURE - SM_policy_audit_Terminate_Notify</a>                                                    |  |
| <a href="#">FEATURE - SM_Sy_Enhancement</a>                                                                   |  |
| <a href="#">FEATURE - UE_CHF_ServerHeader_MExx</a>                                                            |  |
| <a href="#">FEATURE - UE_Concurrency_Extended_ID</a>                                                          |  |
| <a href="#">FEATURE - UMAudit</a>                                                                             |  |

**Example:** On clicking **FEATURE - Binding\_Header\_UEPolicy\_and\_N1N2Transfer**, the following screen opens:

**Figure 4-52 FEATURE - AM\_CHF\_ServerHeader\_MExx**



Based on the features covered under Documentation, the **Build Requires Parameters** screen displays test cases.

To navigate back to the Policy-NewFeatures pipeline, click **Back to Policy-NewFeatures** link available on the top left corner of the screen.

## 4.5.6 Running Policy Regression Pipeline

This section describes how to run test cases for CNC Policy Regression pipeline.

The **Policy-Regression** pipeline is a pre-configured pipeline where all the test cases of previous releases are available. For an example, for Policy 24.1.0, this pipeline has all the test cases released till Policy 24.1.0.

### Running Policy Test Cases

To run CNC Policy test cases in regression pipeline, perform the following steps:

1. Click **Policy-Regression** in the **Name** column.
2. Click the **Build with Parameters** link available in the left navigation pane of the **Policy-Regression** Pipeline screen. The following screen appears.

**Figure 4-53 Regression Pipeline**

The screenshot displays the Oracle Communications Cloud Native Core Automated Test Suite (ATS) interface for the Policy-Regression pipeline. The interface is divided into several sections:

- Header:** Oracle Communications Cloud Native Core, Automated Test Suite - POLICY
- EXECUTION OPTIONS:**
  - TestSuite:** Regression
  - SUT:** PCF
  - Configuration\_Type:** Product\_Config
  - FilterWithTags:** No
  - Fetch\_Log\_Upon\_Failure:** YES
- FEATURES AND TESTCASES:**
  - Features:** All
  - Stage: stage1**
    - Group: group1**
      - 0\_Add\_NFSetIDList
      - AM\_Concurrency
      - AM\_UDR\_SessRetry\_ServerHeader
      - 1\_Bulk\_Import\_Export
      - AM\_Error\_mapping
      - Asynchronous\_Access\_nCHF

At the bottom, there are buttons for "Build" and "Cancel".

**Note**

ATS display empty list for test cases as it is referring to the **custom** folder for **Policy-Regression** pipeline.

3. Select any of following valid values from the drop-down for **SUT**:
  - PCF
  - CN-PCRF
  - Converged Policy
4. Select any of following valid values from the drop-down for **Configuration\_Type**:
  - **Product\_Config**: On selecting this option, test cases from product folders are populated on ATS UI and product configuration is applied to them via the key-value pair and yaml files defined or present in the "Product Config" folder.
  - **Custom\_Config**: On selecting this option, test cases from custom folders are populated on ATS UI and custom configuration is applied to them via the key-value pair and yaml files defined or present in the "Custom Config" folder. To use the parameterization feature, always select the **Custom\_Config** option. User can copy, add, or delete the required test cases that are available for the Converged Policy, PCF or PCRF and place them appropriately within the **custom** folder for **Policy-NewFeatures**. Reload the page to view the test cases available in the **custom** NewFeatures folder.
5. For the **Select\_Option** field, select any of the following drop-down values:
  - **All**: By default, all the Policy test cases are selected for execution.
  - **Single/MultipleFeatures**: This option allows you to select any number of test cases you want to run from the list of total test cases. Select the check-box for each feature you want to run.

Based on your selection, related **Test Cases** appear on the page.

6. To collect logs for any given build, select YES from the drop-down menu of **Fetch\_Log\_Upon\_Failure**.

- Click **Build** and select **Console Output** to view the test results. The following is a sample test result output:

**Figure 4-54 Policy-Regression Console Output**

```
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnfest/ocnf_steps.py:14571
2023-06-23 09:13:01,012[32m INFO LOG.stage1.group1.STEP:14625 [0m] [32mFetched metrics in 0.26s[0m
2023-06-23 09:13:01,014[32m INFO LOG.stage1.group1.STEP:14631 [0m] [32mPostFetch metrics for policy-ds service are saved to '/tmp/group1-Post-policy-ds'[0m
Then Waiting for next_message
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnfest/pcf_steps.py:2515
2023-06-23 09:13:01,023[32m INFO LOG.stage1.group1.STEP:2532 [0m] [32mWait for 1s[0m
Then Validate policy-ds metric client_response_total containing method="SEARCH",(.*)response="200 OK",(.*)workflow="SSV-GET" is greater_than_or_equal_to 1
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnfest/ocnf_steps.py:14678
2023-06-23 09:13:02,031[32m INFO LOG.stage1.group1.STEP:14711 [0m] [32mPrefetch value for metric client_response_total method="SEARCH",(.*)response="200 OK",(.*)workflow="SSV-GET" : 299[0m
2023-06-23 09:13:02,034[32m INFO LOG.stage1.group1.STEP:14731 [0m] [32mPostFetch value for metric client_response_total method="SEARCH",(.*)response="200 OK",(.*)workflow="SSV-GET" : 300[0m
2023-06-23 09:13:02,035[32m INFO LOG.stage1.group1.STEP:14763 [0m] [32mMetrics incremented successfully![0m
Then Validate policy-ds metric client_response_total containing method="UPDATE",(.*)response="200 OK",(.*)workflow="SSV-UPDATE" is greater_than_or_equal_to 1
../../../../../../../../../../env/lib64/python3.9/site-packages/ocnfest/ocnf_steps.py:14678
```

#### Note

The Regression pipeline does not have any sanity option. However, users must perform all the steps performed in the **Policy-NewFeatures** pipeline. Ensure that the pipeline script is configured according to the environment variables.

#### Note

For more details on consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

## 4.5.7 Policy Regression Documentation

This section describes the documentation for **Policy-Regression** pipeline.

To view the documentation for any of the CNC Policy features, on the ATS home page, click PolicyRegression. Then, click **Documentation** in the left navigation pane.

The following screen appears:



Figure 4-55 Policy-Regression Feature List

[Back to #1](#) pages

My Project

Related Pages

Here is a list of all related documentation pages:

|                                                                                                |  |
|------------------------------------------------------------------------------------------------|--|
| FEATURE - Add_NFSetIDList                                                                      |  |
| FEATURE - Binding Service                                                                      |  |
| FEATURE - Bulk_Import_Export                                                                   |  |
| FEATURE - 3GPP_User_Location_Info_Support                                                      |  |
| FEATURE - AM_Collision_Detection                                                               |  |
| FEATURE - AM_Concurrency                                                                       |  |
| FEATURE - AM_Error_mapping                                                                     |  |
| FEATURE - AM_Late_Arrival                                                                      |  |
| FEATURE - AM_Notify_SessRetry_ODD                                                              |  |
| FEATURE - AM_Policy_Audit                                                                      |  |
| FEATURE - AM Policy UPDATE-NOTIFY-RETRY and TERMINATE-NOTIFY-RETRY                             |  |
| FEATURE - AM_SR_Enable_DNS_Resolution                                                          |  |
| FEATURE - AM_UDR_SessRetry_ServerHeader                                                        |  |
| FEATURE - AM_UpdateNotify_terminate                                                            |  |
| FEATURE - AM Service Refactoring                                                               |  |
| FEATURE - Asynchronous_Access_nCHF                                                             |  |
| FEATURE - Asynchronous_Access_nCHF_Multi_Session                                               |  |
| FEATURE - Asynchronous_Access_nCHF                                                             |  |
| FEATURE - Support for 3GPP NF Sets and Binding in SBI interfaces - AM Service.                 |  |
| FEATURE - Service Name and Scope support in Binding Header towards UDR and CHF in AM flow      |  |
| FEATURE - Support for 3GPP NF Sets and Binding in SBI interfaces in Binding service            |  |
| FEATURE - Support for 3GPP NF Sets and Binding in N5 interface - Policy Authorization Service  |  |
| FEATURE - Support for 3GPP NF Sets and Binding in SBI interfaces in SM service                 |  |
| FEATURE - Service Name and Scope support in Binding Header towards UDR, CHF and AMF in UE flow |  |
| FEATURE - Binding_Service_Audit                                                                |  |

**Example:** On clicking **FEATURE - Binding Service**, the following screen opens:

Figure 4-56 FEATURE - Binding Service

[Back to #1](#) pages 2

My Project

FEATURE - Binding Service

This feature aims to validate the binding service test cases of PCRF.\n

Pre-conditions:

#1.Importing common PGW configurations for all scenarios

#2.Setting the log level to DEBUG

#3.Setting PCRF DataPath

SCENARIO: This test case aims to validate a basic Rx Session Binding with Gx Session using IPv6 Address in a VoIP call.

#1.Pre fetching metrics

#2.Sending Gx CCR-I message and validating CCA-I

#3.Sending Rx AAR-I message and validating AAA-I

#4.Validating RAR after AAR-I is triggered

#5.Sending Rx AAR-U message and validating AAA-U

#6.Validating RAR after AAR-U is triggered

#7. Sending STR message and validating STA

#8.Validating RAR after STR is triggered

This page shows features of only those test cases that are released in previous releases.

## 4.6 Running SCP Test Cases using ATS

This section describes how to run SCP test cases using ATS.

## 4.6.1 Logging in to ATS

### Logging in to ATS GUI in Non-OCI Setup

Before logging in to ATS, ensure that ATS and Stubs are deployed using Helm charts. A sample screenshot is provided as follows:

**Figure 4-57 Verifying ATS Pod**

```
[root@master ocats]# helm status ocats-ocscp -n scpsvc
NAME: ocats-ocscp
LAST DEPLOYED: Thu Aug 31 05:16:22 2023
NAMESPACE: scpsvc
STATUS: deployed
REVISION: 1
TEST SUITE: None
NOTES:
Copyright 2020 (C), Oracle and/or its affiliates. All rights reserved.

Thank you for installing ocats-ocscp.

Your release is named ocats-ocscp , Release Revision: 1.
To learn more about the release, try:

$ helm status ocats-ocscp
$ helm get ocats-ocscp

[root@master ocats]# kubectl get pods -n scpsvc | grep ocats
ocats-ocscp-ocats-scp-778f449f9d-lxfwz 1/1 Running 0 2m52s
[root@master ocats]# kubectl get svc -n scpsvc | grep ocats
ocats-ocscp-ocats-scp LoadBalancer 10.101.29.158 <pending> 8080:30345/TCP 2m56s
```

#### Note

In the Verifying ATS Pod image, the ATS nodeport is highlighted in red as **30345**. For more information about ATS deployment, see [Deploying ATS and Stub in the Kubernetes Cluster](#). Similarly, verify Stub deployments.

To log in to the ATS GUI, open the browser and either provide the external IP address of the worker node and nodeport of the ATS service as <Worker-Node-IP>:<Node-Port-of-ATS> or provide the IP address of the load balancer and serviceport of the ATS service as <ATS-LB-IP>:<Service-Port-of-ATS>.

The ATS login screen appears.

Figure 4-58 Logging in to ATS GUI

**Oracle Communications Cloud Native Core - Automated Test Suite**

Welcome to ATS!

Username  
scpuser

Password  
••••••••

☐ Keep me signed in

Sign in

**Running ATS****Note**

It is recommended to keep the rerun count to minimum 1 for running ATS testcases.

To run ATS:

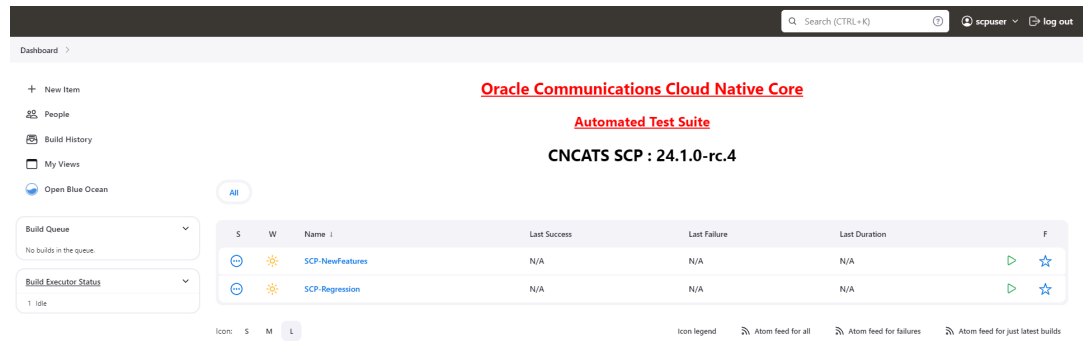
1. Enter the login credentials and click **Sign in**.

**Note**

To modify default login password, see [Modifying Login Password](#).

The following screens appear to display pre-configured pipelines for SCP individually.

Figure 4-59 ATS SCP First Login Screen



The different pre-configured pipelines for SCP are as follows:

- **SCP-NewFeatures:** This pipeline has all the test cases delivered as part of this release.
- **SCP-Regression:** This pipeline covers all the test cases from the previous releases.

### Logging in to ATS GUI in OCI Setup

To access the ATS GUI in OCI, see the [Accessing ATS GUI in OCI](#) section.

### Logging in to ATS GUI with HTTPS Enabled

To access the ATS GUI, see the [Enable ATS GUI with HTTPS](#) section.

#### Note

For more information, see the [Support for Transport Layer Security](#).

## 4.6.2 SCP-NewFeatures Pipeline

This section describes how to configure new feature pipeline parameters, run SCP test cases, and view functionalities.

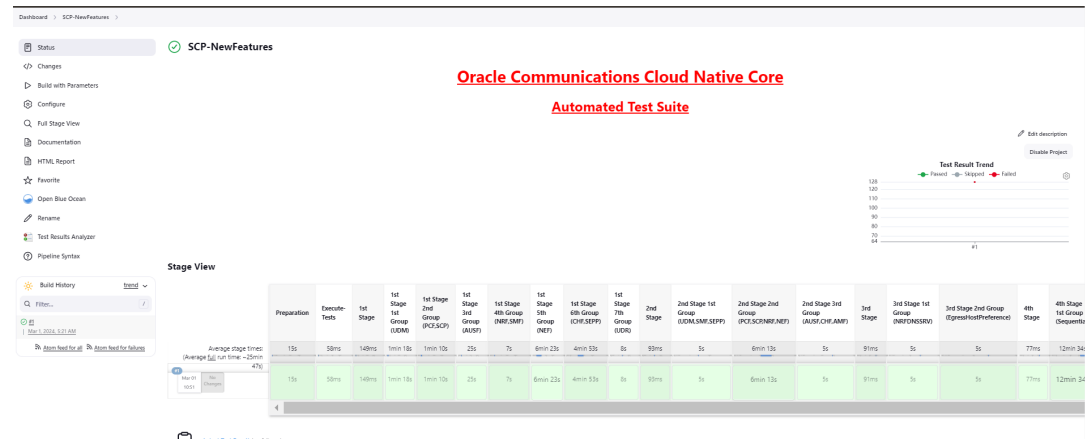
### 4.6.2.1 Configuring New Feature Pipelines

This is a pre-configured pipeline where all the SCP test cases are executed.

To configure its parameters, do the following:

1. Click **SCP-NewFeatures** in the **Name** column.  
The SCP-NewFeatures screen appears.

Figure 4-60 SCP-NewFeatures



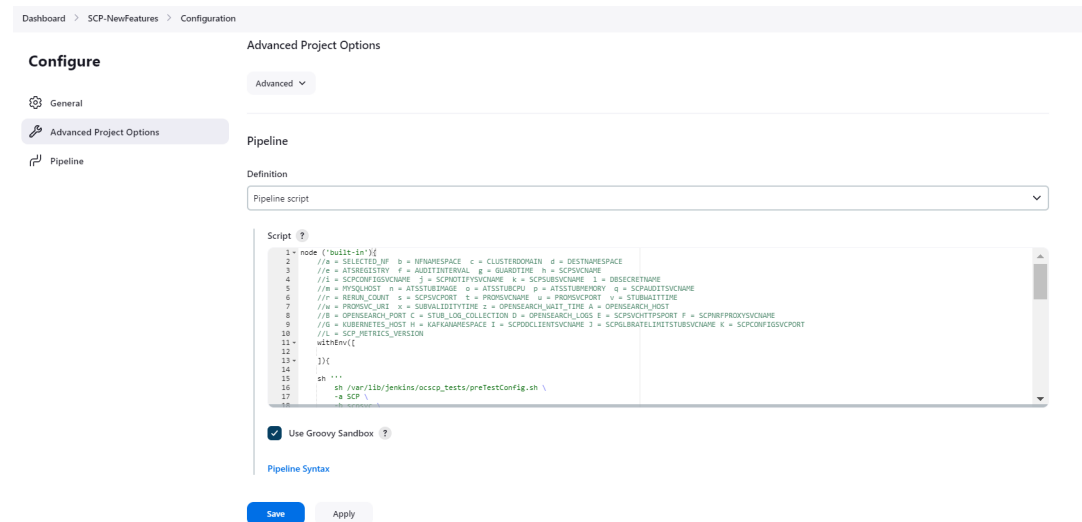
According to the installation guide, the configuration parameters can be modified in the `ocats_ocscp_values.yaml` file before deployment. If you are executing SCP test cases for the first time without enabling the parameters in the deployment file, then you have to set the input parameters before executing any test case. There is no need to set these parameters again unless there is a change in the configuration.

- In the left navigation pane, click **Configure** to provide input parameters and scroll down to the pipeline script as shown in the following image:  
The General tab appears.

#### Note

Ensure that the following screen loads completely before you perform any action on it. Also, do not modify any other configuration than what is specified in the subsequent steps.

The control moves from the **General** tab to the **Pipeline** tab, as shown in the following screenshot:

**Figure 4-61 Pipeline Tab**

3. Modify the values of the pipeline script as required, and then click **Save**. Parameter "-l" must be changed as part of "Pipeline script". The content of the pipeline script is as follows:

```
node ('built-in'){
 //a = SELECTED_NF b = NFNAMESPACE c = CLUSTERDOMAIN d =
DESTNAMESPACE
 //e = ATSREGISTRY f = AUDITINTERVAL g = GUARDTIME h = SCPSVCNAME
 //i = SCPCONFIGSVCSVCNAME j = SCPNOTIFYSVCSVCNAME k = SCPSUBSVCSVCNAME l =
DBSECRETNAME
 //m = MYSQLHOST n = ATSSTUBIMAGE o = ATSSTUBCPU p = ATSSTUBMEMORY
q = SCPAUDITSVCNAME
 //r = RERUN_COUNT s = SCPSVCPORT t = PROMSVCSVCNAME u = PROMSVCSVCPORT v
= STUBWAITTIME
 //w = PROMSVC_URI x = SUBVALIDITYTIME z = OPENSEARCH_WAIT_TIME A =
OPENSEARCH_HOST
 //B = OPENSEARCH_PORT C = STUB_LOG_COLLECTION D = OPENSEARCH_LOGS E =
SCPSVCHTTPSPORT F = SCPNRFPROXYSVCNAME
 //G = KUBERNETES_HOST H = KAFKANAMESPACE I = SCPDDCLIENTSVCSVCNAME J =
SCPGLBRATELIMITSTUBSVCSVCNAME K = SCPCONFIGSVCSVCPORT
 //L = SCP_METRICS_VERSION
 withEnv([

]){
 sh '''
 sh /var/lib/jenkins/ocscp_tests/preTestConfig.sh \
 -a SCP \
 -b scpsvc \
 -c cluster.local \
 -d scpsvc \
 -e cgbu-ocscp-dev-docker.dockerhub-phx.oci.oraclecorp.com/ocats \
 -f 60 \
 -g 10 \
 -h ocscp-scp-worker \
 -i ocscp-scp-configuration \
 -j ocscp-scp-notification \
```

```

-k ocscp-scpc-subscription \
-l appuser-secret \
-m mysql.default.svc.cluster.local \
-n ocats-pystub:tag \
-o 0.5 \
-p 0.5G \
-q ocscp-scpc-audit \
-r 2 \
-s 8000 \
-t occne-prometheus-server.occne-infra \
-u 80 \
-v 300 \
-w /api/ \
-x 120 \
-z 0 \
-A occne-opensearch-cluster-master.occne-infra \
-B 9200 \
-C no \
-D opensearch \
-E 9443 \
-F ocscp-scp-nrfproxy \
-G kubernetes.default \
-H scpsvc \
-I ocats-ocscp-ocats-ddclientstub \
-J ocats-ocscp-ocats-scpglbratelimitstub \
-K 8081 \
-L v1 \
...
if(env.Include_Regression && "${Include_Regression}" == "YES"){
 sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocscp_tests/jenkinsData/Jenkinsfile-Merged"
}
else{
 load "/var/lib/jenkins/ocscp_tests/jenkinsData/Jenkinsfile-
NewFeatures"
}
}
}

```

The description of these parameters is as follows:

#### Note

Do not modify anything other than these parameters.

- **a:** Selected NF
- **b:** NameSpace in which SCP is deployed
- **c:** K8s Cluster Domain where SCP is deployed
- **d:** Test Stubs NameSpace - **Must be same as SCP Namespace**
- **e:** Docker registry where test stub image is available

- **f:** Audit Interval provided in SCP Deployment file
- **g:** Guard Time provided SCP Deployment file
- **h:** SCP-Worker microservice name as provided during deployment
- **i:** SCPC-Configuration microservice name as provided during deployment
- **j:** SCPC-Notification microservice name as provided during deployment
- **k:** SCPC-Subscription microservice name as provided during deployment
- **l:** DB Secret name as provided during deployment
- **m:** Mysql Host name as provided during deployment
- **n:** Test Stub Image Name with tag
- **o:** Test Stub CPU requests and limit
- **p:** Test Stub Memory requests and limit
- **q:** SCPC-Audit microservice name as provided during deployment
- **r:** re-run count
- **s:** SCPSVCPORT with which SCP is deployed
- **t:** PROMSVCNAME with which Prometheus server is deployed. **Note:** Prometheus service name is followed by namespace in which Prometheus service is existing. Prometheus service is in a different namespace related to deployed SCP namespace.
- **u:** PROMSVCPORT with which Prometheus server is deployed
- **v:** Time to wait for Stub Creation
- **w:** API URI of Prometheus (/api/) - For examples, use **/api/** or **/prometheus/api/**
- **x:** Subscription Validity Time to set by ATS in response to subscription request from SCP
- **y:** Application Debug Logs collection on Rerun if set to true. The default value is false
- **z:** Opensearch connection timeout
- **A:** Opensearch host/service name
- **B:** Opensearch port
- **C:** Stub log collection
- **D:** Opensearch API Path Prefix
- **E:** SCP HTTPS Signaling Port
- **F:** SCP NRF Proxy service name
- **G:** Kubernetes Host
- **I:** scpddclientstub name as provided during deployment
- **J:** scpglobalratelimitstub name as provided during deployment
- **K:** port of configuration pod in SCP deployment
- **L:** scp metrics version

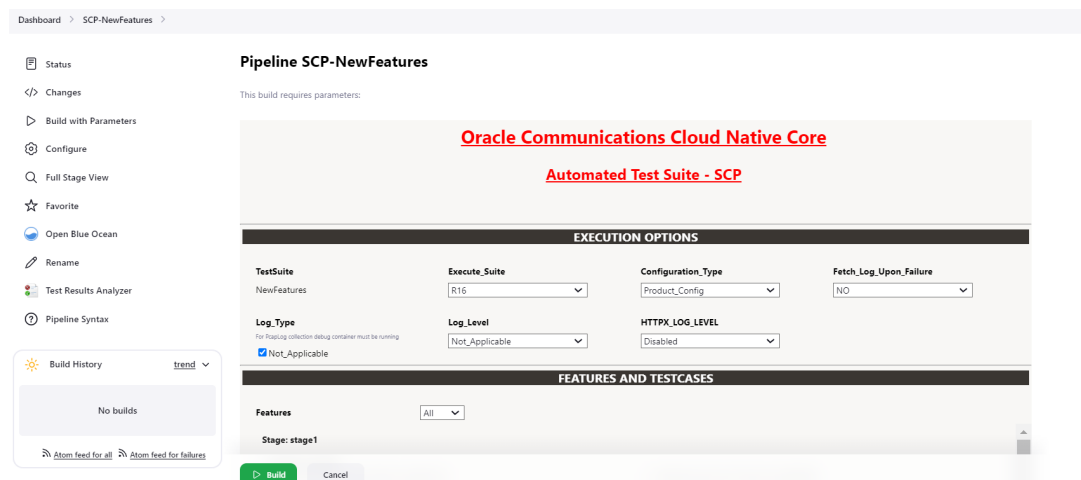
### 4.6.2.2 Running SCP New Feature Pipelines

To run SCP test cases, perform the following steps



1. Click **Build with Parameters** link available in the left navigation pane of the Pipeline SCP-NewFeatures screen.
2. Select the required option from the **Execute\_Suite** drop-down menu to run Suite for R16 or R15 test cases:
  - **R15**: To execute Release 15 test cases for SCP.
  - **R16**: To execute Release 16 test cases for SCP.
3. Select the required option from the **Features** drop-down menu to run the test cases:
  - **All**: To execute all the test cases for SCP based on Release15 or Release16 selection.
  - **Sanity**: Enabled for SCP. Sanity cases are run to do a quick sanity check on SCP SUT features.
  - **Select**: This option allows users to individually select specific features for separate execution. Additionally, users can select the run for an entire stage or group instead of running all feature files.
4. Select one of the following configuration types from the **Configuration\_Type** drop-down menu:
  - **Product\_Config**: To run test cases from the features or regression directory. Test cases in this directory should not be modified.
  - **Custom\_Config**: To run test cases from the features or cust\_regression directory. Any test case customization required should be done in this directory.
5. Select an appropriate option and click **Build** to start test execution.

Figure 4-62 Build



6. Click **Full Stage View** in the left navigation pane to view the status of the run.
7. Hover over values of **Execute Tests** and then click **Logs** to check execution results and logs.

#### Note

A sample consolidated test report is shown in the following image when rerun is set to 2, which is the default value. All the features passed within the first rerun, therefore, the second rerun stage has not been run in this case.

**Figure 4-63 Sample consolidated report**

```
Initial Run :-
Features RUN 21, PASS 20, FAIL 1

1st Rerun:-
Features RUN 1, PASS 1, FAIL 0

Final Result:-
Total-Features RUN 21, PASS 21, FAIL 0
```

For more information about consolidated test report, see [Managing Final Summary Report, Build Color, and Application Log](#).

**Note**

- The New Feature tests in this release have been written according to SCP-ATS parameterization method (Golden Configuration). The individual features are not populated with direct values for the parameters, however, with variable names. While performing the tests, these variables are mapped according to the data provided adjacent to them in the feature specific `<feature_name>.yaml` and `global.yaml` that can be found in the `product_config` directory. This method allows the decoupling of data that can be customer specific from the features. For more information about parameterization (Golden Configuration), see [Parameterization](#).

To view the detail view of the respective group, rerun logs, and the consolidated report summary:

- a. Click **Open Blue Ocean Homepage** from the left navigation pane.
- b. Click the build number in the run column to view the consolidated output of all the feature that are run.
- c. Click **Download icon** to download the log files.

### 4.6.2.3 SCP-NewFeatures Documentation

This pipeline has an HTML report of all the feature files that you can test as part of the SCP ATS release.

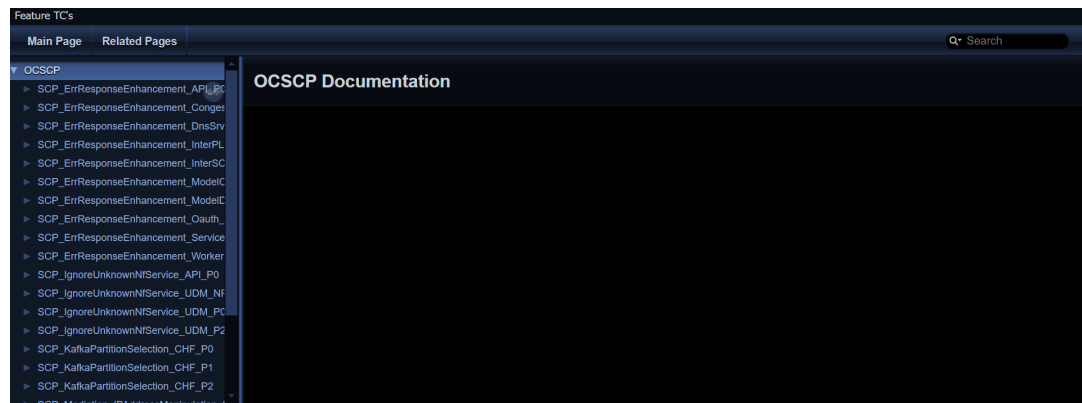
1. To view SCP functionalities, go to the **SCP-NewFeatures** pipeline and click the **Documentation** option in the left navigation pane. It lists all the test cases provided as part of SCP ATS -24.1.1.

**Note**

The **Documentation** option appears only if the **SCP-NewFeatures** pipeline test cases are executed at least once.

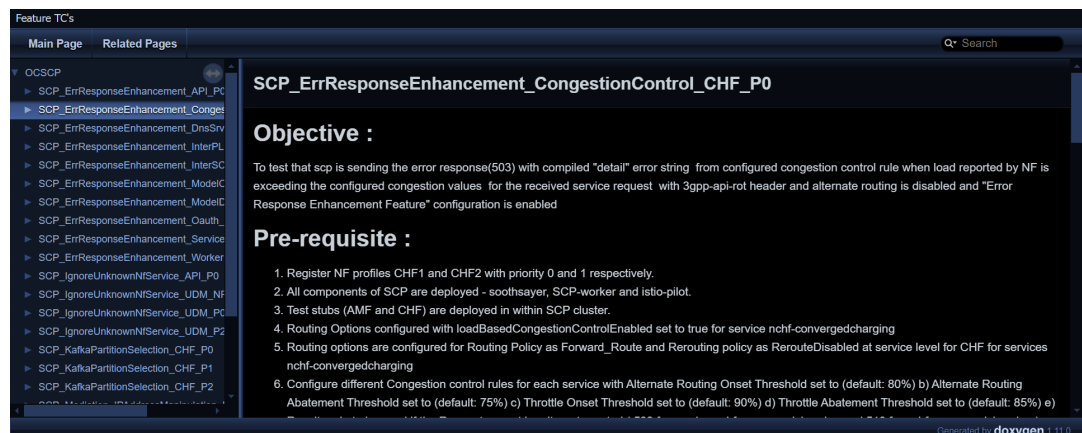
The following screen appears:

**Figure 4-64 SCP-NewFeatures Documentation**



2. Click any functionality to view its test cases and scenarios for each test case.

**Figure 4-65 Sample Scenario**



Based on the functionalities covered under Documentation, the screen displays test cases.

3. To navigate back to the Pipeline SCP-NewFeatures screen, click the **Back to SCP-NewFeatures** link available in the upper left corner of the screen.

## 4.6.3 SCP-Regression Pipeline

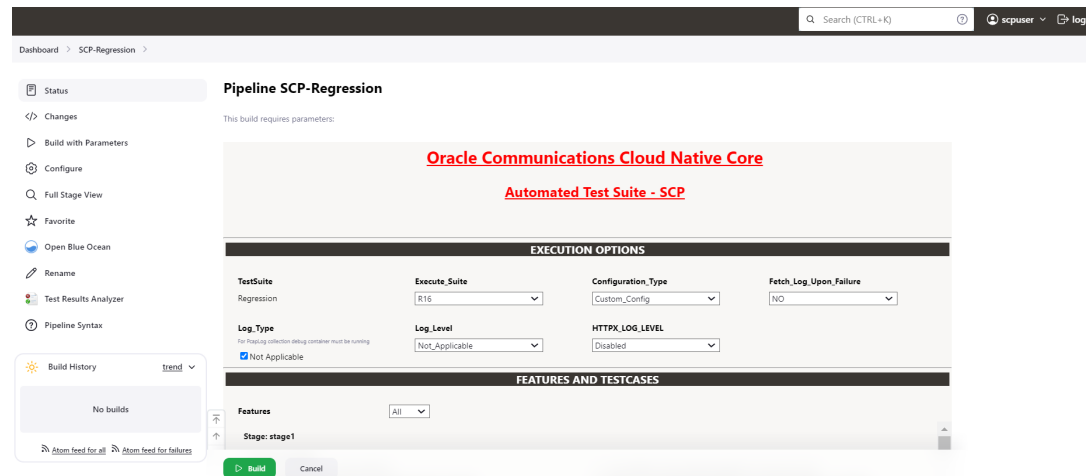
This section describes how to configure regression pipeline parameters, run SCP test cases, and view functionalities.

### 4.6.3.1 Configuring Regression Pipelines

This pre-configured pipeline has all the test cases from previous releases.

1. Click **SCP-Regression Pipeline**:

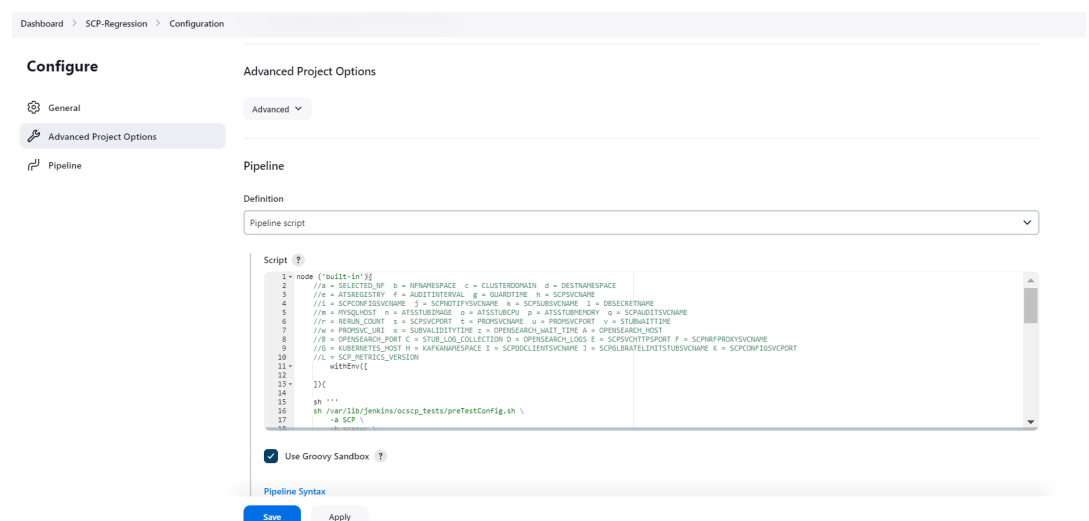
**Figure 4-66 SCP-Regression Pipeline**



According to the installation guide, the configuration parameters can be modified in the `ocats_ocscp_values.yaml` file before deployment. If you are executing SCP-Regression pipeline test cases for the first time without enabling the parameters in the deployment file, you have to set the input parameters before execution. Subsequent execution does not require any input unless there is a need to change any configuration.

2. In the left navigation pane, click **Configure** to provide input parameters and scroll-down to the pipeline script as shown in the following image:

**Figure 4-67 Regression - Pipeline Script**



3. Modify the values of the Pipeline script as required, and then click **Save**. Parameter "-I" should be changed as part of "Pipeline script". The content of the pipeline script is as follows:

```
node ('built-in'){
 //a = SELECTED_NF b = NFNAMESPACE c = CLUSTERDOMAIN d =
DESTNAMESPACE
 //e = ATSREGISTRY f = AUDITINTERVAL g = GUARDTIME h = SCPSVCNAME
 //i = SCPCONFIGSVCSVCNAME j = SCPNOTIFYSVCSVCNAME k = SCPSUBSVCSVCNAME l =
DBSECRETNAME
 //m = MYSQLHOST n = ATSSTUBIMAGE o = ATSSTUBCPU p = ATSSTUBMEMORY
q = SCPAUDITSVCNAME
 //r = RERUN_COUNT s = SCPSVCPORT t = PROMSVCSVCNAME u = PROMSVCSVCPORT v
= STUBWAITTIME
 //w = PROMSVC_URI x = SUBVALIDITYTIME z = OPENSEARCH_WAIT_TIME A =
OPENSEARCH_HOST
 //B = OPENSEARCH_PORT C = STUB_LOG_COLLECTION D = OPENSEARCH_LOGS E =
SCPSVCHTTPSPORT F = SCPNRFPROXYSVCNAME
 //G = KUBERNETES_HOST H = KAFKANAMESPACE I = SCPDDCLIENTSVCSVCNAME J =
SCPGLBRATELIMITSTUBSVCSVCNAME K = SCPCONFIGSVCSVCPORT
 //L = SCP_METRICS_VERSION
 withEnv([

]){

sh '''
sh /var/lib/jenkins/ocscp_tests/preTestConfig.sh \
 -a SCP \
 -b scpsvc \
 -c cluster.local \
 -d scpsvc \
 -e cgbu-ocscp-dev-docker.dockerhub-phx.oc1.oraclecorp.com/ocats \
 -f 60 \
 -g 10 \
 -h ocscp-scp-worker \
 -i ocscp-scp-configuration \
 -j ocscp-scp-notification \
 -k ocscp-scp-subscription \
 -l appuser-secret \
 -m mysql.default.svc.cluster.local \
 -n ocats-pystub:tag \
 -o 0.5 \
 -p 0.5G \
 -q ocscp-scp-audit \
 -r 0 \
 -s 8000 \
 -t occne-prometheus-server.occne-infra \
 -u 80 \
 -v 300 \
 -w /api/ \
 -x 120 \
 -z 0 \
 -A occne-opensearch-cluster-master.occne-infra \
 -B 9200 \
 -C no \
 -D opensearch \
```

```

 -E 9443 \
 -F ocscp-scp-nrfproxy \
 -G kubernetes.default \
 -H scpsvc \
 -I ocats-ocscp-ocats-ddclientstub \
 -J ocats-ocscp-ocats-scpoglbratelimitstub \
 -K 8081 \
 -L v1 \
 ...
 if(env.Include_NewFeatures && "${Include_NewFeatures}" == "YES"){
 sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
 load "/var/lib/jenkins/ocscp_tests/jenkinsData/Jenkinsfile-Merged"
 }
 else{
 load "/var/lib/jenkins/ocscp_tests/jenkinsData/Jenkinsfile-
Regression"
 }

}
}

```

The description of these parameters is as follows:

- **a:** Selected NF
- **b:** NameSpace in which SCP is deployed
- **c:** K8s Cluster Domain where SCP is deployed
- **d:** Test Stubs NameSpace - **Must be same as SCP Namespace**
- **e:** Docker registry where test stub image is available
- **f:** Audit Interval provided in SCP Deployment file
- **g:** Guard Time provided SCP Deployment file
- **h:** SCP-Worker microservice name as provided during deployment
- **i:** SCPC-Configuration microservice name as provided during deployment
- **j:** SCPC-Notification microservice name as provided during deployment
- **k:** SCPC-Subscription microservice name as provided during deployment
- **l:** DB Secret name as provided during deployment
- **m:** Mysql Host name as provided during deployment
- **n:** Test Stub Image Name with tag
- **o:** Test Stub CPU requests and limit
- **p:** Test Stub Memory requests and limit
- **q:** SCPC-Audit microservice name as provided during deployment
- **r:** re-run count
- **s:** SCPSVCPORT with which SCP is deployed
- **t:** PROMSVCTYPENAME with which prometheus server is deployed. **Note:** Prometheus service name is followed by namespace in which Prometheus service is existing. Prometheus service is in a different namespace related to deployed SCP namespace.
- **u:** PROMSVCTYPEPORT with which prometheus server is deployed

- **v:** Time to wait for Stub Creation
- **w:** API URI of Prometheus (/api/) - Use **"/api/"** or **"/prometheus/api/"**
- **x:** Subscription Validity Time to set by ATS in response to subscription request from SCP
- **y:** Application Debug Logs collection on Rerun if set to true. The default value is false
- **z:** Opensearch connection timeout
- **A:** Opensearch host/service name
- **B:** Opensearch port
- **C:** Stub log collection
- **D:** Opensearch API Path Prefix
- **E:** SCP HTTPS Signaling Port
- **F:** SCP NRF Proxy service name
- **G:** Kubernetes Host
- **I:** scpddclientstub name as provided during deployment
- **J:** scpglobalratelimitstub name as provided during deployment
- **K:** port of configuration pod in scp deployment
- **L:** scp metrics version

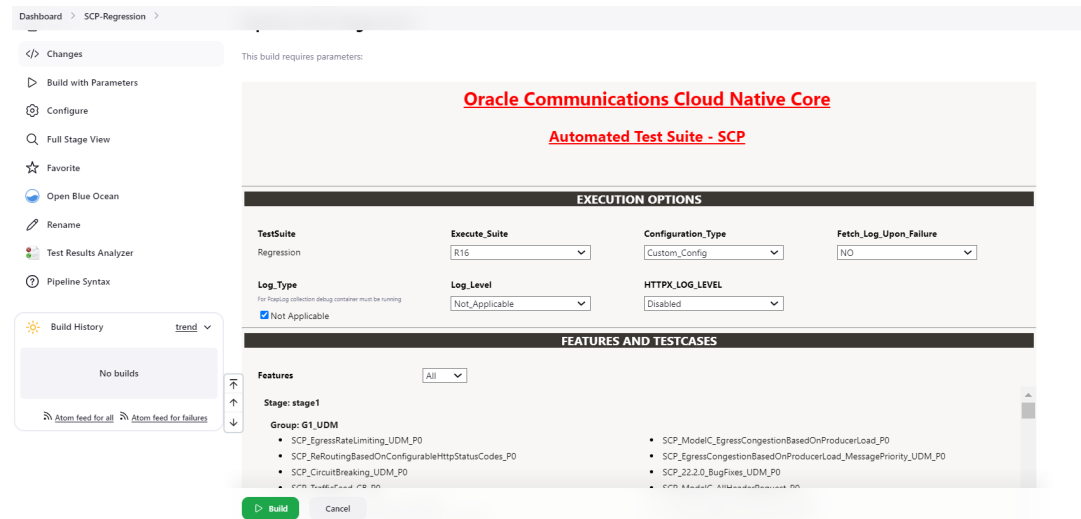
### 4.6.3.2 Running Regression Pipelines

This section describes how to run test cases for SCP Regression pipeline.

1. To run the pipeline, click **Build with Parameters** in the left navigation pane.
2. Select the required option from the **Execute\_Suite** drop-down menu to run Suite for R16 or R15 test cases:
  - **R15:** To run Release 15 test cases for SCP.
  - **R16:** To run Release 16 test cases for SCP.
3. Select the required regression option from the **Features** drop-down menu to run the test cases:
  - **All:** To run all the test cases for SCP based on Release15 or Release16 selection. This runs all the cases except SCP\_Audit\_nnrfdisc. If SCP is deployed with nnrfdisc for audit or registration with NRF is disabled, then the **All** option should not be used and instead Single or MultipleFeatures option can be used to select appropriate cases for execution.
  - **Sanity:** Enabled for SCP. Sanity cases are run to do a quick sanity check on SCP SUT features.
  - **Select:** This option allows users to individually select specific features for separate execution. Additionally, users can select the run for an entire stage or group instead of running all feature files.
4. Select one of the following configuration types from the **Configuration\_Type** drop-down menu:
  - **Product\_Config:** To run test cases from the features or regression directory. Test cases in this directory should not be modified.

- **Custom\_Config:** To run test cases from the features or cust\_regression directory. Any test case customization required should be done in this directory.
5. Select an appropriate option and click **Build** to start test execution.

Figure 4-68 Build



6. Click **Full Stage View** in the left navigation pane to view the status of the run.
7. Hover over the **Execute-Tests** stage of the pipeline, and then click **Logs** to check execution results and logs.  
To view the detail view of the respective group, rerun logs, and the consolidated report summary:
  - a. Click **Open Blue Ocean Homepage** from the left navigation pane.
  - b. Click the build number in the run column to view the consolidated output of all the feature that are run.
  - c. Click **Download icon** to download the log files.

### 4.6.3.3 SCP-Regression Documentation

This pipeline has an HTML report of all the feature files that you can test as part of the SCP ATS release.

1. To view SCP functionalities, go to the **SCP-Regression** pipeline and click the **Documentation** option in the left navigation pane. It lists all the test cases provided as part of SCP ATS -24.1.1.

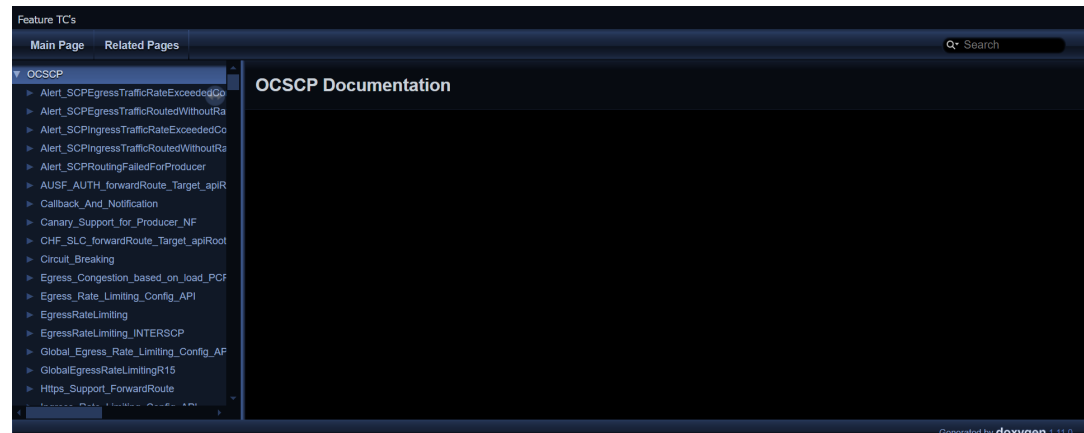
#### Note

The **Documentation** option appears only if the **SCP-Regression** pipeline test cases are executed at least once.  
In SCP 24.1.0, the feature files are renamed as per the naming convention. For more information, see [Renaming of Regression Feature Files](#)

The following screen appears:

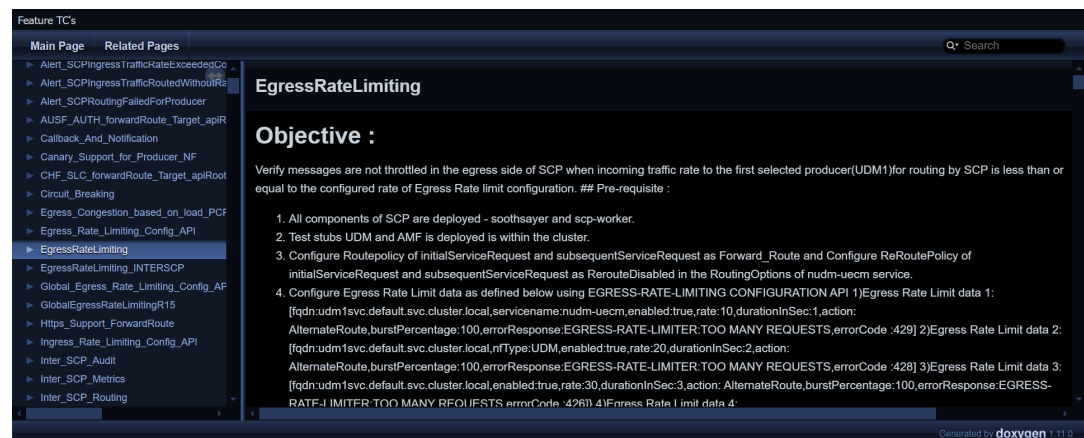


Figure 4-69 SCP-Regression Documentation



2. Click any functionality to view its test cases and scenarios for each test case.

Figure 4-70 Sample Scenario



Based on the functionalities covered under Documentation, the screen displays test cases.

3. To navigate back to the Pipeline SCP-Regression screen, click the **Back to SCP-Regression** link available in the upper left corner of the screen.

#### 4.6.3.3.1 Renaming of Regression Feature Files

The following tables detail the feature file that has been renamed in alignment with the naming convention:

Table 4-7 Regression Feature File

| Feature Names prior to SCP 24.1.0                      | New Feature Names for SCP 24.1.0                      |
|--------------------------------------------------------|-------------------------------------------------------|
| Route_group_static_config_api_validation               | SCP_AlternateRoutingUsingStaticConfig_API_P0          |
| Alternate_Routing_Using_Static_Config_EgressRL_SMF     | SCP_AlternateRoutingUsingStaticConfig_EgressRL_SMF_P0 |
| Alternate_Routing_Using_Static_Configuration_Https_SMF | SCP_AlternateRoutingUsingStaticConfig_HTTPS_SMF_P0    |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                          | New Feature Names for SCP 24.1.0                                            |
|----------------------------------------------------------------------------|-----------------------------------------------------------------------------|
| Alternate_Routing_Using_Static_Configuration_Notification_CircuitBreaking. | SCP_AlternateRoutingUsingStaticConfig_NotificationMessageCircuitBreaking_P0 |
| Alternate_Routing_Using_Static_Configuration_SMF                           | SCP_AlternateRoutingUsingStaticConfig_SMF_P0                                |
| Alternate_Routing_Using_Static_Config_Overload_Control                     | SCP_AlternateRoutingUsingStaticConfig_OverloadControl_P0                    |
| SCP_Audit_nnrf_disc_SMF                                                    | SCP_Audit_nnrf-disc_SMF_P0                                                  |
| SCP_Audit_nnrf_nfm_SMF                                                     | SCP_Audit_nnrf-nfm_SMF_P0                                                   |
| SCP_Audit_nnrf_nfm_SMF_user_agent                                          | SCP_Audit_nnrfnfmSCPsUserAgentHeaderValidation_P0                           |
| Dns_Srv_Call_Back_Notification_CD_OD_Cases                                 | SCP_CallbackNotification_DNSSRVbasedCBOD_P0                                 |
| Callback_And_Notification                                                  | SCP_CallbackNotification_P0                                                 |
| SCP_CCA_Header_Validation_API_P0                                           | SCP_CCAvalidation_API_P0                                                    |
| CCA_validation_AUSF_P0                                                     | SCP_CCAvalidation_AUSF_P0                                                   |
| CCA_validation_AUSF_P1                                                     | SCP_CCAvalidation_AUSF_P1                                                   |
| CCA_validation_InterPLMN_CHF_P0                                            | SCP_CCAvalidation_InterPLMNRouting_CHF_P0                                   |
| CCA_validation_InterSCP_CHF_P0                                             | SCP_CCAvalidation_InterSCP_CHF_P0                                           |
| CCA_validation_ModelID_CHF_P0                                              | SCP_CCAvalidation_ModelID_CHF_P0                                            |
| CCA_validation_ModelID_CHF_P1                                              | SCP_CCAvalidation_ModelID_CHF_P1                                            |
| SCP_Circuit_Breaking_Config_API                                            | SCP_CircuitBreaking_ConfigAPI_P0                                            |
| Circuit_Breaking_PCF                                                       | SCP_CircuitBreaking_PCF_P0                                                  |
| Circuit_Breaking_SMF                                                       | SCP_CircuitBreaking_SMF_P0                                                  |
| SCP_Congestion_Control_Config_API                                          | SCP_CongestionControl_ConfigAPI_P0                                          |
| S_Correlation_Header_interplmn_model_C                                     | SCP_CorrelationHeader_InterPLMNModelC_P0                                    |
| Correlation_Headersepp_model_D_Inter_PLMN_and_Mediation                    | SCP_CorrelationHeader_InterPLMNRoutingBasedOnModelDandMediation_P0          |
| Interscp_Correlation_Header_model_C                                        | SCP_CorrelationHeader_InterSCP_P0                                           |
| Correlation_Header_model_C_cases                                           | SCP_CorrelationHeader_ModelC_P0                                             |
| Notification_Correlation_Header_model_C                                    | SCP_CorrelationHeader_ModelCNotificationRequest_P0                          |
| Correlation_Header_SCP_generated_error                                     | SCP_CorrelationHeader_SCPGeneratedError_P0                                  |
| DefaultNotificationCallbackInitialRouteAndDiscovery_p0                     | SCP_DefaultNotificationCallback_InitialRouteAndDiscovery_P0                 |
| Dns_Srv_Call_Back_Notification_Https                                       | SCP_DefaultNotificationCallbackUri_DNSSRV_HTTPS_P0                          |
| Dns_Srv_Call_Back_Notification                                             | SCP_DefaultNotificationCallbackUri_DNSSRV_P0                                |
| DefaultNotificationCallbackUriModelC_Https_p1                              | SCP_DefaultNotificationCallbackUri_ModelC_HTTPS_P1                          |
| DefaultNotificationCallbackUriModelC_p0                                    | SCP_DefaultNotificationCallbackUri_ModelC_P0                                |
| DefaultNotificationCallbackUriModelC_p1                                    | SCP_DefaultNotificationCallbackUri_ModelC_P1                                |
| DefaultNotificationCallbackUriModelC_CB_p0                                 | SCP_DefaultNotificationCallbackUri_ModelCCB_P0                              |
| DefaultNotificationCallbackUriModelC_SEPP_p1                               | SCP_DefaultNotificationCallbackUri_ModelCInterPLMNRouting_P1                |
| DefaultNotificationCallbackUriModelC_SEPP_p1                               | SCP_DefaultNotificationCallbackUri_ModelCInterPLMNRouting_P1                |
| Alternate_resolution_api_validation                                        | SCP_DNSSRV_AlternateResolution_API_P0                                       |
| Dns_Srv_Based_Routing_For_ModelC-Headers_PCF_cases                         | SCP_DNSSRV_ModelCHeaders_PCF_P0                                             |
| Dns_Srv_Based_Routing_For_ModelC-Headers_UDM_cases                         | SCP_DNSSRV_ModelCHeaders_UDM_P0                                             |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                        | New Feature Names for SCP 24.1.0                                           |
|--------------------------------------------------------------------------|----------------------------------------------------------------------------|
| Dns_Srv_Producer_Overload_Control                                        | SCP_DNSSRV_ProducerBasedOverloadControl_P0                                 |
| Alert_egress_congestion_based_on_producer_load_and_message_priority_CHF  | SCP_EgressCongestionBasedOnProducerLoad_AlertMessagePriority_CHF_P0        |
| Egress_congestion_based_on_producer_load_and_message_priority_PCF        | SCP_EgressCongestionBasedOnProducerLoad_MessagePriority_PCF_P0             |
| Egress_congestion_based_on_producer_load_and_message_priority_UDM        | SCP_EgressCongestionBasedOnProducerLoad_MessagePriority_UDM_P0             |
| SCP_Egress_Host_Preference_alternate_routing_SMF_P0_P1                   | SCP_EgressHostPreference_AlternateRouting_SMF_P1                           |
| SCP_Egress_Host_Preference_alternate_routing_withoutHeader_AUSF_P1       | SCP_EgressHostPreference_AlternateRoutingWithoutHeader_AUSF_P1             |
| SCP_Egress_Host_Preference_API_P0                                        | SCP_EgressHostPreference_API_P0                                            |
| SCP_Egress_Host_Preference_CHF_P0                                        | SCP_EgressHostPreference_CHF_P0                                            |
| SCP_Egress_Host_Preference_CHF_P1                                        | SCP_EgressHostPreference_CHF_P1                                            |
| SCP_Egress_Host_Preference_InterPLMN_Routing_P0                          | SCP_EgressHostPreference_InterPLMNRouting_P0                               |
| SCP_Egress_Host_Preference_InterPLMN_Routing_P1                          | SCP_EgressHostPreference_InterPLMNRouting_P1                               |
| SCP_Egress_Host_Preference_InterPLMN_Routing_Non_ASM_P0                  | SCP_EgressHostPreference_InterPLMNRoutingNonASM_P0                         |
| SCP_Egress_Host_Preference_InterSCP_Routing_CHF_P0                       | SCP_EgressHostPreference_InterSCPRouting_CHF_P0                            |
| SCP_Egress_Host_Preference_InterSCP_Routing_CHF_P1                       | SCP_EgressHostPreference_InterSCPRouting_CHF_P1                            |
| SCP_Egress_Host_Preference_InterSCP_Routing_CHF_Non_ASM_P0               | SCP_EgressHostPreference_InterSCPRoutingNonASM_CHF_P0                      |
| SCP_Egress_Host_Preference_ModelID_routing_CHF_NRF_P0                    | SCP_EgressHostPreference_ModelIDRoutingCHF_NRF_P0                          |
| SCP_Egress_Host_Preference_ModelID_routing_CHF_NRF_P1                    | SCP_EgressHostPreference_ModelIDRoutingCHF_NRF_P1                          |
| SCP_Egress_Host_Preference_ModelID_routing_without_apiRootHdr_CHF_NRF_P1 | SCP_EgressHostPreference_ModelIDRoutingWithoutAPIRootHdrCHFandNRF_P1       |
| SCP_Egress_Host_Preference_Notification_PCF                              | SCP_EgressHostPreference_NotificationRequest_PCF_P0                        |
| SCP_Egress_Host_Preference_SCP_Generated_Audit_SMF_NRF_P1                | SCP_EgressHostPreference_SCPGeneratedAuditSMF_NRF_P1                       |
| SCP_Egress_Host_Preference_SMF_P0_P1                                     | SCP_EgressHostPreference_SMF_P1                                            |
| SCP_Egress_Host_Preference_SCP_Generated_Subscription_SCP_NRF_P1         | SCP_EgressHostPreference_SCPgeneratedSubscription_NRF_P1                   |
| Alert_SCPEgressTrafficRateExceededConfiguredLimit_AUSF                   | SCP_EgressRateLimiting_AlertTrafficRateExceededConfiguredLimit_AUSF_P0     |
| Alert_SCPEgressTrafficRoutedWithoutRateLimitTreatment_AUSF               | SCP_EgressRateLimiting_AlertTrafficRoutedWithoutRateLimitTreatment_AUSF_P0 |
| Egress_Rate_Limiting_Config_API                                          | SCP_EgressRateLimiting_Config_API_P0                                       |
| EgressRateLimiting_INTERSCP                                              | SCP_EgressRateLimiting_InterSCP_P0                                         |
| EgressRateLimiting                                                       | SCP_EgressRateLimiting_P0                                                  |
| EgressRateLimiting_UDM_Cases                                             | SCP_EgressRateLimiting_UDM_P0                                              |
| RateLimitingRelease16_AUSF                                               | SCP_EgressRateLimitingRelease16_AUSF_P0                                    |
| SCP_EgressRateLimiting_DNSSRV.feature                                    | SCP_EgressRateLimiting_DNSSRV_P0.feature                                   |
| Enhanced_Suspended_State_Routing_API_Validation                          | SCP_EnhancedSuspendedStateRouting_API_P0                                   |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                                       | New Feature Names for SCP 24.1.0                                             |
|-----------------------------------------------------------------------------------------|------------------------------------------------------------------------------|
| Enhanced_Suspended_State_Routing_Alternate_Routing_Using_Static_Config_DNS_SRV          | SCP_EnhancedSuspendedStateRouting_ARusingStaticConfigandDNSSRV_P0            |
| Enhanced_Suspended_State_Routing_Inter_PLMN                                             | SCP_EnhancedSuspendedStateRouting_InterPLMNRouting_P0                        |
| Enhanced_Suspended_State_Routing_Mode2_API_Validation_P0                                | SCP_EnhancedSuspendedStateRouting_Mode2_API_P0                               |
| Enhanced_Suspended_State_Routing_Mode2_Alternate_Routing_Using_Static_Config_DNS_SRV_P0 | SCP_EnhancedSuspendedStateRouting_Mode2AlternateRoutingStaticConfigDNSSRV_P0 |
| Enhanced_Suspended_State_Routing_Mode2_ModelC_SMF_P1                                    | SCP_EnhancedSuspendedStateRouting_Mode2ModelC_SMF_P1                         |
| Enhanced_Suspended_State_Routing_Mode2_ModelC_Inter_PLMN_P0                             | SCP_EnhancedSuspendedStateRouting_Mode2ModelCInterPLMN_P0                    |
| Enhanced_Suspended_State_Routing_Mode2_ModelC_InterSCP_P0                               | SCP_EnhancedSuspendedStateRouting_Mode2ModelCInterSCP_P0                     |
| Enhanced_Suspended_State_Routing_Mode2_ModelC_Overload_Congestion_SMF_P1                | SCP_EnhancedSuspendedStateRouting_Mode2ModelCOverloadCongestion_SMF_P1       |
| Enhanced_Suspended_State_Routing_Mode2_ModelID_P0                                       | SCP_EnhancedSuspendedStateRouting_Mode2ModelID_P0                            |
| Enhanced_Suspended_State_Routing_Mode2_ModelID_Inter_PLMN_P0                            | SCP_EnhancedSuspendedStateRouting_Mode2ModelIDInterPLMNRouting_P0            |
| Enhanced_Suspended_State_Routing_Mode2_ModelID_InterSCP_P0                              | SCP_EnhancedSuspendedStateRouting_Mode2ModelIDInterSCP_P0                    |
| Enhanced_Suspended_State_Routing_ModelC_SMF                                             | SCP_EnhancedSuspendedStateRouting_ModelC_SMF_P0                              |
| Enhanced_Suspended_State_Routing_Parallel_ModelC_SMF_UDM_P0                             | SCP_EnhancedSuspendedStateRouting_ModelC_SMFandUDM_P0                        |
| Enhanced_Suspended_State_Routing_ModelC_UDM                                             | SCP_EnhancedSuspendedStateRouting_ModelC_UDM_P0                              |
| Enhanced_Suspended_State_Routing_Model_C_Inter_SCP                                      | SCP_EnhancedSuspendedStateRouting_ModelCInterSCP_P0                          |
| Enhanced_Suspended_State_Routing_ModelID                                                | SCP_EnhancedSuspendedStateRouting_ModelID_P0                                 |
| Enhanced_Suspended_State_Routing_ModelID_InterSCP                                       | SCP_EnhancedSuspendedStateRouting_ModelIDInterSCP_P0                         |
| Enhanced_NF_Status_Processing_SUSPENDED                                                 | SCP_EnhancedSuspendedStateRouting_SuspendedMode2_P0                          |
| Enhanced_NF_Status_Processing_SUSPENDED_SMF                                             | SCP_EnhancedSuspendedStateRouting_SuspendedMode2_SMF_P0                      |
| Enhanced_NF_Status_Processing_UNDISCOVERABLE                                            | SCP_EnhancedSuspendedStateRouting_UNDISCOVERABLE_P0                          |
| SCP_Error_Profile_Config_API_P0                                                         | SCP_ErrorProfile_ConfigAPI_P0                                                |
| SCPFailureHandling_default_behavior_serverheader_AUSF                                   | SCP_FailureHandling_DefaultBehaviorServerHeader_AUSF_P0                      |
| SCPFailureHandling_default_behavior_serverheader                                        | SCP_FailureHandling_DefaultBehaviorServerHeader_P0                           |
| SCPFailureHandling_DNSSRV_based_EnhancedServerHeaderV2_NEF_p0                           | SCP_FailureHandling_DNSSRVbasedEnhancedServerHeaderV2_NEF_P0                 |
| SCPFailureHandling_EnhancedServerHeader                                                 | SCP_FailureHandling_EnhancedServerHeader_P0                                  |
| SCPFailureHandling_EnhancedServerHeaderV2_API                                           | SCP_FailureHandling_EnhancedServerHeaderV2_API_P0                            |
| SCPFailureHandling_EnhancedServerHeaderV2_p0                                            | SCP_FailureHandling_EnhancedServerHeaderV2_P0                                |
| SCPFailureHandling_EnhancedServerHeaderV2_p1                                            | SCP_FailureHandling_EnhancedServerHeaderV2_P1                                |
| FailureHandling_InterSCP_EnhancedServerHeader                                           | SCP_FailureHandling_InterSCPEnhancedServerHeader.P0                          |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                             | New Feature Names for SCP 24.1.0                                       |
|-------------------------------------------------------------------------------|------------------------------------------------------------------------|
| FormUrlEncoded                                                                | SCP_FormUrlEncoded_P0                                                  |
| Global_Egress_Rate_Limiting_Config_API                                        | SCP_GlobalEgressRateLimiting_API_P0                                    |
| GlobalEgressRateLimiting_P0                                                   | SCP_GlobalEgressRateLimiting_P0                                        |
| SCP_HealthCheck_Support_API_P0                                                | SCP_HealthCheckSupport_API_P0                                          |
| SCP_HealthCheck_Support_HTTPS_P0                                              | SCP_HealthCheckSupport_HTTPS_P0                                        |
| SCP_HealthCheckInterSCPMoelC_HTTPS_P0                                         | SCP_HealthCheckSupport_InterSCPMoelC_HTTPS_P0                          |
| SCP_HealthCheckInterSCPMoelC_P0                                               | SCP_HealthCheckSupport_InterSCPMoelC_P0                                |
| SCP_HealthCheckInterSCPMoelD_P0                                               | SCP_HealthCheckSupport_InterSCPMoelD_P0                                |
| Alert_SCPIngressTrafficRateExceededConfiguredLimit                            | SCP_IngressRateLimiting_AlertTrafficRateExceededConfiguredLimit_P      |
| Alert_SCPIngressTrafficRoutedWithoutRateLimitTreatment                        | SCP_IngressRateLimiting_AlertTrafficRoutedWithoutRateLimitTreatment_P0 |
| Ingress_Rate_Limiting_Config_API                                              | SCP_IngressRateLimiting_API_P0                                         |
| Ingress_Rate_Limiting_Config_API_SC1                                          | SCP_IngressRateLimiting_ConfigAPI_P0                                   |
| Ingress_Rate_Limiting_Config_API_enhancement_for_nflnstanceId                 | SCP_IngressRateLimiting_ConfigAPIenhancementNFInstanceId_P0            |
| Ingress_Rate_Limiting_enhancement_for_nflnstanceId_to_support_UserAgentHeader | SCP_IngressRateLimiting_NFInstanceIdEnhancementUserAgentHeader_P0      |
| SEPP_Audit_nnrf_nfm                                                           | SCP_InterPLMNRouting_Audit_nnrf-nfm_P0                                 |
| SCP-SEPP_ModelID_Routing                                                      | SCP_InterPLMNRouting_ModelIDBased_P0                                   |
| SEPP_Routing_NRF                                                              | SCP_InterPLMNRouting_NRF_P0                                            |
| OutlierDetectionSepp_P0                                                       | SCP_InterPLMNRouting_OutlierDetection_P0                               |
| SCP-SEPP_Routing                                                              | SCP_InterPLMNRouting_P0                                                |
| SEPP_plmnInfo_DELETE_API                                                      | SCP_InterPLMNRouting_PLMNInfoDeleteAPI_P0                              |
| SEPP_plmnInfo_GET_API                                                         | SCP_InterPLMNRouting_PLMNInfoGetAPI_P0                                 |
| SEPP_plmnInfo_PATCH_API                                                       | SCP_InterPLMNRouting_PLMNInfoPatchAPI_P0                               |
| SEPP_plmnInfo_PUT_API                                                         | SCP_InterPLMNRouting_PLMNInfoPutAPI_P0                                 |
| SCP-SEPP_Routing_UDM                                                          | SCP_InterPLMNRouting_UDM_P0                                            |
| Rel16_Inter_SCP_Audit_nnrf_nfm_p0                                             | SCP_InterSCP_Audit_nnrf-nfm_P0                                         |
| InterSCP_MateInfoListRouting                                                  | SCP_InterSCP_MateInfoListRouting_P0                                    |
| Inter_SCP_Routing_NplusK_p0                                                   | SCP_InterSCP_NplusKRouting_P0                                          |
| Inter_SCP_Subscription_SMF_NRF                                                | SCP_InterSCP_SubscriptionWithForeignNRFforNFTypeSMF_P0                 |
| Inter_SCP_Routing                                                             | SCP_InterSCP Routing_P0                                                |
| Jaeger_Api_Validation                                                         | SCP_Jaeger_API_P0                                                      |
| SCP_LCI_Support_Config_API                                                    | SCP_LCIsupport_API_P0                                                  |
| Scp_Lci_Support_InterPLMN                                                     | SCP_LCIsupport_InterPLMNRouting_P0                                     |
| SCP_Lci_Support_InterSCP                                                      | SCP_LCIsupport_InterSCP_P0                                             |
| SCP_Lci_Support_InterSCP_PCF_SCP                                              | SCP_LCIsupport_InterSCP_PCF_P0                                         |
| SCP_Lci_Support_InterSCP_UDM                                                  | SCP_LCIsupport_InterSCP_UDM_P0                                         |
| SCP_Lci_Support_Miscellaneous                                                 | SCP_LCIsupport_Miscellaneous_P0                                        |
| Scp_Lci_Support_ModelID                                                       | SCP_LCIsupport_ModelID_P0                                              |
| Scp_Lci_Support_ModelID_notification_cases                                    | SCP_LCIsupport_ModelIDNotificationRequest_P0                           |
| SCP_Lci_Support                                                               | SCP_LCIsupport_P0                                                      |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                        | New Feature Names for SCP 24.1.0                                        |
|--------------------------------------------------------------------------|-------------------------------------------------------------------------|
| Local_N_Plus_K                                                           | SCP_LocalNplusK_P0                                                      |
| Location_hdr_update_for_host_mismatch_API_validation                     | Location_hdr_update_for_host_mismatch_API_validation                    |
| App_Routing_Options_Config_API                                           | SCP_Mediation_AppRoutingOptionsConfig_API_P0                            |
| SCP_Mediation_Config_API                                                 | SCP_Mediation_ConfigAPI_P0                                              |
| SCP_Mediation_Worker_side_cases_PCF_Cases                                | SCP_Mediation_FunctionalCases_PCF_P0                                    |
| SCP_Mediation_Cases_AUSF                                                 | SCP_Mediation_FunctionalCasesPart1_AUSF_P0                              |
| SCP_Mediation_Worker_side_cases_AUSF                                     | SCP_Mediation_FunctionalCasesPart2_AUSF_P0                              |
| SCP_Mediation_Worker_side_cases_UDM_SMF                                  | SCP_Mediation_FunctionalTestCases_UDMandSMF_P0                          |
| SCP_Mediation_Worker_side_cases_UDM_AUSF_SMF_Notification_cases          | SCP_Mediation_FunctionalTestCasesForUDMnAUSFnSMFnNotificationRequest_P0 |
| SCP_Mediation_Cases_PCF                                                  | SCP_Mediation_PCF_P0                                                    |
| Alert_SCPRoutingFailedForProducer_CHF                                    | SCP_ModelC_AlertSCPRoutingFailedForProducer_CHF_P0                      |
| ModelC_allheaderRequest                                                  | SCP_ModelC_AllHeaderRequest_P0                                          |
| PCF_AMPolicy_forwardRoute_Target_apiRoot                                 | SCP_ModelC_AMPolicyTargetAPIRoot_PCF_P0                                 |
| ModelC_CatchAll_Routing_Configuration_AUSF                               | SCP_ModelC_CatchAllRoutingConfiguration_AUSF_P0                         |
| ModelC_Direct_Routing_Configuration_AUSF                                 | SCP_ModelC_DirectRoutingConfiguration_AUSF_P0                           |
| ModelC_ProducerOverloadControl                                           | SCP_ModelC_EgressCongestionBasedOnProducerLoad_P0                       |
| Egress_Congestion_based_on_load_PCF_SMPolicy_forwardRoute_Target_apiRoot | SCP_ModelC_EgressCongestionBasedOnProducerLoad_PCF_P0                   |
| Https_Routing_For_ModelC_Headers                                         | SCP_ModelC_HTTPS_P0                                                     |
| ModelC_IngressRateLimiting                                               | SCP_ModelC_IngressRateLimiting_P0                                       |
| Model-C_based_Inter_SCP_Routing                                          | SCP_ModelC_InterSCProuting_P0                                           |
| ModelC_Inter_SCP_Routing_Options_Notification_Cases                      | SCP_ModelC_InterSCPRoutingOptionsNotificationCases_P0                   |
| InterSCP_VersionSupport                                                  | SCP_ModelC_InterSCPVersionSupport_P0                                    |
| ModelC_NFInstance_AUSF_Cases                                             | SCP_ModelC_NFInstance_AUSF_P0                                           |
| ModelC_NFInstance_NEF_Cases                                              | SCP_ModelC_NFInstance_NEF_P0                                            |
| NF_Notification                                                          | SCP_ModelC_NFprofileNotificationWithMultipleVersions_P0                 |
| ModelC_NFSet_NEF_Cases                                                   | SCP_ModelC_NFSet_NEF_P0                                                 |
| ModelC_NFSet_PCF_Cases                                                   | SCP_ModelC_NFSet_PCF_P0                                                 |
| ModelC_NFSet_SMF_Cases                                                   | SCP_ModelC_NFSet_SMF_P0                                                 |
| ModelC_ReverseLookup_CHF_Cases                                           | SCP_ModelC_ReverseLookup_CHF_P0                                         |
| ModelC_ReverseLookup_UDM_Cases                                           | SCP_ModelC_ReverseLookup_UDM_P0                                         |
| ModelC_ReverseLookup_UDR_Cases                                           | SCP_ModelC_ReverseLookup_UDR_P0                                         |
| ModelC_Inter_SCP_Routing_Options                                         | SCP_ModelC_RoutingOptionsInterSCP_P0                                    |
| AUSF_AUTH_forwardRoute_Target_apiRoot                                    | SCP_ModelC_TargetAPIRoot_AUSF_P0                                        |
| CHF_SLC_forwardRoute_Target_apiRoot                                      | SCP_ModelC_TargetAPIRoot_CHF_P0                                         |
| SMF_forwardRoute_Target_apiRoot                                          | SCP_ModelC_TargetAPIRoot_SMF_P0                                         |
| UDM_SDM_forwardRoute_Target_apiRoot                                      | SCP_ModelC_TargetAPIRootSDM_UDM_P0                                      |
| UDM_UECM_forwardRoute_Target_apiRoot                                     | SCP_ModelC_TargetAPIRootUECM_UDM_P0                                     |
| Version_support                                                          | SCP_ModelC_VersionSupport_P0                                            |
| ModelID_Config_API                                                       | SCP_ModelID_API_P0                                                      |
| ModelID_enforceReqSpecificSvcDiscovery                                   | SCP_ModelID_EnforceReqSpecificSvcDiscovery_P0                           |
| ModelID_based_Routing_implicit_notification_p0                           | SCP_ModelID_ImplicitNotificationRouting_P0                              |



Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                                    | New Feature Names for SCP 24.1.0                                                           |
|----------------------------------------------------------------------|--------------------------------------------------------------------------------------------|
| ModelID_based_Routing                                                | SCP_ModelID_Routing_P0                                                                     |
| Notification_Patch_Profile_Support_SMF                               | SCP_NotificationRequest_PatchProfileSupport_SMF_P0                                         |
| SCP_Oauth2_Access_Token_Grularity_Config_API_P0                      | SCP_Oauth2Support_AccessTokenGranularity_API_P0                                            |
| SCP_Oauth2_Support_Alternative_Config_ModelID_SMF_NRF_P0             | SCP_Oauth2Support_AlternativeConfigModelID_SMFandNRF_P0                                    |
| SCP_Oauth2_Grularity_config_ModelID_SMF_NRF_P0                       | SCP_Oauth2Support_GrularityConfigModelID_SMFandNRF_P0                                      |
| SCP_Oauth2_Local-PLMN-required-config_API_P0                         | SCP_Oauth2Support_LocalPLMNRequiredConfig_API_P0                                           |
| SCP_OAuth2_support_ModelIC_CHF_P0                                    | SCP_Oauth2Support_ModelIC_CHF_P0                                                           |
| Scp_Oauth2_Support_ModelIC_SMF_NRF_P0                                | SCP_Oauth2Support_ModelIC_SMFandNRF_P0                                                     |
| SCP_OAuth2_Support_Config_API_P0                                     | SCP_OAuth2Support_ConfigAPI_P0                                                             |
| SCP_Oauth2_Support_ModelIC_Error_Handling_CHF_NRF_P1                 | SCP_Oauth2Support_ModelICErrorHandling_CHFandNRF_P1                                        |
| SCP_Oauth2_Support_ModelID_SMF_NRF_P0_P1                             | SCP_Oauth2Support_ModelID_SMFandNRF_P1                                                     |
| SCP_Oauth2_Support_ModelID_interSCP_P0                               | SCP_Oauth2Support_ModelIDinterSCP_P0                                                       |
| SCP_Oauth2_NRF_Configuration_Config_API_P0                           | SCP_Oauth2Support_NRFConfiguration_API_P0                                                  |
| SCP_Observability_And_Inter_Microservice_Resilience_P0               | SCP_ObservabilityAndInterMicroserviceResilience_P0                                         |
| SCP_OutlierDetection_API                                             | SCP_OutlierDetection_API_P0                                                                |
| OutlierDetectionInterSCP_P0                                          | SCP_OutlierDetection_InterSCP_P0                                                           |
| OutlierDetectionProducer_AUSF_P0                                     | SCP_OutlierDetection_ProducerAUSF_P0                                                       |
| PseudoHeadersModification                                            | SCP_PseudoHeadersModification_P0                                                           |
| PseudoHeadersModificationHttps                                       | SCP_PseudoHeadersModification_HTTPS_P0                                                     |
| NRF_Registration                                                     | SCP_RegistrationWithNRF_P0                                                                 |
| SCP_Registration_With_PLMNList                                       | SCP_RegistrationWithNRF_PLMNList_P0                                                        |
| Scp_profile_mateScplInfo_update                                      | SCP_RegistrationWithNRF_ProfileMateSCPlInfoUpdate_P0                                       |
| Rel16_Canary_support_PCF_Cases                                       | SCP_Rel16CanarySupport_PCF_P0                                                              |
| Rel16_Canary_support_SMF_Cases                                       | SCP_Rel16CanarySupport_SMF_P0                                                              |
| Rel16_Canary_support_UDM_Cases                                       | SCP_Rel16CanarySupport_UDM_P0                                                              |
| Re_routing_based_on_configurable_http_status_codes_api_validation_p0 | SCP_ReroutingBasedonConfigurableHttpStatusCodes_API_P0                                     |
| Re_routing_based_on_configurable_http_status_codes_p0                | SCP_ReRoutingBasedOnConfigurableHttpStatusCodes_P0                                         |
| SCP_Ro_Enhancements_For_CB_CC_SMF_P0                                 | SCP_RoutingOptionEnhancements_CBAndCongestionControl_SMF_P0                                |
| SCP_Ro_Enhancements_For_CB_CC_InterSCP_InterPLMN_SMF_PCF_P0          | SCP_RoutingOptionEnhancements_CBAndCongestionControlForInterPLMNandSCPRouting_SMFandPCF_P0 |
| SCP_Ro_Enhancement_For_CC_CB_Notification_PCF_P0                     | SCP_RoutingOptionEnhancements_CBAndCongestionControlForNotificationReq_PCF_P0              |
| SCP_Notification_Routing_Option_Config_API                           | SCP_RoutingOptionForNotificationReq_API_P0                                                 |
| SCP_Routing_Option_Set_API                                           | SCP_RoutingOptionForNotificationReq_RoutingConfigSet_API_P0                                |
| Routing_Rules_API_R16                                                | SCP_RoutingRules_R16_API_P0                                                                |
| Sbi_Message_Priority_Config_API                                      | SCP_SBImessagePriority_ConfigAPI_P0                                                        |
| SbiMessagePriority                                                   | CP_SBImessagePriority_P0                                                                   |
| SCP_Features_API                                                     | SCP_SCPFeatures_API_P0                                                                     |
| SCP_Generated_Error_Codes_SMF                                        | SCPGeneratedErrorCodes_SMF_P0                                                              |

Table 4-7 (Cont.) Regression Feature File

| Feature Names prior to SCP 24.1.0                             | New Feature Names for SCP 24.1.0                             |
|---------------------------------------------------------------|--------------------------------------------------------------|
| SCP_Generated_Error_Codes_UDM                                 | SCP_SCPGeneratedErrorCodes_UDM_P0                            |
| NRF_Subscription                                              | SCP_Subscription_SubscriptionWithNRFforNfTypeUDM_P0          |
| Configurataion_Support_For_NEF_services_NEF_P1                | SCP_SupportForNEFservices_Configuration_APIs_P1              |
| Support_For_NEF_services_NEF_Enhanced_NF_Status_P0            | SCP_SupportForNEFservices_EnhancedSuspendedStateRouting_P0   |
| Support_For_NEF_services_NEF_SEPP_P0                          | SCP_SupportForNEFservices_InterPLMNRouting_P0                |
| Support_For_NEF_services_NEF_P0                               | SCP_SupportForNEFservices_NEF_P0                             |
| Support_For_NEF_services_NEF_NRF_SCP_p0                       | SCP_SupportForNEFservices_NRF_P0                             |
| Traffic_Feed_REST_API                                         | SCP_TrafficFeed_API_P0                                       |
| TrafficFeed_CB                                                | SCP_TrafficFeed_CB_P0                                        |
| TrafficFeed_Notification                                      | SCP_TrafficFeed_NotificationRequest_P0                       |
| TrafficFeed_PCF                                               | SCP_TrafficFeed_PCF_P0                                       |
| TrafficFeed_UDM                                               | SCP_TrafficFeed_UDM_P0                                       |
| Consumer_Info_API                                             | SCP_UserAgent_ConsumerInfo_API_P0                            |
| Scp_User_Agent_Info_API                                       | SCP_UserAgent_Info_API_P0                                    |
| Worker_Pod_Overload_Action_Policy_Config_API                  | SCP_WorkerPodOverLoadControl_ActionPolicy_API_P0             |
| CPUOverloadCtrlByNoAction                                     | SCP_WorkerPodOverLoadControl_CPUbasedNoAction_P0             |
| Worker_Pod_Overload_Ctrl_Policy_Config_API                    | SCP_WorkerPodOverLoadControl_CtrlPolicy_API_P0               |
| Worker_Pod_Overload_Discard_Policy_Config_API                 | SCP_WorkerPodOverLoadControl_DiscardPolicy_API_P0            |
| Worker_Traffic_Pod_Overload_P0                                | SCP_WorkerPodOverLoadControl_PendingTransaction_P0           |
| Worker_Traffic_Pod_Overload_P1                                | SCP_WorkerPodOverLoadControl_PendingTransaction_P1           |
| API_Worker_Pod_Pending_Transaction_Overload_Config_P0.feature | SCP_WorkerPodOverLoadControl_PendingTransactionConfig_API_P0 |
| SCP_1_12_0_Bug_Fixes                                          | SCP_1_12_0_BugFixes_P0                                       |
| SCP_1.9.0_Bug_Fixes_SMF                                       | SCP_1.9.0_BugFixes_SMF_P0                                    |
| SCP_1.9.0_Bug_Fixes_UDR                                       | SCP_1.9.0_BugFixes_UDR_P0                                    |
| SCP_22.2.0_Bug_Fixes_PCF_Cases                                | SCP_22.2.0_BugFixes_PCF_P0                                   |
| SCP_22.2.0_Bug_Fixes_UDM_Cases                                | SCP_22.2.0_BugFixes_UDM_P0                                   |
| SCP_22.3.0_Bug_Fixes                                          | SCP_22.3.0_BugFixes_P0                                       |
| SCP_22.3.1_Bug_Fixes                                          | SCP_22.3.1_BugFixes_P0                                       |
| 22.4.0_Bug_Fixes                                              | SCP_22.4.0_BugFixes_P0                                       |
| 22.4.0_Bug_Fixes_Mediation                                    | SCP_22.4.0_Bug_Fixes_Mediation                               |
| 22.4.4_Bug_Fixes                                              | SCP_22.4.4_BugFixes_PCF_P0                                   |
| 23.1.0_Bug_Fixes                                              | SCP_23.1.0_BugFixes_P0                                       |
| 23.2.0_Bug_Fixes_HTTPS                                        | SCP_23.2.0_BugFixes_HTTPS_P0                                 |
| 23.2.0_Bug_Fixes                                              | SCP_23.2.0_BugFixes_P0                                       |

The following table displays the ATS Regression feature tests that have been split into SingleIpEndpoint and MultipleIpEndpoint categories, and renamed accordingly:



**Table 4-8    ATS Regression Feature Tests**

| Feature Test Name before Splitting                          | Feature Test Name after Splitting                                                                                                                                                                                                                                                           |
|-------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SCP_SupportForNEFservices_EnhancedSuspendedStateRouting_P0  | <ul style="list-style-type: none"> <li>SCP_SupportForNEFservices_SingleIpEndpoint_EnhancedSuspendedStateRouting_P0</li> <li>SCP_SupportForNEFservices_MultipleIpEndpoint_EnhancedSuspendedStateRouting_P0</li> </ul>                                                                        |
| SCP_RoutingOptionEnhancements_CBAndCongestionControl_SMF_P0 | <ul style="list-style-type: none"> <li>SCP_RoutingOptionEnhancements_CBAndCongestionControl_SMF_P0</li> <li>SCP_RoutingOptionEnhancements_SingleIpEndpoint_CBAndCongestionControl_SMF_P0</li> <li>SCP_RoutingOptionEnhancements_MultipleIpEndpoint_CBAndCongestionControl_SMF_P0</li> </ul> |
| SCP_OutlierDetection_AUSF_P0                                | <ul style="list-style-type: none"> <li>SCP_OutlierDetection_MultipleIpEndpoint_AUSF_P0</li> <li>SCP_OutlierDetection_SingleIpEndpoint_AUSF_P0</li> </ul>                                                                                                                                    |
| SCP_Oauth2Support_GranularityConfigModelID_SMFandNRF_P0     | <ul style="list-style-type: none"> <li>SCP_Oauth2Support_GranularityConfigModelID_SMFandNRF_P0</li> <li>SCP_Oauth2Support_SingleIpEndpoint_GranularityConfigModelID_SMFandNRF_P0</li> <li>SCP_Oauth2Support_MultipleIpEndpoint_GranularityConfigModelID_SMFandNRF_P0</li> </ul>             |
| SCP_ModelC_NFInstance_NEF_P0                                | <ul style="list-style-type: none"> <li>SCP_ModelC_NFInstance_NEF_P0</li> <li>SCP_ModelC_SingleIpEndpoint_NFInstance_NEF_P0</li> <li>SCP_ModelC_MultipleIpEndpoint_NFInstance_NEF_P0</li> </ul>                                                                                              |
| SCP_LCIsupport_Miscellaneous_P0                             | <ul style="list-style-type: none"> <li>SCP_LCIsupport_Miscellaneous_P0</li> <li>SCP_LCIsupport_SingleIpEndpoint_Miscellaneous_P0</li> <li>SCP_LCIsupport_MultipleIpEndpoint_Miscellaneous_P0</li> </ul>                                                                                     |
| SCP_EgressHostPreference_InterSCPRoutingNonASM_CHF_P0       | <ul style="list-style-type: none"> <li>SCP_EgressHostPreference_SingleIpEndpointInterSCPRoutingNonASM_CHF_P0</li> <li>SCP_EgressHostPreference_MultipleIpEndpointInterSCPRoutingNonASM_CHF_P0</li> </ul>                                                                                    |
| SCP_DNSSRV_ModelCHeaders_PCF_P0                             | <ul style="list-style-type: none"> <li>SCP_DNSSRV_ModelC_SingleIpEndpoint_PCF_P0</li> <li>SCP_DNSSRV_ModelC_MultipleIpEndpoint_PCF_P0</li> </ul>                                                                                                                                            |
| SCP_AlternateRoutingUsingStaticConfig_SMF_P0                | <ul style="list-style-type: none"> <li>SCP_AlternateRoutingUsingStaticConfig_SMF_P0</li> <li>SCP_AlternateRoutingUsingStaticConfig_SingleIpEndpoint_SMF_P0</li> <li>SCP_AlternateRoutingUsingStaticConfig_MultipleIpEndpoint_SMF_P0</li> </ul>                                              |
| SCP_22.3.0_BugFixes_P0                                      | <ul style="list-style-type: none"> <li>SCP_22.3.0_BugFixes_SingleIpEndpoint_P0</li> <li>SCP_22.3.0_BugFixes_MultipleIpEndpoint_P0</li> </ul>                                                                                                                                                |

## 4.7 Running SEPP Test Cases using ATS

This section describes how to run SEPP test cases using ATS.

### 4.7.1 Prerequisites

To run SEPP Test Cases using SEPP ATS 24.1.0, you need to ensure that following prerequisites are fulfilled:

- Debug container should be DISABLED for all microservices during ATS runs.

- To run the ATS testcases, the user should not configure any value for apiPrefix key in ocsepp-custom-values.yaml file. It should be empty.
- The number of replicas of all SEPP microservices pods must be set to 1 as ATS is enabled to perform the metric validations for metrics obtained from a single pod.
- Create Kubernetes secret with certificates/keys (public and private) for both PLMN and N32 gateways before deploying SEPP.
- Deploy SEPP 24.1.0 with default helm configurations using helm charts.
- Ensure all microservices of SEPP are up and running.
- Create Kubernetes secret with certificates/keys (public and private) for ATS client and stub server microservices before deploying SEPP ATS.
- Update ocats\_ocsepp\_values\_24.1.0.yaml file with the latest secrets and certificates.
- Deploy ATS using helm charts.
- Ensure Prometheus service is up and running.(Optional)
- The user must not initiate a job in two different pipelines at the same time.
- User must not abort a running job, this may lead to data corruption in the database.
- The user needs to edit cn32f-svc and pn32f-svc deployment yaml. Set CACHE\_REFRESH\_TIMEOUT to 1000.
- User need to set n32cHandshakePlmnIdListValidationEnabled to true in localProfile section of ocsepp-custom-values.yaml file. Or n32cHandshakePlmnIdListValidationEnabled Parameter can also be changed after SEPP deployment by editing cn32c-svc and pn32c-svc deployment yaml.
- User needs to update the plmnIdList provided in the ATS installation guide to run the ATS testcases.
- The user needs to edit cn32f-svc and pn32f-svc deployment yaml. Set "TOPOLOGY\_HIDING\_CACHE\_REFRESH\_TIMEOUT" to 1000.
- The user needs to edit cn32f-svc and pn32f-svc deployment yaml. Set "SECURITY\_CACHE\_REFRESH\_TIMEOUT" to 1000.
- All the features are set to disable by default. Ensure that the features must be disabled while configuring SEPP ATS.
- The user needs to edit cn32f-svc, pn32f-svc, plmn-egress-gateway, and n32-egress-gateway deployment yaml. Set "REQUEST\_TIMEOUT" to 2000.
- The user needs to edit n32-ingress-gateway deployment yaml. Set "REQUEST\_TIMEOUT" to 5000.
- The user needs to edit configmap of n32-ingress-gateway. Set "requestTimeout" to 5000.
- The user needs to edit n32-ingress-gateway deployment yaml. SET "IDLE\_TIMEOUT" to 5000.
- The user needs to edit pn32f-svc deployment yaml. SET "EVICT\_SAN\_HEADER\_CACHE\_DELAY" to 100.
- The user needs to edit pn32f-svc deployment yaml. SET "NRF\_DISCOVERY\_CACHE\_REFRESH\_TIMEOUT" to 1000.
- To run the CAT3 testcases, user must have pre-installed coherence-svc while SEPP installation.
- For DNS SRV and SOR, plmn-egress-gateway should be deployed in REST mode.

- To run the DNS SRV testcases, user must have pre-installed alternate-route service while SEPP installation.
- To run the Load Sharing among Multiple Remote SEPP Nodes testcases, user must have pre-installed alternate-route service while SEPP installation.
- To run the Egress Rate Limiter testcases user must have enabled the Egress Rate Limiter feature while SEPP deployment. Also SET "EgressRateLimiterFlag" to true in `ocats_ocsepp_values_24.1.0.yaml`.
- To run the Ingress Rate Limiter testcases user must have enabled the Ingress Rate Limiter feature while SEPP deployment. Also SET "IngressRateLimiterFlag" to true in `ocats_ocsepp_values_24.1.0.yaml`.
- In case of ASM, update config manager port at the following path inside the BDD client pod:
  - In the `/var/lib/jenkins/ocsepp_tests/product_config/global.yaml` file, update config manager port to **9092**.
  - In the `/var/lib/jenkins/ocsepp_tests/custom_config/global.yaml` file, update config manager port to **9092**.

### Custom Folder Implementation

ATS provides custom test case folders (`cust_newfeatures`, `cust_regression` and `cust_performance`) using which you can add new test cases, remove unwanted test cases and modify existing test cases. It does not impact the original product packaged test cases available in the newfeatures, regression and performance folders. For more details, refer to [Custom Folder Implementation](#).

## 4.7.2 Logging into ATS

Before logging into ATS, you need to ensure that ATS is deployed successfully using Helm charts as shown below:

#### Note

To modify default log in password, refer to [Modifying Login Password](#).

**Figure 4-71 Pods**

```
[root@general-cluster-master ~]# kubectl get po $n
```

| NAME                                                 | READY | STATUS  | RESTARTS | AGE |
|------------------------------------------------------|-------|---------|----------|-----|
| ocsepp-release-cn32c-svc-5f46898d8d-htf68            | 1/1   | Running | 0        | 89s |
| ocsepp-release-cn32f-svc-78798dd6fb-2fwnr            | 1/1   | Running | 0        | 89s |
| ocsepp-release-config-mgr-svc-647d8c845f-xh47s       | 1/1   | Running | 0        | 89s |
| ocsepp-release-n32-egress-gateway-7745b6746b-9dtcl   | 2/2   | Running | 0        | 89s |
| ocsepp-release-n32-egress-gateway-7745b6746b-mtfk6   | 2/2   | Running | 0        | 89s |
| ocsepp-release-n32-ingress-gateway-5b4dc467cc-87wzh  | 2/2   | Running | 0        | 89s |
| ocsepp-release-n32-ingress-gateway-5b4dc467cc-8td78  | 2/2   | Running | 0        | 88s |
| ocsepp-release-plmn-egress-gateway-568ffcdcc7-7zm7j  | 2/2   | Running | 0        | 88s |
| ocsepp-release-plmn-egress-gateway-568ffcdcc7-t9hbd  | 2/2   | Running | 0        | 89s |
| ocsepp-release-plmn-ingress-gateway-6b54cb9cfc-5bkz9 | 2/2   | Running | 0        | 88s |
| ocsepp-release-plmn-ingress-gateway-6b54cb9cfc-t2mc4 | 2/2   | Running | 0        | 88s |
| ocsepp-release-pn32c-svc-84c978bcd-5nn47             | 1/1   | Running | 0        | 88s |
| ocsepp-release-pn32f-svc-78c558947c-vfhnp            | 1/1   | Running | 0        | 87s |
| sepp-ats-rel-bddclient-76745c5547-wp28r              | 1/1   | Running | 0        | 12h |
| sepp-ats-rel-stubserver-2-7d56d9887c-smfkr           | 1/1   | Running | 0        | 89s |
| sepp-ats-rel-stubserver-3-77f7cc8688-r246j           | 1/1   | Running | 0        | 89s |
| sepp-ats-rel-stubserver-57c998d5b4-rc56f             | 1/1   | Running | 0        | 89s |
| sepp-mysql-540/c5b99a-jzbt5                          | 1/1   | Running | 0        | 14h |

For more information on verifying ATS deployment, see [Verifying ATS Deployment](#).  
There are following ways to login to ATS GUI.

- When an external load balancer (metalLB in case of OCCNE) is available and an external IP Address is provided to the ATS service, login to ATS GUI using <External-IP>:8080.
- When an external IP Address is not provided to the ATS service, open the browser and provide the external IP Address of the worker node and nodeport of the ATS service to login to ATS GUI.  
<Worker-Node-IP>:<Node-Port-of-ATS>

**Note**

In the **Verifying ATS Deployment** screen, ATS nodeport is highlighted in red as 30076. For more details on ATS deployment, refer to **SEPP ATS Installation Procedure**.

- When ATS is installed with TLS enabled:
  - If the external load balancer IP is used in the ssl.conf file to open the ATS GUI then the URL will be "https://<IP>:<https\_nodeport>".
  - If the external load balancer IP or worker node IP is not present on the setup, then with the common name provided in the ssl.conf file, the user can open the ATS GUI. The user needs to do the ssh port forwarding and then need to update the common name (CN) in the etc/hosts/ file of the local machine against the localhost IP (127.0.0.1).
  - ATS GUI URL format:
    - \* https://127.0.0.1:<port-forwarding-port>
    - \* https://<common name>:<port-forwarding-port>

Open a browser and provide IP Address and port details as <Worker-Node-IP>:<NodePort-of-ATS> (As per the above example: 10.98.101.171:32013). The ATS login screen appears.

**Figure 4-72 ATS Login**

**Oracle Communications Cloud Native Core - Automated Test Suite**

**Welcome to ATS!**

Username

Password

☐ Keep me signed in

Sign in

To run ATS:

- Enter the **username** as 'seppuser' and **password** as 'sepppasswd'. Click **Sign in**. The following screen appears showing pre-configured pipelines for SEPP individually (3 Pipelines).

**Figure 4-73 Pre-Configured Pipelines**

| S | W | Name             | Last Success | Last Failure | Last Duration | F   |
|---|---|------------------|--------------|--------------|---------------|-----|
| ☐ | ☀ | SEPP-NewFeatures | N/A          | N/A          | N/A           | ▶ ☆ |
| ☐ | ☀ | SEPP-Regression  | N/A          | N/A          | N/A           | ▶ ☆ |

- SEPP-NewFeatures:** This pipeline has all the test cases that are delivered as part of SEPP ATS - 24.1.0
- SEPP-Regression:** This pipeline has all the test cases of previous releases.
- SEPP-HealthCheck:** This pipeline has the utility to run the helm test functionality.

#### **Note**

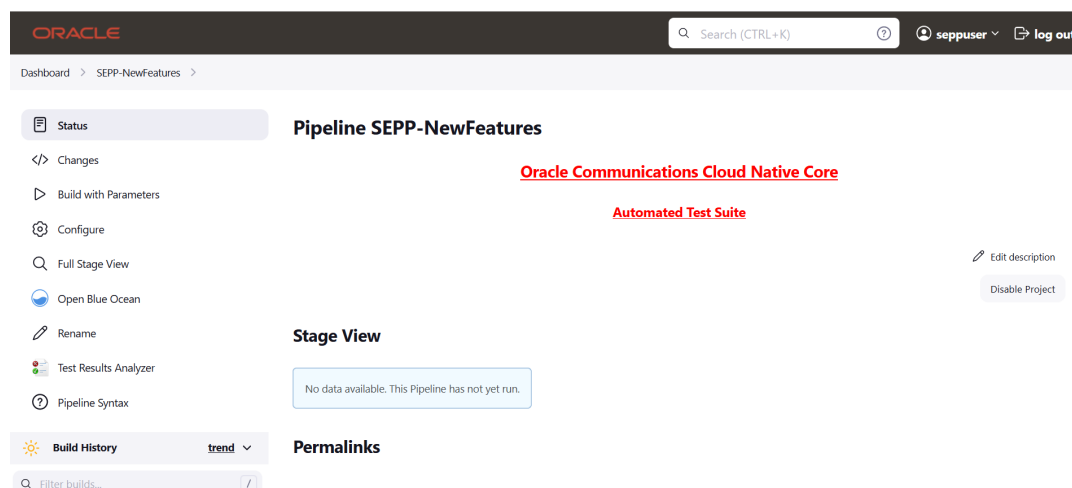
Currently, SEPP does not have any performance pipeline.

## 4.7.3 SEPP-NewFeatures Pipeline

After identifying the SEPP pipelines, the user needs to do one-time configuration in ATS as per their SUT deployment. In this pipeline, all the new test cases related to SEPP are run. To configure its parameters:

- Click **SEPP-NewFeatures** in the Name column. The following screen appears:

Figure 4-74 SEPP-NewFeatures Pipeline



In the above screen:

- Click **Configure** to configure SEPP-New Features.
  - Click **Documentation** to view the documented test cases which are part of this SEPP release.
  - Click blue dots inside **Build History** box to view the success console logs of the "All" and "Sanity" respectively.
  - The **Stage View** shows pipeline that is already run for the customer reference.
  - Click **Open Blue Ocean** to view the respective group, re-run logs and the consolidated report summary.
2. Click **Configure**. Users must wait for the page to load completely. Once the page loads completely, click the **Pipeline** tab to reach the Pipeline configuration as shown below: The Pipeline section of the configuration page appears as follows:

Figure 4-75 Newfeature Pipeline

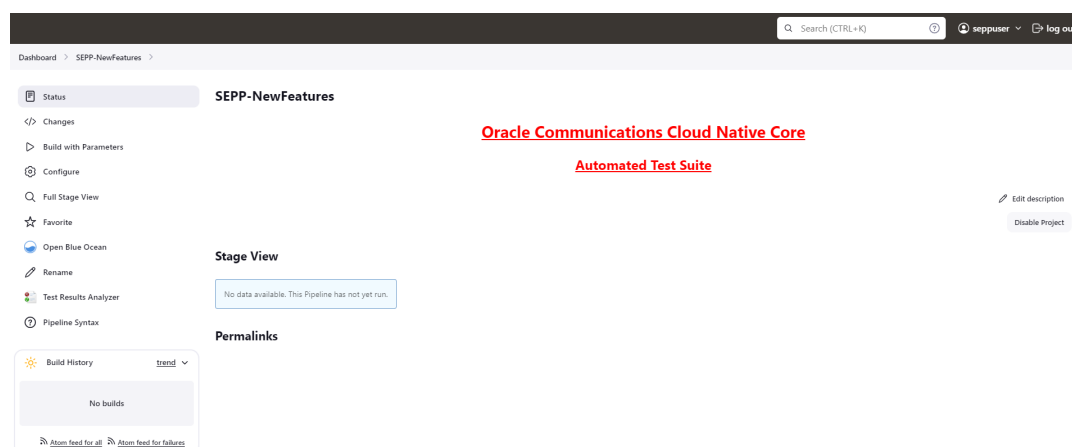


**Figure 4-76 Pipeline script**

```

node ('built-in'){
//a = SELECTED_NF b = SEPPCONFIGSVCNAME c = CSEPPIGWNAME d = PSEPPIGWNAME
//e = SEPPSTUBNAME f = NFINSTANCEID g = PROMSVCIIP h = PROMSVCPOR i = PROMSVCUR j = RERUN_COUNT
//k = ELK_WAIT_TIME l = ELK_HOSTNAME m = ELK_PORT n = STUB_LOGS o = LOG_TYPE p = GLOBAL_WAIT_TIME
withenv([

]){

sh '''
sh /var/lib/jenkins/ocsepp_tests/preTestConfig.sh \
-a SEPP \
-b ocsepp-release-config-mgr-svc.csepp \
-c ocsepp-release-plmn-ingress-gateway.csepp \
-d ocsepp-release-n32-ingress-gateway.csepp \
-e sepp-ats-rel-stubserver.csepp \
-f 9faf1bbc-6e4a-4454-a507-aef01a101a06 \
-g prometheus.cne-infra \
-h 9090 \
-i /api/ \
-j 1 \
-k 10 \
-l occne-elastic-elasticsearch-master.occne-infra \
-m 9200 \
-n yes \
-o kubernetes \
-p 5
'''
if(env.Include_Regression && "${Include_Regression}" == "YES"){
sh '''sh /var/lib/jenkins/common_scripts/merge_jenkinsfile.sh'''
load "/var/lib/jenkins/ocsepp_tests/jenkinsData/Jenkinsfile-Merged"
}
else{
load "/var/lib/jenkins/ocsepp_tests/jenkinsData/Jenkinsfile-NewFeatures"
}
}
}
}

```

In the above screen, you can change the values of the 'Pipeline script'.

- The user must not change any other value apart from line number 12 to line 20 and line number 26.
- User can change only those parameters that are marked as "a" to "h" as per your requirement.
- A new parameter "p" has been introduced to customize the global wait time for the scale up and scale down of pods test cases. Pods scaling time is different on different setups. Sometimes pods take longer than usual for scaling up or scaling down operations. To handle this situation you can increase the global wait time parameter as required. Recommended value is 60.
- Recommended value for rerun count is "3".

a: Name of the NF to be tested in capital (SEPP).  
b: Config manager service name and namespace  
c: Plmn ingress service name and namespace  
d: N32 ingress service name and namespace  
e: Stub server service name and namespace  
f: SEPP Instance ID configured  
g: Prometheus hostname and namespace  
h: Prometheus port  
i: Prometheus API endpoint  
j: Re run count  
k: ELK wait time before logs are fetched  
l: ELK hostname and namespace

m: ELK port  
 n: Stub log collection enable  
 o: Type of log collection method ( kubernetes/Elasticsearch)  
 p: GLOBAL WAIT TIME

### Note

Do not change any value if OCCNE cluster is used and SEPP, ATS and STUB are deployed in ocsepp namespace.

- Click **Save** after making necessary changes. The **Pipeline SEPP-NewFeatures** screen appears.

## Running SEPP Test Cases

To run SEPP test cases:

- Click the **Schedule a Build with parameters** icon present on the SEPP-NewFeatures screen in the extreme right column corresponding to SEPP-NewFeatures row. The following screen appears:

Figure 4-77 SEPP-NewFeatures

The screenshot displays the 'Pipeline SEPP-NewFeatures' interface. At the top, it says 'This build requires parameters:'. Below this, there are two main sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'. In the 'EXECUTION OPTIONS' section, there are four dropdown menus: 'TestSuite' (set to 'NewFeatures'), 'Execute\_Suite' (set to 'SEPP'), 'Configuration\_Type' (set to 'Product\_Config'), and 'Fetch\_Log\_Upon\_Failure' (set to 'NO'). There is also a 'Log\_Level' dropdown set to 'Not Applicable'. In the 'FEATURES AND TESTCASES' section, there is a 'Features' dropdown set to 'All'. Below this, it shows 'Stage: sequential\_stage' and a list of features: 'group1' (containing 'CN32C\_PORT\_SEGREGATION\_REINIT\_IPX') and 'PN32F\_SL\_2340\_IPX'. At the bottom, there are 'Build' and 'Cancel' buttons.

- In the above screen, there is an option **Execute\_Suite** to select SEPP or Roaming\_Hub. If you have deployed SUT as SEPP, select **SEPP** option and if you have deployed SUT as Roaming\_Hub, select **Roaming\_Hub** option.
- In the above screen, there is an option **Configuration\_Type** to select product/custom configuration.
- In the above screen, there are three **Select\_Option(s)**, which are:
  - All:** This is the default option. It runs all the SEPP test cases. Scroll-down and click **Build** to run all the test cases.
  - Sanity:** It is recommended to run Sanity before running any test case. It ensures all the deployments are done properly.
  - Single/Multiple Feature:** This option allows you to select any number of test cases you want to run from total number of test cases available for execution. After selecting the test cases, scroll-down and click **Build** to run the selected SEPP test cases.



- Click **Build** to execute the ATS run.

In the above screen, the feature files are now shown in the respective stage and group.

## 4.7.4 NewFeatures - Documentation

To view SEPP functionalities, go to SEPP-NewFeatures pipeline and click the **Documentation** link in the left navigation pane. The following screen appears:

**Figure 4-78 Documentation**

| Related Pages                                      |  |
|----------------------------------------------------|--|
| Here is a list of all related documentation pages: |  |
| <a href="#">CN32C_PORT_SEGREGATION_REINIT</a>      |  |
| <a href="#">CN32F_SI_2340</a>                      |  |

Click any functionality to view its test cases and scenarios of each test case. A sample screen is given below:

**Figure 4-79 Test cases and Scenarios**

| CN32F_SI_2340                                                                                                                                                                                                  |                                                                                                           |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|
| #Copyright 2020 (C), Oracle and/or its affiliates. All rights reserved.                                                                                                                                        |                                                                                                           |
| <b>Description :</b>                                                                                                                                                                                           |                                                                                                           |
| To Validate PLMNIngress Gateway successfully performs the egress rate limiting based on the configured Bucket options and N32 EGW routes the non discarded messages on the basis of virtualHost or N32F Param. |                                                                                                           |
| <b>Pre-requisite :</b>                                                                                                                                                                                         |                                                                                                           |
| 1. All components of SEPP are deployed - plmn gateways, n32 gateways, config-mgr, n32c and n32f microservices.                                                                                                 |                                                                                                           |
| 2. Test stubs (bddclient, stubserver) are deployed.                                                                                                                                                            |                                                                                                           |
| <b>cn32f_eri_message_priority_greater_routing_to_high_priority</b>                                                                                                                                             |                                                                                                           |
| <b>Objective :</b>                                                                                                                                                                                             |                                                                                                           |
| Verify incoming request at plmn ingress gateway is dropped when bucket capacity is exhausted and msg priority header value is greater or equal to cutoff priority.                                             |                                                                                                           |
| Procedure                                                                                                                                                                                                      | Expected Result                                                                                           |
| 1. Enable global egress rate limiting with both 3gpp-sbi-target-apiroot and .authority headers.                                                                                                                | 1. Request should be successful                                                                           |
| 2. Create egress rate limiting list with bucket capacity as 1, refill rate as 1, duration as 100, request token as 1.                                                                                          | 2. Request should be successful                                                                           |
| 3. Send message from bdd client with 3gpp-sbi-target-apiroot header                                                                                                                                            | 3. Request is sent successfully and token in bucket would be exhausted                                    |
| 4. Consumer NF(bddclient) sends request with 3gpp-sbi-target-apiroot header and message priority greater than or equal to cutoff priority to producer NF                                                       | 4. Verify plmn ingress gateway rejects the message as its priority is greater or equal to cutoff priority |
| 5. N32 Egress Gateway routes to the highest priority Producer SEPP                                                                                                                                             | 5. Verify n32 egress gateway forwards request to the highest priority producer sepp.                      |

Based on the functionalities covered under the documentation, the build Requires parameter screen displays test cases. To navigate back to the Pipeline SEPP-NewFeatures screen, click the Back to SEPP-NewFeatures link available on top left corner of the screen.

## 4.7.5 SEPP Regression Pipeline

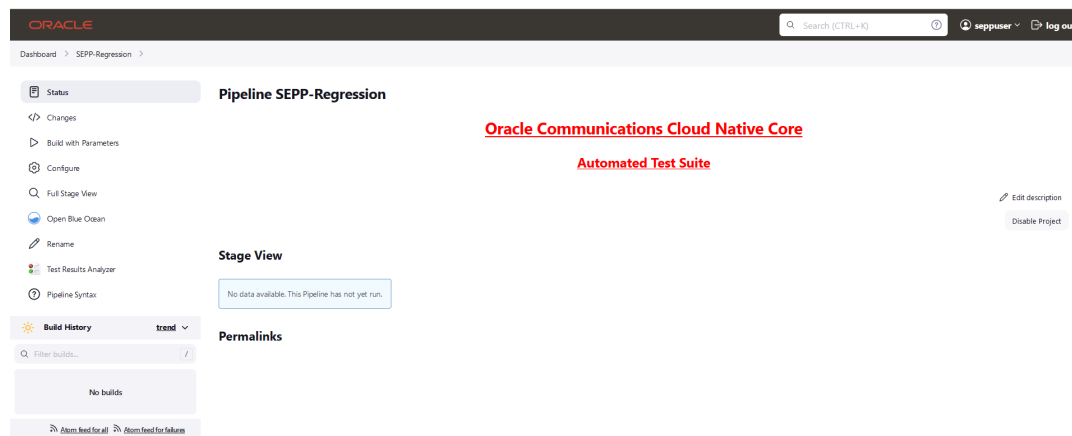
This section describes how to run test cases for SEPP Regression pipeline.

The **SEPP\_Regression** pipeline is a pre-configured pipeline where all the test cases of previous releases are available. For example, for SEPP 23.4.x, this pipeline has all the test cases released till SEPP 23.3.x.

To view **SEPP\_Regression** pipeline details, perform the following steps:

1. Click **SEPP\_Regression** in the **Name** column. The following screen appears:

**Figure 4-80 SEPP Regression**



2. Click **Build with Parameters** in the left navigation pane.
3. Click **Documentation** to view the documented test cases which are part of this SEPP release.
4. Click blue dots in the **Build History** box to view the success console logs of the "All" and "Sanity" respectively.
5. The **Stage View** shows pipeline that is already run for the customer reference.
6. Click **Configure** to configure SEPP Regression. Users must wait for the page to load completely. Once the page loads completely.
7. click the **Pipeline** tab to see the Pipeline configuration.

Copy the required test cases that are available in the SEPP folder and place them appropriately within the **custom** folder for SEPP\_Regression. Reload the page to view the test cases available in the **custom** Regression folder.

The configuration method and parameters are same as the **NewFeatures** pipeline.

## 4.7.6 SEPP Regression Documentation

This section describes the documentation for **SEPP-Regression** pipeline.

To view the documentation for any of the SEPP features, on the ATS home page, click SEPP-Regression. Then, click **Documentation** in the left navigation pane.

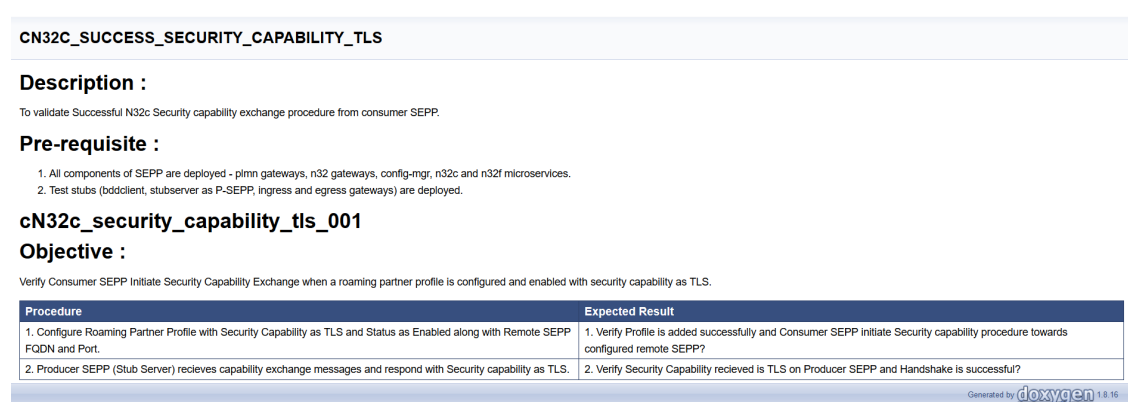
This page shows features of only those test cases that are released in previous releases.

The following screen shows all the documentation features:

**Figure 4-81 SEPP-Regression documentation**

Click any functionality to view its test cases and scenarios of each test case.

The following screen appears if the CN32C\_SUCCESS\_SECURITY\_CAPABILITY\_TLS is selected.

**Figure 4-82 CN32C\_SUCCESS\_SECURITY\_CAPABILITY\_TLS**

## 4.7.7 Running SEPP Health Check Pipeline

To run health check on SEPP:

1. Navigate to **SEPP-HealthCheck** pipeline and click **Configure**.
2. The **General** tab appears. The user must wait for the page to load completely.
3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

### ! Important

Ensure that the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

**Figure 4-83 SEPP-HealthCheck**

Do not change any value other than line numbers 17, 18, and 19. You can change the parameters marked as "a","b",and "c" as per user requirement. The parameters details are provided as comments in line number 7, 8, and 9. The parameters description is as follows:

- a - helm releases [Provide Release Name with Comma Separated if more than 1 ]
- b - namespace of SUT.
- c - helm version supported

4. Click **Save**. The **Pipeline SEPP-HealthCheck** page appears.
5. Click **Build Now**. This triggers health check for SUT.

Example output:

```

Started by user seppuser
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/.jenkins/workspace/SEPP-HealthCheck
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Helm-Test-Init)
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh
+ set +x
export HelmTestRelease=ocsepp-release-naman
export HelmTestNamespace=sepp2
export HelmCmd=helm
[Pipeline] sh
+ export NF=SEPP
+ NF=SEPP
+ /env/bin/python3 /env/lib/python3.9/site-packages/ocnfctest_lib/
helmtest_with_secret.py
[Pipeline] }
[Pipeline] // catchError
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage

```

```

[Pipeline] { (Validate-Helm-Test)
[Pipeline] catchError
[Pipeline] {
[Pipeline] echo
2022-08-29 05:14:23,045[32m INFO LOG.HELM-TEST:108 [0m] [32mhelm test
ocsepp-release-naman -n sepp2[0m
2022-08-29 05:14:53,006[32m INFO LOG.HELM-TEST:120 [0m] [32m
NAME: ocsepp-release-naman
LAST DEPLOYED: Wed Aug 24 10:08:10 2022
NAMESPACE: sepp2
STATUS: deployed
REVISION: 1
TEST SUITE: ocsepp-release-naman-test
Last Started: Mon Aug 29 05:14:36 2022
Last Completed: Mon Aug 29 05:15:00 2022
Phase: Succeeded[0m

[Pipeline] }
[Pipeline] // catchError
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

## 4.8 Running UDR Test Cases using ATS

This section describes how to run UDR test cases using ATS. It includes:

- [Prerequisites](#)
- [Logging into ATS](#)
- [Running SLF Pipelines](#)
- [Running UDR Pipelines](#)
- [Running EIR Pipelines](#)

### 4.8.1 Prerequisites

This section describes prerequisites for Subscriber Location Function (SLF) pipelines, Unified Data Repository (UDR) pipelines, and Equipment Identity Register (EIR) pipelines.

- [Prerequisites for SLF Pipelines](#)
  - [UDR Custom Values File Configuration for SLF Pipelines](#)
  - [Provisioning Gateway Custom Values File Configuration for SLF Pipelines](#)
- [Prerequisites for UDR Pipelines](#)
  - [UDR Custom Values File Configuration for UDR Pipelines](#)
- [Prerequisites for EIR Pipelines](#)
  - [UDR Custom Values File Configuration for EIR Pipelines](#)

### 4.8.1.1 Prerequisites for SLF Pipelines

The prerequisites to run SLF pipelines are:

- Select any one of the following System Under Test (SUT) deployment models, depending on your requirements.
  - **Model 1:** To run all the scenarios (SUTSuite: 'All') in the SLF-Regression and SLF-NewFeatures pipeline, deploy UDR with the following configurations:
    - \* Four UDRs like UDR1, UDR2, UDR3, and UDR4 where UDR1 and UDR2 are in segment1 and UDR3 and UDR4 in segment2.
    - \* Deploy one Provisioning Gateway along with Provisioning Gateway Auditor service.
    - \* Deploy two NRF-Stubs
    - \* Deploy one ATS in same namespace as UDRs, ProvGW, NRF-Stubs

To deploy four UDRs in same namespace as ProvGw, ATS, and NRF stubs, create four secrets, such as each UDR has unique configuration database (configdb), but UDRs in same segment shares the subscriber database.

Configure the Provisioning Gateway segment so that the UDR1 and UDR3 are preferred SLFs in their segment 1 and segment 2 respectively. You must also deploy bulk import tool, export tool, and OSO.

- **Model 2:** To run the SLF-Performance pipeline, deploy UDR with the resources defined in the [Resource Requirements](#) section.

The UDR must be deployed with the following replica counts:

**Table 4-9 Replica Count**

| Service name        | Setup type | CPU/Pod | Memory/Pod | Number of pods |
|---------------------|------------|---------|------------|----------------|
| ingressgateway-sig  | UDR        | 6       | 4          | 2              |
| ingressgateway-prov | UDR        | 6       | 4          | 2              |
| nudr-drservice      | UDR        | 5       | 4          | 2              |
| nudr-provservice    | UDR        | 5       | 4          | 2              |

The ATS pod must be installed with the following configuration for 1K TPS:

**Table 4-10 ATS Pod Installation for 1K TPS**

| Service name | Setup type | CPU/Pod | Memory/Pod | Number of pods |
|--------------|------------|---------|------------|----------------|
| ocats-udr    | ATS        | 4       | 6          | 1              |

The ATS pod must be installed with the following configuration for 2K TPS:

**Table 4-11 ATS Pod Installation for 2K TPS**

| Service name | Setup type | CPU/Pod | Memory/Pod | Number of pods |
|--------------|------------|---------|------------|----------------|
| ocats-udr    | ATS        | 8       | 16         | 1              |

**Note**

OSO must also be deployed on the same namespace.

**Note**

The above configuration is tested for both, constant 1K TPS and varying TPS between 1K and 2K, for a duration of 15 minutes.

- **Model 3:** To run the **Multi\_Site** option of the **SUTSuite** field in the SLF Regression pipeline, deploy two UDRs in two different namespaces having two different cnDBTier with database site replication feature enabled between the two. Provisioning Gateway also needs to be deployed on each namespace. Bulk Import tool and export tool needs to be deployed on first namespace. You must also deploy OSO on each namespace.

#### 4.8.1.1.1 UDR Custom Values File Configuration for SLF Pipelines

This section highlights only the necessary changes to be done in the UDR custom-values.yaml file to run ATS.

**Note**

- Deploy UDR in the SLF mode. For more information about UDR installation, see *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*.
- To configure ASM with UDR (including setting istio-proxy, CPU, and memory), see the **ASM Specific Configuration** and **UDR Configuration Parameters** sections in the *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*. The UDR custom-values.yaml file is available with documentation package on MOS.

- Configure the global section of the OCUDR custom values file as follows:

```
global:
 dockerRegistry: <docker-registry>:<docker port>/ocudr
 mysql:
 dbServiceName: <db service to which UDR should be connected>
 port: &dbPortNumber "3306"
 configdbname: &configdbname <config db name>
 dbname: &dbname <subscriber db name>
 ...
 createNetworkPolicy: true #only if network policy is supported in the
environment. SLF pipeline to validate network policy supports only default
policy
 dbCredSecretName: <udr secret name>
 ...
 alternateRouteServiceEnable: true
 alternateRouteServiceHost: <udr-helm-release-name>-alternate-route
 ...
 performanceServiceEnable: true
 ...
```

```

 udrServices: "nudr-group-id-map"
 ...
 diamGatewayEnable: false
 ...
 appInfoBaseUrl: "<udr-helm-release-name>-nudr-app-info:5906"
 ...
 nrfClientNfManagementEnable: true # value is true for UDR1 deployment;
 UDR2, UDR3 and UDR4 should be deployed with false
 deploymentNrfClientService:
 envNfNamespace: <namespace where UDR is to be deployed>
 ...
 enableControlledShutdown: &enableControlledShutDown true
 ...
 serviceMeshCheck: &serviceMeshFlag true #if UDR is intended to be
 deployed with ASM.

```

#### Note

- If UDR is installed with ASM, then use the annotations defined in the *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*.
- Database names provided in the SLF custom.values files must match with the names created in the corresponding database secrets.
- The nfInstanceId for each of SLF must be unique for all model.

- Update the **config-server** section as follows:

```

config-server:
 fullnameOverride: <udr-helm-release-name>-config-server

```

- Update the **ingressgateway-sig** section as follows:

```

ingressgateway-sig:
 global:
 istioIngressTlsSupport:
 ingressGateway: true #only if UDR is being deployed with ASM.
 ...
 lciHeaderConfig:
 enabled: true
 loadThreshold: 30
 localLciHeaderValidity: 10000
 ...
 svcToSvcInstanceIdMapping:
 - svcName: "<udr-helm-release-name>-nudr-drservice"
 serviceInstanceId: <serviceInstanceId-defined-in-appProfiles-for-
 nudr-group-id-map>
 ...
 xfccHeaderValidation:
 validation:
 enabled: &xfccValidationEnabled true #should be disabled for Model
 2 and Model 3.
 ...
 serverHeaderDetails:

```



```

 enabled: true
 ...
 oauthValidatorEnabled: true #should be disabled for model 2 and model 3.
 nrfPublicKeyKubeSecret: <oauth secret name for sig>
 nrfPublicKeyKubeNamespace: <UDR namespace>
 validationType: relaxed
 ...
 enableControlledShutdown: true
 controlledShutdownConfigMode: REST
 ...
 userAgentHeaderValidationConfigMode: REST

```

- Update the **ingressgateway-prov** section as follows:

```

ingressgateway-prov:
 global:
 istioIngressTlsSupport:
 ingressGateway: true #only if UDR is being deployed with ASM.
 xfccHeaderValidation:
 validation:
 enabled: &xfccValidationEnabled false
 ...
 serverHeaderDetails:
 enabled: false
 ...
 oauthValidatorEnabled: true #should be disabled for model 2 and model 3.
 nrfPublicKeyKubeSecret: <oauth secret name for prov>
 nrfPublicKeyKubeNamespace: <UDR namespace>
 validationType: relaxed
 ...
 enableControlledShutdown: true
 controlledShutdownConfigMode: REST

```

- Configure targets and ports in the **alternate-route.staticVirtualFqdns** section with the NRF stubs FQDNs and ports.

#### Note

This is applicable only for Model 1.

```

alternate-route:
 staticVirtualFqdns:
 - name: http://abc.test.com
 alternateFqdns:
 - target: <stub1 fqdn>
 port: <stub1 port>
 priority: 10
 - target: <stub2 fqdn>
 port: <stub2 port>
 priority: 20

```

- Update the **appinfo** section as follows:

```

appinfo:
 ...
 dbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-tier/
status/local
 realtimeDbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc
port>/db-tier/status/cluster/local/realtime
 replicationUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-
tier/status/replication
 ...
 prometheusUrl: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-
name>/prometheus"
 alertmanagerUrl: "http://<alertmanager fqdn or IP>:<prometheus port>/
<domain-name>/alertmanager"
 tagNamespace: kubernetes_namespace # change this field only if it is
being deployed with OSO. Please refer to UDR User Guide for more
information.
 ...
 core_services:
 udr: *slf
 ...
 fullnameOverride: <udr-helm-release-name>-app-info

```

Run the below curl command from ATS pod to find domain name of OSO:

```
curl -i http://<OSO FQDN or IP>:<OSO PORT>/
```

Example:

```

(env) [jenkins@ocats-udr-5c55f59d66-jkzml ~]$ curl -i http://oso-svr:80/
HTTP/1.1 302 Found
Content-Type: text/html; charset=utf-8
Location: /cne-23-1/prometheus
Date: Thu, 20 Apr 2023 06:28:48 GMT
Content-Length: 48

```

```
Found.
```

Prometheus URL is `http://oso-svr:80/cne-23-1/prometheus`.

- Update the **nrf-client** section as follows:

#### Note

This is applicable only for Model 1.

```

nrf-client:
 profile: |
 #uncomment appProfiles that is dedicated to used for SLF mode (nudr-
group-id-map) and comment the rest of the appProfiles.
 enableVirtualNrfResolution=true

```

```
#virtualNrfFqdn value must be same as that of alternate-
route.staticVirtualFqdns.name[0] without the url scheme as shown below
virtualNrfFqdn=abc.test.com
nrf-client-nfmanagement:
...
replicas: 1
...
enablePDBSupport: false
```

- Update the **perf-info** section as follows:

```
perf-info:
 envMySQLDatabase: <configdb name used for UDR>
 ...
 configmapPerformance:
 prometheus: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-
name>/prometheus"
 log:
 level:
 perfinfo: "WARN"
```

#### Note

The values for namespace, container name, and servicename depend on CNE or OSO versions. For information, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Update the **egressgateway** section as follows:

```
egressgateway:
 userAgentHeaderConfigMode: REST
```

- Configure the following parameters in the bulk-import tool yaml file:

#### Note

To test the bulk import tool, it is important to install the bulk import tool as a separate helm deployment in the same namespace as that of UDR. For information about the bulk import tool installation and sample bulk import yaml file, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*. This is applicable for Model 1 only.

- Ensure the secret is used to install subscriber bulk import tool is same as UDR1. The Bulk Import tool should communicate with UDR1.

#### **bulk import deployment**

```
global:
 mysql:
 dbServiceName: "<DB Endpoint provided for UDR deployment>"
 port: "<PORT Used provided UDR deployment>"
 ...
 dbCredSecretName: '<UDR1 Secret>'
```

```

...
xmlToCsv:
 enabled: false
...
customExtension:
 allResources:
 labels: {}
 annotations:
 oracle.com/cnc: "true"
...
serviceMeshCheck: true #if exprt tool is intended to be deployed with
ASM ...
...
ocudrReleaseName: '<UDR1 Helm name>'

```

#### ① Note

- \* To run the bulk import scenarios, disable **xmlToCsv**.
- \* Subscriber bulk import tool must always be clean installed.

### Subscriber Export tool Configuration

Configure the following parameters in the nudr-export-tool yaml file:

#### ① Note

To test the subscriber export tool, it is important to install the subscriber export tool as a separate helm deployment in the same namespace as that of UDR. For information about the subscriber export tool, see Subscriber Export Tool section in *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Ensure the secret is used to install the subscriber export tool is same as UDR1. The subscriber export tool should communicate with UDR1. Update the **nudr-export-tool deployment** section as follows

```

global:
 dockerRegistry: <docker-registry>:<docker port>/ocudr
 mysql:
 dbServiceName: "<DB Endpoint provided for UDR deployment>"
 port: "<PORT Used provided UDR deployment>"
 ...
 dbCredSecretName: '<UDR1 Secret>'
 ...
sftpDetails:
 ...
 sftpExportEnabled: false
 ...
 ...
serviceMeshCheck: true #if export tool is intended to be deployed with ASM
...
ocudrReleaseName: '<UDR1 Helm name>'
...

```

```
batch:
 corePoolSize: 1
 maxPoolSize: 1
exportMode: SLF_EXPORT
```

#### **Note**

Subscriber export tool must always be clean installed.

#### Other configuration details

- Configure Prometheus with UDR and Provisioning Gateway default alerts. For more information about UDR alerts, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide* and for Provisioning Gateway alerts, see *Oracle Communications Cloud Native Core, Provisioning Gateway Installation Guide*.
- Certificates used for Provisioning Gateway TLS should be provided in `/var/lib/jenkins/ocslf_tests/certs` in pod.
- Create Envoy Filter for both, UDR1 setup and ATS as described in [Enabling Service Mesh](#) for XFCC header validation scenarios and server header scenarios if installed with ASM.
- Create Peer Authentication as described in [Enabling Service Mesh](#) and *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*, if installed with ASM.

#### 4.8.1.1.2 Provisioning Gateway Custom Values File Configuration for SLF Pipelines

#### **Note**

For more information about Provisioning Gateway installation, see *Oracle Communications Cloud Native Core Provisioning Gateway Installation Guide*. The ProvGW custom-values.yaml file is available with documentation package on MOS.

- Update the global section of the ProvGW custom-values.yaml file as follows:

```
global:
 dockerRegistry: <docker-registry>:<docker port>/provgw
 mysql:
 dbServiceName: <db service to which UDR should be connected>
 port: &dbPortNumber "3306"
 configdbname: &configdbname <ProvGw DB name>
 dbname: &dbname <ProvGw DB name>
 ...
 createNetworkPolicy: true #only if network policy is supported in the
 environment. SLF pipeline to validate network policy supports only default
 policy
 ...
 serviceMeshCheck: &serviceMeshFlag true #if UDR is intended to be
 deployed with ASM.
 ...
 usageMode: "SLF"
 udr:
```

```

httpsEnabled: false
segDetails:
 - name: SEG-1
 fqdnValues: <UDR1 fqdn>,<UDR2 fqdn>
 preferred: <UDR1 fqdn>
 - name: SEG-2
 fqdnValues: <UDR3 fqdn>,<UDR4 fqdn>
 preferred: <UDR3 fqdn>

```

### ① Note

Database names provided in the Provisioning Gateway custom values files must match with the names created in the corresponding database secret.

- Database monitoring fqdn and port must provided under provgw-service:  
**provgw-service**

```

provgw-service:
...
 probes:
 dbStatusCheckUrl: "http://<DB Monitoring svc>:<DB monitoring port>/db-
tier/status/local"

```

- Enable the **auditor-service** parameter as follows:

```

auditor-service:
 enabled: true
...
 sftpDetails:
 secrets:
 privatekey:
 name: <private key secret created for SFTP from export tool>
 publickey:
 name: <public key secret created for SFTP from export tool>
...
 usePersistentVolume: true
...
 probes:
 dbStatusCheckUrl: "http://<DB Monitoring svc>:<DB monitoring port>/db-
tier/status/local"

```

- Configure **provgw-config** as below:

```

provgw-config:
...
 probes:
 dbStatusCheckUrl: "http://<DB Monitoring svc>:<DB monitoring port>/db-
tier/status/local"

```

- Under the **prov-ingressgateway annotations** section, set the **traffic.sidecar.istio.io/excludeInboundPorts** as '8443'.

Ensure the `ssl` section has secret names and namespace that are created as part of Transport Layer Security (TLS) support. Set the `initssl` and `enableIncomingHttps` parameters as 'true'.

```
prov-ingressgateway:
 deployment:
 customExtension:
 labels: {}
 annotations:
 traffic.sidecar.istio.io/excludeInboundPorts: "8443"
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <ProvGw Namespace>
 rsa:
 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

 certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <ProvGw Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

 caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <ProvGw namespace>
 fileName: <caBundle file name>

 keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <ProvGw namespace>
 fileName: <key txt file>

 trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <ProvGw namespace>
 fileName: <trust txt file>
 ...
 initssl: true
 ...
 enableIncomingHttps: true
```

#### Note

For information about Provisioning Gateway installation and ASM configuration for Provisioning Gateway, see *Oracle Communications Cloud Native Core Provisioning Gateway Installation Guide*. This section highlights the necessary changes that you are required to do in the Provisioning Gateway custom-values.yaml file to run ATS.

## 4.8.1.2 Prerequisites for UDR Pipelines

The prerequisites to run UDR pipelines are:

- Select any one of the following SUT deployment models depending on your requirements.
  - **Model 1: Single Site Deployment**  
To run UDR-NewFeatures, select **SUTSuite:** as 'All' and for UDR-Regression, select **SUTSuite:** 'ALL', Features: All (Except UDR\_Multisite). For this, you must deploy one UDR, one Provisioning Gateway in UDR mode, two diameter stubs, one Fourg stub, one NOTIFY Stub, two NRF-stubs, one SCP stub, and one ATS in same namespace.
  - **Model 2: Multiple Site Deployment:** To run UDR-Regression pipeline where **SUTSuite:** is 'MULTISITE', you must deploy:
    - \* Two UDRs in two different namespaces that has two different cnDBTier with database site replication enabled between the two.
    - \* ATS in any one namespace.
    - \* Namespace-1 with one UDR, two diameter stubs, one Fourg stub, one NOTIFY stub, two NRF-Stubs, one SCP stub, and one ATS.
    - \* Namespace-2 with one UDR.

### 4.8.1.2.1 UDR Custom Values File Configuration for UDR Pipelines

This section highlights only the necessary changes that you are required to do in the UDR custom-values.yaml file to run ATS.

#### Note

For information about UDR installation and ASM configuration for UDR, see the *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide* available on [My Oracle Support](#).

- Update the global section as follows:

```
global:
 dockerRegistry: <docker-registry>:<docker port>/ocudr
 mysql:
 dbServiceName: <db service to which UDR should be connected>
 port: &dbPortNumber "3306"
 configdbname: &configdbname <config db name>
 dbname: &dbname <subscriber db name>
 ...
 createNetworkPolicy: true #only if network policy is supported in the
environment. SLF pipeline to validate network policy supports only default
policy
 ...
 dbCredSecretName: <udr secret name>
 ...
 alternateRouteServiceEnable: true
 alternateRouteServiceHost: <udr-helm-release-name>-alternate-route
 performanceServiceEnable: true
 ...
```



```

autoCreateSubscriber: false
FourGPolicyConfiguration:
 ondemandMigration:
 enabled: &onDemandMigrationEnabled true
...
appInfoBaseUrl: "<udr-helm-release-name>-nudr-app-info:5906"
...
deploymentNrfClientService:
 envNfNamespace: <UDR namespace>
...
egressHttpsEnabled: &egressHttpsFlag true

```

#### Note

- Database names provided in the UDR custom values files must match with the names created in the corresponding database secret.
- The nfInstanceId for each of UDR must be unique for all model.

- Update the nudr-notify-service section as follows:

```

nudr-notify-service:
 ...
 notification:
 retrycount: "3"

```

- Update the ingressgateway-sig section as follows:

```

ingressgateway-sig:
 global:
 ...
 lciHeaderConfig:
 enabled: true
 loadThreshold: 30
 localLciHeaderValidity: 10000
 ...
 svcToSvcInstanceIdMapping:
 - svcName: "<udr-helm-release-name>-nudr-drservice"
 ...
 xfccHeaderValidation:
 validation:
 enabled: &xfccValidationEnabled false
 ...
 service:
 ...
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

```

```

certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <caBundle file name>

keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <key txt file>

trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <trust txt file>
...
serverHeaderDetails:
 enabled: true
...
initssl: true
...
oauthValidatorEnabled: &oauthEnabled true
...
nrfPublicKeyKubeSecret: <oauth secret name>
nrfPublicKeyKubeNamespace: <UDR namespace>
validationType: relaxed
...
...
enableIncomingHttps: true
...
userAgentHeaderValidationConfigMode: REST

```

- Update the ingressgateway-prov section as follows:

```

ingressgateway-prov:
 global:
 xfccHeaderValidation:
 validation:
 enabled: &xfccValidationEnabled false
 ...
 service:
 ...
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:

```

```

 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

 certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

 caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <caBundle file name>

 keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <key txt file>

 trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <trust txt file>
...
 serverHeaderDetails:
 enabled: false
...
 initssl: true
...
 oauthValidatorEnabled: &oauthEnabled true
...
 nrfPublicKeyKubeSecret: <oauth secret name>
 nrfPublicKeyKubeNamespace: <UDR namespace>
 validationType: relaxed
...
...
 enableIncomingHttps: true

```

- Update the alternate-route section as follows:

```

alternate-route:
 staticVirtualFqdns:
 - name: http://abc.test.com
 alternateFqdns:
 - target: <nrf stub1 fqdn>
 port: <nrf stub1 port>
 priority: 10
 - target: <nrf stub2 fqdn>
 port: <nrf stub2 port>
 priority: 20

```

- Update the nrf-client section as follows:

```
nrf-client:
 configmapApplicationConfig:
 ...
 profile:
 ...
 enableVirtualNrfResolution=true
 #virtualNrfFqdn value must be same as that of alternate-
route.staticVirtualFqdns.name[0] without the url scheme as shown below
 virtualNrfFqdn=abc.test.com
 ...

nrfRetryConfig=[{"serviceRequestType":"ALL_REQUESTS","primaryNRFRetryCount":
:1,"nonPrimaryNRFRetryCount":1,"alternateNRFRetryCount":1,"errorReasonsForF
ailure":
["503","504","500","SocketTimeoutException","JsonProcessingException","Unkn
ownHostException","NoRouteToHostException","IOException"],"gatewayErrorCodes":
["503"],"requestTimeout":10},
{"serviceRequestType":"AUTONOMOUS_NFREGISTER","primaryNRFRetryCount":1,"non
PrimaryNRFRetryCount":1,"alternateNRFRetryCount":-1,"errorReasonsForFailure
":
["503","504","500","SocketTimeoutException","JsonProcessingException","Unkn
ownHostException","NoRouteToHostException","IOException"],"gatewayErrorCodes":
["503"],"requestTimeout":10}]
...
nrf-client-nfmanagement:
 enablePDBSupport: false
```

- Update the egressgateway section to run SCP routing and TLS notification scenarios as follows:

```
egressgateway:
 ...
 global:
 lciHeaderConfig:
 enabled: true
 localLciHeaderValidity: 1000
 ...
 svcToSvcInstanceIdMapping:
 - svcName: "<udr-helm-release-name>-nudr-drservice"
 ...
 sbiRouting:
 peerConfiguration:
 - id: peer1
 host: '<scp stub svc and this should match with scp stub
expectedAuthorityHeader in scp stub custom values>'
 port: '<scp stub port>'
 apiPrefix: "/"
 peerSetConfiguration:
 - id: set0
 httpConfiguration:
 - priority: 1
 peerIdentifier: peer1
 ...
 routesConfig:
```

```

...
- id: scp_route
...
 path: /scp/**
 order: 2
 metadata:
 ...
 sbiRoutingEnabled: true
...
 initssl: true
 enableOutgoingHttps: *egressHttpsFlag
...
 service:
 ...
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

 certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

 caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <caBundle file name>

 keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <key txt file>

 trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <trust txt file>
 ...
 userAgentHeaderConfigMode: REST

```

- Update the perf-info section as follows:

```

perf-info:
 envMysqlDatabase: <configdb name used for UDR>
...
 configmapPerformance:
 prometheus: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-

```

```
name>/prometheus"
log:
 level:
 perfinfo: "WARN"
```

Formation of Prometheus URL is as below:

```
(env) [jenkins@ocats-udr-5c55f59d66-jkzml ~]$ curl -i http://oso-svr:80/
HTTP/1.1 302 Found
Content-Type: text/html; charset=utf-8
Location: /cne-23-1/prometheus
Date: Thu, 20 Apr 2023 06:28:48 GMT
Content-Length: 48
```

```
Found.
```

Prometheus url would be: `http://oso-svr:80/cne-23-1/prometheus`

#### Note

The values for namespace, container name and service name depend on CNE versions. For more information, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Update the app-info section as follows:

```
appinfo:
 ...
 dbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-tier/
status/local
 realtimeDbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc
port>/db-tier/status/cluster/local/realtime
 replicationUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-
tier/status/replication
 ...
 prometheusUrl: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-
name>/prometheus"
 alertmanagerUrl: "http://<alertmanager fqdn or IP>:<prometheus port>/
<domain-name>/alertmanager"
 ...
 fullnameOverride: <udr-helm-release-name>-app-info
```

- Configure the following parameters in the bulk-import tool yaml file:

#### Note

To test the bulk import tool, it is important to install the bulk import tool as part of UDR. For information about Bulk Import tool installation and sample bulk import yaml file, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Ensure the secret used to install bulk import tool is same as UDR. Configure the Bulk Import deployment values.yaml file as follows:

```
global:
 mysql:
 dbServiceName: "<DB Endpoint provided for UDR deployment>"
 port: "<PORT Used provided UDR deployment>"
 ...
 dbCredSecretName: '<UDR Secret>'
 ...
 batchConfig:
 asynchronous: false
 ...
 xmlToCsv:
 enabled: true
 ...
 ocudrReleaseName: '<UDR Helm name>'
```

- Do not change the containers name. It should be 'nudr-bulk-import' and 'nudr-xmltocsv'.

**Note**

Subscriber bulk import tool must always be clean installed.

- Configure the following parameters in the ocudr\_nudr\_migration\_tool\_custom\_values.yaml file:

**Note**

Migration tool must be installed as a separate helm deployment. For information about Migration tool installation, see "Migration Tool" section in *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Ensure the secret used to install the migration tool is same as UDR. Configure the ocudr\_nudr\_migration\_tool\_custom\_values.yaml file as follows:

```
global:
 mysql:
 dbServiceName: "<DB Endpoint provided for UDR deployment>"
 port: "<PORT Used provided UDR deployment>"
 ...
 dbCredSecretName: '<UDR Secret>'
 ...
 fourG:
 realm: "oracle.com"
 identity: "tekelec.oracle.com"
 nodePort: 3000
 nodeHost: "fourgstub-stub-service"
 maxConnectionRetry: 100
 ...
 hikari:
 queryTimeout: "-1"
```

```

...
ocudrReleaseName: '<UDR Helm name>'
...
sourceEndPointURL: 'http://10.75.229.43:3868'
udrServiceBaseURL: '<UDR Helm name>-ingressgateway-prov:80'
...
gracefulShutdown:
 gracePeriod: 30s
...

```

- Configure the following parameters in the `ocudr_nudr_export_tool_custom_values.yaml` file:

#### Note

The subscriber export tool must be installed as a separate helm deployment. For information about Migration tool installation, see "Subscriber Export Tool " section in *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Ensure the secret used to install the subscriber export tool is same as UDR. Configure the `ocudr_nudr_export_tool_custom_values.yaml` file as follows:

```

global:
 dockerRegistry: <docker-registry>:<docker port>/ocudr
mysql:
 dbServiceName: "<DB Endpoint provided for UDR deployment>"
 port: "<PORT Used provided UDR deployment>"
 ...
 dbCredSecretName: '<UDR Secret>'
 ...
createExportToolPVC:
 ...
 fileExtension: ".exml"
 ...
 ...
sftpDetails:
 ...
 sftpExportEnabled: false
 ...
 ...
ocudrReleaseName: '<UDR Helm name>'
 ...
batch:
 corePoolSize: 1
 maxPoolSize: 1
 exportMode: EXML_EXPORT

```

#### Note

Subscriber export tool must always be clean installed.

- Other configuration details:



- Configure Prometheus with UDR default alerts. For more information about UDR alerts, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.
- Certificates used for UDR-TLS should be provided in `/var/lib/jenkins/ocudr_tests/certs` in ATS pod as defined for `severheaderFlag` in `values.yaml` file (must be true during UDR installation). For more information, see [Installing ATS for UDR](#).

#### 4.8.1.2.2 Provisioning Gateway Custom Values File Configuration for UDR Pipelines

##### **Note**

For more information about Provisioning Gateway installation and ASM installation for Provisioning Gateway, see *Oracle Communications Cloud Native Core Provisioning Gateway Installation Guide*. The `ProvGW custom-values.yaml` file is available with documentation package on MOS. This section highlights only the necessary changes that are required in the `ProvGW custom-values.yaml` file to run ATS.

- Update the global section of the `ProvGW custom-values.yaml` file as follows:

```
global:
 dockerRegistry: <docker-registry>:<docker port>/provgw
 mysql:
 dbServiceName: <db service to which UDR should be connected>
 port: &dbPortNumber "3306"
 configdbname: &configdbname <ProvGw DB name>
 dbname: &dbname <ProvGw DB name>
 ...
 createNetworkPolicy: true #only if network policy is supported in the
 environment. SLF pipeline to validate network policy supports only default
 policy
 ...
 usageMode: "UDR"
```

##### **Note**

Database names provided in the Provisioning Gateway custom values files must match with the names created in the corresponding database secret.

- Enable the `provgw-service` parameter as follows:

```
provgw-service:
 ...
 soapService:
 udrIp: <UDR-Prov fqdn>
 udrSignallingIp: <UDR-Sig Fqdn>
 convertToSec: false
 secEntity: |
 - name: QuotaEntity
 elementString: usage
 innerElemenString: quota
 outerFields:
```

```
- version
- quota
- SequenceNumber
innerFields:
 - name
 - cid
 - time
 - totalVolume
 - inputVolume
 - outputVolume
 - serviceSpecific
 - nextResetTime
 - Type
 - grantedTotalVolume
 - grantedInputVolume
 - grantedOutputVolume
 - grantedTime
 - grantedServiceSpecific
 - QuotaState
 - RefInstanceId
 - custom1
- name: DynamicQuotaEntity
elementString: definition
innerElemenString: DynamicQuota
outerFields:
 - version
 - DynamicQuota
 - SequenceNumber
innerFields:
 - Type
 - name
 - InstanceId
 - Priority
 - InitialTime
 - InitialTotalVolume
 - InitialInputVolume
 - InitialOutputVolume
 - InitialServiceSpecific
 - activationdatetime
 - expirationdatetime
 - purchaseddatetime
 - Duration
 - InterimReportingInterval
 - custom1
- name: Subscriber
elementString: subscriber
outerFields:
 - IMSI
 - MSISDN
 - NAI
 - EXTID
 - ACCOUNTID
 - BillingDay
 - Entitlement
 - Tier
 - SequenceNumber
```

```

 - custom1
 - custom2
 - custom3
 - custom4
 - custom5
 - custom6
 - custom7
 - custom8
 - custom9
 - custom10
 - custom11
 - custom12
 - custom13
 - custom14
 - custom15
 - name: StateEntity
 elementString: state
 innerElemenString: property
 outerFields:
 - version
 - property
 - SequenceNumber
 innerFields:
 - name
 - value
 - name: CustomEntity
 elementString: customusage
 innerElemenString: customent
 outerFields:
 - version
 - customent
 - SequenceNumber
 innerFields:
 - name
 - custcid
 - custtotal
 ...
 probes:
 dbStatusCheckUrl: "http://<DB Monitoring svc>:<DB monitoring port>/db-
tier/status/local"

```

- Disable auditor-service and configure auditor-service section as follows:

```

auditor-service:
 enabled: false

```

- Enable the **provgw-config** section as below:

```

provgw-config:
 ...
 probes:
 dbStatusCheckUrl: "http://<DB Monitoring svc>:<DB monitoring port>/db-
tier/status/local"

```

- Under the **prov-ingressgateway annotations** section, set the **routesConfig** for provgw-service port as '62001'. Ensure the ssl section has secret names and namespace that are

created as part of Transport Layer Security (TLS) support. Set the **initssl** and **enableIncomingHttps** parameters as 'true'.

```
prov-ingressgateway:
 deployment:
 ...
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

 certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

 caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <caBundle file name>

 keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <key txt file>

 trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <trust txt file>
 ...
 initssl: true
 ...
 enableIncomingHttps: true
 ...
 routesConfig:
 - id: traffic_mapping_rest_group_prov
 uri: http://{ .Release.Name }-provgw-service:62001
```

#### Note

For information about Provisioning Gateway installation and ASM configuration for Provisioning Gateway, see *Oracle Communications Cloud Native Core Provisioning Gateway Installation Guide*. This section highlights the necessary changes that you are required to do in the Provisioning Gateway custom-values.yaml file to run ATS.

### 4.8.1.3 Prerequisites for EIR Pipelines

To run EIR Pipelines, you must deploy one EIR, two NRF-stubs, one ATS, subscriber bulk import tool, and subscriber export tool in same namespace.

#### 4.8.1.3.1 UDR Custom Values File Configuration for EIR Pipelines

##### **Note**

Deploy UDR in the EIR mode. For more information about UDR installation, see *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*. This section highlights only the necessary changes that you are required to do in the UDR custom-values.yaml file to run ATS.

- Update the global section as follows:

```
global:
 dockerRegistry: <docker-registry>:<docker port>/ocudr
 mysql:
 dbServiceName: <db service to which UDR should be connected>
 port: &dbPortNumber "3306"
 configdbname: &configdbname <config db name>
 dbname: &dbname <subscriber db name>
 ...
 dbCredSecretName: <udr secret name>
 ...
 alternateRouteServiceEnable: true
 alternateRouteServiceHost: <udr-helm-release-name>-alternate-route
 ...
 performanceServiceEnable: true
 ...
 udrServices: "n5g-eir-eic"
 ...
 diamGatewayEnable: false
 ...
 appInfoBaseUrl: "<udr-helm-release-name>-nudr-app-info:5906"
 ...
 nrfClientNfManagementEnable: true # value is true for UDR1 deployment;
 UDR2, UDR3 and UDR4 should be deployed with false
 deploymentNrfClientService:
 envNfNamespace: <namespace where UDR is to be deployed>
```

##### **Note**

Database names provided in the Provisioning Gateway custom values files must match with the names created in the corresponding database secret.

- Update the config-server as follows:

```
config-server:
 fullnameOverride: <udr-helm-release-name>-config-server
```

- Update the ingressgateway-sig as follows:

```
ingressgateway-sig:
 global:
 ...
 lciHeaderConfig:
 enabled: true
 loadThreshold: 30
 localLciHeaderValidity: 10000
 ...
 svcToSvcInstanceIdMapping:
 - svcName: "<udr-helm-release-name>-nudr-drservice"
 ...
 xfccHeaderValidation:
 validation:
 enabled: &xfccValidationEnabled true
 ...
 service:
 ...
 ssl:
 tlsVersion: TLSv1.2
 privateKey:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA private key pem file>
 ecdsa:
 fileName: <ECDSA private key pem file>

 certificate:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR Namespace>
 rsa:
 fileName: <RSA certificate>
 ecdsa:
 fileName: <ECDSA certificate>

 caBundle:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <caBundle file name>

 keyStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <key txt file>

 trustStorePassword:
 k8SecretName: <TLS support secret name>
 k8Namespace: <UDR namespace>
 fileName: <trust txt file>
```

- ```

...
serverHeaderDetails:
  enabled: true
...
initssl: true
...
oauthValidatorEnabled: &oauthEnabled true
nfType: 5G_EIR
producerScope: n5g-eir-eic
...
nrfPublicKeyKubeSecret: <oauth secret name>
nrfPublicKeyKubeNamespace: <UDR namespace>
validationType: relaxed
...
...
enableIncomingHttps: true
...
userAgentHeaderValidationConfigMode: REST

```
- Update the ingressgateway-prov as follows:

```

ingressgateway-prov:
  global:
    xfccHeaderValidation:
      validation:
        enabled: &xfccValidationEnabled false
...
serverHeaderDetails:
  enabled: false
...
initssl: true
...
oauthValidatorEnabled: &oauthEnabled true
nfType: 5G_EIR
producerScope: n5g-eir-prov
...
nrfPublicKeyKubeSecret: <oauth secret name>
nrfPublicKeyKubeNamespace: <UDR namespace>
validationType: relaxed
...
...
enableIncomingHttps: true

```
 - Configure targets and ports in the alternate-route.staticVirtualFqdns section with the NRF stubs FQDNs and ports.

```

alternate-route:
  staticVirtualFqdns:
    - name: http://abc.test.com
      alternateFqdns:
        - target: <nrf stub1 fqdn>
          port: <nrf stub1 port>
          priority: 10
        - target: <nrf stub2 fqdn>

```

```
port: <nrf stub2 port>
priority: 20
```

- Update the appinfo section as follows:

```
appinfo:
...
  dbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-tier/
status/local
  realtimeDbStatusUri: http://<DB monitoring SVC>:<DB monitoring svc
port>/db-tier/status/cluster/local/realtime
  replicationUri: http://<DB monitoring SVC>:<DB monitoring svc port>/db-
tier/status/replication
...
  prometheusUrl: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-
name>/prometheus"
  alertmanagerUrl: "http://<alertmanager fqdn or IP>:<prometheus port>/
<domain-name>/alertmanager"
...
  core_services:
    5g_eir: *eir
  fullnameOverride: <udr-helm-release-name>-app-info
```

- Update the nrf-client section as follows:

```
nrf-client:
  profile: |
    #uncomment appProfiles that is dedicated to used for EIR mode (n5g-eir-
eic) and comment the rest of the appProfiles.
    enableVirtualNrfResolution=true
    #virtualNrfFqdn value must be same as that of alternate-
route.staticVirtualFqdns.name[0] without the url scheme as shown below
    virtualNrfFqdn=abc.test.com
  nrf-client-nfmanagement:
    ...
    replicas: 1
    ...
    enablePDBSupport: false
```

- Update the perf-info section as follows:

```
pperf-info:
  envMySQLDatabase: <configdb name used for UDR>
...
  configmapPerformance:
    prometheus: "http://<Prometheus fqdn or IP>:<prometheus port>/<domain-
name>/prometheus"
  log:
    level:
      perfinfo: "WARN"
```


Note

The values for namespace, container name, and servicename depend on CNE or OSO versions. For information, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*..

- Update the egressgateway section as follows:

```
egressgateway:
  userAgentHeaderConfigMode: REST
```

- Configure the following parameters in the bulk-import tool yaml file:

Note

To test the subscriber bulk import tool, it is important to install the bulk import tool as a separate helm deployment in the same namespace as that of EIR. For information about the bulk import tool installation and sample bulk import yaml file, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*..

- Ensure the secret is used to install subscriber bulk import tool is same as EIR. Configure the Bulk Import deployment values.yaml file as follows:

```
global:
  mysql:
    dbServiceName: "<DB Endpoint provided for UDR deployment>"
    port: "<PORT Used provided UDR deployment>"
    ...
    dbCredSecretName: '<EIR Secret>'
    ...
  batchConfig:
    asynchronous: false
    ...
  xmlToCsv:
    enabled: false
    ...
  pdbiToCsv:
    enabled: true
    ...
  ocudrReleaseName: '<EIRHelm name>'
```

Note

Subscriber bulk import tool must always be clean installed.

- Configure the following parameters in the nudr-export-tool deployment yaml file:

Note

To test the subscriber export tool, it is important to install the subscriber export tool as a separate helm deployment in the same namespace as that of UDR. For information about the subscriber export tool installation, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

- Ensure the secret is used to install subscriber export tool is same as EIR. Configure the subscriber export tool deployment values.yaml file as follows:

```
global:
  dockerRegistry: <docker-registry>:<docker port>/ocudr
  mysql:
    dbServiceName: "<DB Endpoint provided for EIR deployment>"
    port: "<PORT Used provided UDR deployment>"
    ...
    dbCredSecretName: '<EIR Secret>'
    ...
  sftpDetails:
    ...
    sftpExportEnabled: false
    ...
  ...
  ocudrReleaseName: '<EIR Helm name>'
  ...
  exportMode: EIR_EXPORT
```

Note

Subscriber export tool must always be clean installed.

- Other configuration details:
 - Configure Prometheus with EIR default alerts. For more information about EIR alerts, see *Oracle Communications Cloud Native Core, Unified Data Repository User Guide*.

4.8.2 UDR Golden Configuration Support

The ATS golden configuration feature will facilitate the provisioning of customer specific configuration to run test cases. UDR ATS has been integrated with the ATS golden configuration feature. This feature enables the customer to provide their own configuration and save the configuration once the ATS runs are completed.

When running a test, ATS detects the existence of a <feature_name>.yaml configuration file within the <conf_type>_config directory. ATS will populate the parameterized steps and data files with the values present in the global and feature-specific configuration files. The two configuration types are:

- Product configuration
- Custom configuration

You must use the custom_config directory files for provisioning the golden configuration data.

To Configure ATS Golden Configuration

- Modify the parameter values in the /ocslf_tests/custom_config/global.yaml and/or feature config file (/ocslf_tests/custom_config/<feature>.yaml).
- Select the custom_config in the jenkins pipeline GUI to run the test cases with data provided in the custom_config directory.

ATS Golden Configuration Directory

ATS golden configuration uses the below directories located within the /var/lib/jenkins/ocslf_tests directory:

- Parameterized features
 - features/
 - * cust_newfeatures/
 - * cust_regression/Singlesite/
- Configuration files
 - custom_config/
 - * yaml
 - * <feature_name>.yaml
- Test data
 - cust_data/

Parameterized features

New and regression feature configuration files use the same custom_config directory.

ATS Golden Configuration Feature Configuration files

This section describes the ATS golden configuration files:

- The global.yaml configuration file contains Key-Value pair parameters. The Key-Value pair parameters applies to more than one parameterized feature file.
- The Key-Value pair parameters in the <feature>.yaml configuration file includes the following fields:
 - subscriber profile data information
 - cust_data payload file values
 - data folder name
 - other feature specific fields
- The <feature>.yaml file is divided into scenario specific sections, containing Key-Value pair parameters. Each parameter in the parameterized feature file takes values only from its own scenario File_Parameters dictionary. You can use the same parameter name in more than one scenario with different scenario specific values.
- When goldenConfig.py is running, each feature config yaml contains the global section at the top between #START_GLOBAL and #END_GLOBAL strings.
- Each SLF <feature>.yaml file allows to make the following changes, if applicable:

- Use global.yaml anchors to copy the values to scenario specific sections that uses the yalm alias. The data folder name is taken from the global section and is used when a behave step is needed to open a json payload file.
- Provisioning the subscriber profile data.
- Overwrite data_folder payload file values. The payload file values will be overwritten in the specified path. The golden configuration feature will overwrite data string with cust_data even if each path is written with the data directory.
- Provisioning custom values to feature specific fields.

Test data

The test data files includes the json payload files used by each feature. A new cust_data directory, which is a copy of the original data directory, is created to maintain the data directory with product configuration unchanged because the golden configuration feature allows to overwrite data in these json files.

ATS Golden Configuration backup for SLF and Provisioning Gateway

The SLF and Provisioning Gateway backup process is implemented in the before_all and after_all behave hooks to allow ATS pipeline runs to implement feature specific configuration and revert them to the customer specific golden configuration at the end.

ATS Golden Configuration backup Considerations for SLF:

- Backup will be automatically executed for SLF and Provisioning Gateway deployments.
- If failure to export SLF or Provisioning Gateway golden configuration, pipeline execution will be aborted and feature test will not be performed.
- If you fail to import SLF or Provisioning Gateway golden configuration then you are required to manually backup by using 'slfgoldenconfig.zip' and 'provgwgoldenconfig.zip' that is located in ATS pod in the /var/lib/jenkins/ocslf_tests/ directory.

Selective Run of Features and Scenarios

For SLF ATS golden configuration support, test execution of certain features is depended on the values set for fields in the ocslf_tests/custom_config/global.yaml file. The values provided in the global.yaml must match with the SUT configurations (SLF values.yaml).

```
#START_GLOBAL
global:
  data_folder: &data_folder cust_data
  # To select an appropriate retries value take from SLF values.yaml the
  localLciHeaderValidity value (value in
  # milliseconds) and perform the following formula:
  (localLciHeaderValidity/5000)+1.
  # It is recommended to use 6 as the minimum retries value for igw-sig-
  lciHeaderRequestRetries.
  igw-sig-lciHeaderRequestRetries: &igw-sig-lciHeaderRequestRetries '6'
  # Values provided in this global.yaml should match with the SUT
  configurations (SLF values.yaml).
  alternateRouteServiceEnable: true #true/false
  performanceServiceEnable: true #true/false
  igw-sig-lciheader-enabled: true #true/false
  igw-sig-oauth-enabled: true #true/false
  igw-sig-userAgentHeaderValidationConfigMode: REST #HELM/REST
  igw-sig-userAgentHeaderValidation-enabled: true #true/false
```

```

    igw-sig-userAgentHeader-validationType: &igw-sig-userAgentHeader-
validationType relaxed #relaxed/strict
    igw-prov-oauth-enabled: true #true/false
    nrf-client-enableVirtualNrfResolution: true #true/false
    egw-userAgentHeaderConfigMode: REST #HELM/REST
    egw-userAgentHeader-enabled: true #true/false
    egw-addFqdnToHeader: true #true/false
    egressgateway_nfFqdn: &egressgateway_nfFqdn udr.oracle.com
#END_GLOBAL

```

List of Parameterized Features

The below tables provide the list of parameterized features for EIR, SLF, and UDR:

Table 4-12 Parameterized Features for EIR

EIR Feature
Prometheus_Scenarios.feature
EIR_XFCC.feature
EIR_User_Agent_Header.feature
EIR_User_AgentHeader_Nrf.feature
EIR_TLS_Signalling.feature
EIR_TLS_Provisioning.feature
EIR_Subscriber_Prov.feature
EIR_Signalling.feature
EIR_Server_Header.feature
EIR_Provisioning_Err.feature
EIR_PDBI_Import.feature
EIR_OAuth2_Signalling.feature
EIR_OAuth2_Provisioning.feature
EIR_MGM_API.feature
EIR_Default_Response.feature
EIR_CSV_Import.feature
EIR_AccessLog.feature
EIR_IMEISV.feature
EIR_IMEISV_Default_Response.feature
EIR_IMEISV_AccessLog.feature
EIR_PDBI_IMEISV_Import.feature
EIR_Subscriber_Export_Tool.feature
EIR_22_4_x_Patch_PDBI_Import.feature
EIR_Validation_Additional_Scenarios.feature
EIR_PDBI_Synchronous_Scenarios.feature
EIR_Subscriber_Export_Tool_New.feature

Table 4-13 Parameterized Features for SLF

SLF Feature
SLF_BulkImport.feature
ProvGw_Config_ImportExport.feature

Table 4-13 (Cont.) Parameterized Features for SLF

SLF Feature
SLF_CHFGroupId.feature
SLF_DefaultGroupId.feature
SLF_User_Agent_Header.feature
SLF_LCI_Header.feature
SLF_Prov_OAuth2_Validation.feature
SLF_User_AgentHeader_Nrf.feature
SLF_LCI_Via_Header.feature
SLF_Subscriber_Activity_Logging.feature
SLF_Subscriber_Export_Tool.feature
SLF_Singlesite_NFCS_BulkImport.feature
SLF_Singlesite_NFCS_Nrf_Client_Scenarios.feature
SLF_Singlesite_NFCS_Prometheus_Scenarios.feature
SLF_Singlesite_NFCS_Scenarios.feature
SLF_Singlesite_NFCS_Subscriber_Export_Tool.feature
ProvGw_Auditor_Status_API.feature
SLF_NF_Scoring_Multisite.feature
SLF_Network_Policy_Validation.feature

Table 4-14 Parameterized Features for UDR

UDR Feature
UDR_LCI_Header_Via.feature
UDR_Subscriber_Activity_Logging.feature
UDR_VSA_PCF_Validation.feature
UDR_PCF_ETag_Validation.feature
UDR_PCF_ETag_Notify_Validation.feature
UDR_AUTO_Enroll_policy_Data.feature
UDR_22.4.x_PCF_VSA_GET_DELETE.feature
ProvGw_SOAP_Additional_scenarios.feature
23_1_1_ProvGw_fixes.feature
UDR_Mgm_Attribute_Mapping.feature
UDR_PCF_Auto_Enrollment_New.feature
UDR_ProvGw_Prov-Headers.feature
UDR_Quota_Migration_Policy_Sh_Soap.feature
UDR_Network_Policy_Validation.feature
UDR_VSA_Delete_All_Validation.feature
UDR_Quota_Migration_Sh_interface.feature
UDR_Quota_Migration_OnDemand_VSA.feature
UDR_Quota_Migration_OnDemand_Sh.feature
UDR_Quota_Migration_OnDemand_SOAP_ProvPolicy.feature
UDR_Quota_Migration_OnDemand_NudrPolicy.feature
UDR_Quota_Migration_OnDemand_Misc.feature
UDR_Network_Policy_Validation.feature

4.8.3 UDR Application Log Collection

Application log collection enables ATS to collect logs from System Under Test (SUT) when there are scenario failures. User can set log level to one of values mentioned below:

- WARN
- INFO
- DEBUG
- ERROR
- TRACE

Application log collection is enabled or disabled from the Jenkins Graphical User Interface (GUI). By default application log collection is disabled. When the application log collection is enabled you must follow the ATS Persistent Volume Claim (PVC) guidelines to deploy the ATS with appropriate PVC size. Depending on the log type, the PVC size for the ATS will vary. For storing PCAP logs ATS requires more PVC size than the application logs. Logs can be collected for application logs or PCAP logs or both based on the selection provided through the Jenkins GUI. For more information, see [Application Log Collection](#) and [PCAP Log Collection](#).

Note

- If the application log collection feature is enabled, it will increase the execution time of the pipeline because some of the steps are executed before and after each scenario. It is recommended to enable the application log collection only if there are consistent failures and pod logs are required to further debug the issues.
- Do not provide any values for `Values.k8sResource.container.prefix` and `Values.k8sResource.container.suffix` in the UDR custom values file and do not modify debug tools container name when PCAP logs are collected.

Configuration

To configure application log collection before installing ATS follow the steps below:

1. ATS must be deployed with PVC enabled. For more information on PVC configurations. For more information, see [Enabling Persistent Volume](#).
2. For PVC size refer to [ATS PVC Size Calculation](#).
3. SUT must be deployed with debug tools to capture PCAP logs on SLF, UDR, and EIR. For more information, see Pod Security Policy, Role, and Role Binding Creation section in *Oracle Communications Cloud Native Core, Unified Data Repository Installation, Upgrade, and Fault Recovery Guide*.
4. To enable a debug tool containers for UDR and Provision Gateway edit the global section of the custom-values.yaml file of UDR and Provision Gateway. For more information, see Using Debug Tools in *Oracle Communications Cloud Native Core, Unified Data Repository Troubleshooting Guide* and *Oracle Communications Cloud Native Core, Provisioning Gateway Guide*.

To configure application log collection after installing ATS follow the steps below:

1. When the ATS and SUT are installed enable the log collection from the ATS Jenkins GUI. This is available on each pipelines on the BuildWithParameters screen of the UDR ATS. This is currently provided only for NewFeatures and Regression Pipelines.

Figure 4-84 SLF-NewFeatures BuildWithParameters

Dashboard > SLF-NewFeatures >

- SLF_Singlesite_NFCS_Scenarios_01
- SLF_Singlesite_NFCS_Scenarios_02_SBI_Incoming
- SLF_Singlesite_NFCS_Scenarios_03_Prov_Incoming
- SLF_Singlesite_NFCS_Scenarios_04_operationalHistory
- SLF_Singlesite_NFCS_Scenarios_05_Err_operationalstate
- SLF_Singlesite_NFCS_Scenarios_06_Err
- SLF_Singlesite_NFCS_Scenarios_07_Err_config
- SLF_Singlesite_NFCS_Scenarios_08_metrics
- SLF_Singlesite_NFCS_Subscriber_Export_Tool
- SLF_Singlesite_NFCS_Subscriber_Export_Tool_01

Fetch_Log_Upon_Failure
YES

Log_Type
☒ AppLog
☐ PcapLog [Debug Container Should be Running]

Log_Level
WARN

Build

2. To enable the application log collection follow the below steps
 - a. From the **Fetch_Log_Upon_Failure** select **YES** from the dropdown list.
 - b. From the **Log_Type** select the required options, **AppLog** or **PcapLog** or both. Follow the below steps depending on the selection and requirement:
 - i. ATS must be deployed with the corresponding PVC size if **AppLog** is selected from the **Log_Type**.
 - ii. SUT must be deployed with debug tools container and ATS must be deployed with corresponding PVC size if **PcapLog** is selected from the **Log_Type** or if both **AppLog** and **PcapLog** are selected.
3. Application log collection provides **Log_Level** values from the dropdown. The log level can be set for all the microservices to capture the SUT log for failure. The possible values for **Log_Level** are as follows:
 - WARN
 - INFO
 - DEBUG
 - ERROR
 - TRACE

The captured logs for AppLogs, PcapLogs, or both are stored on the PVC and can be accessed from the ATS builds path as follows:

- Run the following command to log in to the ATS pod:

```
$kubectl exec -it <ATS_POD> -n <NAMESPACE> bash
```

- Select the builds path to capture AppLogs and PcapLogs :

```
(env) [jenkins@slfats-ocats-udr-56f76cd668-jzj4d ~]$cd /var/lib/
jenkins/.jenkins/jobs/<JOB-NAME>/builds/<BUILD-NUM>/
```

```
(env) [jenkins@slfats-ocats-udr-56f76cd668-jzj4d ~]$ls -lrt
```


ATS PVC Guidelines

ATS must be deployed with required PVC storage to store the application logs collected during the ATS pipeline runs for the failed scenarios. The PVC size can vary depending on the following selections:

- ATS_Type, Pipeline_Type
- Log_Type
- Log_Level
- Number_of_pipeline_runs

The PVC requirement size per scenario for ATS_Type (SLF-ATS, UDR-ATS, and EIR-ATS) and Log_Type (AppLogs, PcapLogs, or both), and the minimum memory requirements to store the logs on the PVC are given below:

1. SLF-ATS:
 - AppLogs only: Per failure scenario: 72MB
 - PcapLogs only: Per failure scenario: 554MB
 - AppLogs and PcapLogs: Per failure scenario: 626MB
2. UDR-ATS:
 - AppLogs only: Per failure scenario: 1MB
 - PcapLogs only: Per failure scenario: 12GB
 - AppLogs and PcapLogs: Per failure scenario: 13GB
3. EIR-ATS:
 - AppLogs only: Per failure scenario: 500KB
 - PcapLogs only: Per failure scenario: 10GB
 - AppLogs and PcapLogs: Per failure scenario: 11GB

ATS PVC Size Calculation

The maximum PVC size for a given ATS_Type, Pipeline_Type, and Log_Type are calculated using the below formula:

Note

The maximum PVC size must be able to store the logs even if all the scenarios fail in all the runs.

$\text{PVC\_size} = \text{Number of scenarios in Pipeline Type} * \text{PVC requirement per scenario for given type [ATS\_Type or Log\_Type]} * \text{Total runs to be executed.}$

Example 1: To capture the application logs for SLF-ATS-NewFeatures and the Log_Type is set as AppLogs only for 10 pipeline runs, the PVC for ATS is calculated as:

The given SLF-ATS-NewFeatures has 19 scenarios. The memory requirement per scenario is 72MB.

$\text{PVC\_size} = 19 * 72\text{MB} * 10 = 13\text{GB}$. The maximum PVC size required for ATS is 13 GB.

Example 2: To capture the application logs for EIR-ATS-Regression and the Log_Type is set as PcapLogs only for 1 pipeline run, the PVC for ATS is calculated as:

The given EIR-ATS-Regression has 242 scenarios. The memory requirement per scenario is 10GB.

$PVC\_size = 242 * 10GB * 1 = 2420GB$. The maximum PVC size required for ATS is 2420 GB.

4.8.4 Parallel Test Execution

Parallel test execution allows you to perform multiple logically grouped tests simultaneously on the same System Under Test (SUT) to reduce the overall execution time of ATS.

ATS currently runs all its tests in a sequential manner, which is time-consuming. With parallel test execution, tests can be run concurrently rather than sequentially or one at a time. Test cases or feature files are now separated into different folders, such as stages and groups, for concurrent test execution. Different stages, such as stage 1, stage 2, and stage 3, run the test cases in a sequential order, and each stage has its own set of groups. For more information about Parallel Test Execution, see [Parallel Test Execution](#).

SLF ATS Stages and Groups

Following table provides different stages and groups for SLF ATS:

Table 4-15 SLF ATS Stages and Groups

Single Site Stages and Groups	Multiple Site Stages and Groups
Singlesite/ ├── Stage_1 ├── bulk_import_basics ├── behave.ini └── SLF_BulkImport.feature ├── export_tool_basics ├── behave.ini └── SLF_Subscriber_Export_Tool.feature ├── independent_features ├── behave.ini └── SLF_Network_Policy_Validation.feature ├── SLF_Sbi_Correlation_Header.feature ├── ingressgateway_features ├── behave.ini └── SLF_Ingress_XFCC_Header.feature ├── SLF_LCI_Via_Header.feature ├── SLF_OAuth2_Validation.feature ├── SLF_Prov_OAuth2_Validation.feature ├── SLF_Server_Header.feature ├── nrf-client └── behave.ini	Multisite/ ├── Multi_Site ├── two_site ├── behave.ini ├── SLF_Multisite.feature ├── SLF_Multisite_NFCS_Scenarios.feature └── SLF_NF_Scoring_Multisite.feature

Table 4-15 (Cont.) SLF ATS Stages and Groups

Single Site Stages and Groups	Multiple Site Stages and Groups
<pre> └── Prometheus_Scenarios_NrfClient.feature └── SLF_Dns_Srv.feature └── SLF_Nrf_Client_Scenarios.feature └── SLF_User_AgentHeader_Nrf.feature └── provgw └── behave.ini └── Prometheus_Scenarios_ProvGw.feature └── ProvGw_Basic_Func_Error.feature └── ProvGw_Basic_Func_Success.feature └── ProvGw_CHFGroupId.feature └── ProvGw_Data_Mismatch.feature └── ProvGw_SSL_Enabled.feature └── slf_basics └── behave.ini └── SLF_CHFGroupId.feature └── SLF_Prov_and_Lookup_NonMetric.feature └── Stage_2 └── provgw_config └── behave.ini </pre>	

Table 4-15 (Cont.) SLF ATS Stages and Groups

Single Site Stages and Groups	Multiple Site Stages and Groups
└─ Provgw_Config.feature └─ slf_config └─ behave.ini └─ - SLF_DefaultGroupId.feature └─ SLF_Mgm_Apis.feature └─ - SLF_Subscriber_Activity_Logging.feature └─ Stage_3 └─ sequential_execution └─ behave.ini └─ - Prometheus_Scenarios.feature └─ ProvgW_Auditor.feature └─ - ProvgW_Auditor_Status_API.feature └─ - ProvgW_Config_ImportExport.feature └─ ProvgW_Loglevel.feature └─ ProvgW_Segdown.feature └─ - SLF_Config_Import_Export.feature └─ SLF_LCI_Header.feature └─ SLF_LogLevels.feature └─ SLF_NF_Scoring.feature └─ - SLF_Prov_and_Lookup_Metric.feature	

Table 4-15 (Cont.) SLF ATS Stages and Groups

Single Site Stages and Groups	Multiple Site Stages and Groups
SLF_Singlesite_NFCS_BulkImport.feature SLF_Singlesite_NFCS_Nrf_Client_Scenarios.feature SLF_Singlesite_NFCS_Prometheus_Scenarios.feature SLF_Singlesite_NFCS_Scenarios.feature SLF_Singlesite_NFCS_Subscriber_Export_Tool.feature SLF_Subscriber_Activity_Logging_Loglevels.feature SLF_User_Agent_Header.feature	

UDR ATS Stages and Groups

Following table provides different stages and groups for UDR ATS:

Table 4-16 UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
Singlesite/ └─ All └─ Stage_1 sequential_execution behave.ini UDR_Bulk_Import_PNR_Vali dation.feature UDR_PUR_Scenarios.featur e UDR_Quota_Dq_Additional_ Scenarios.feature UDR_Sm_Data_Dnn_Schema_V alidation.feature UDR_Subscriber_Export_To ol.feature └─ MISCELLANEOUS └─ Stage_1 sequential_execution behave.ini UDR_Bulk_Import_PNR_Vali dation.feature UDR_PUR_Scenarios.featur e	Singlesite/ └─ All └─ Stage_1 bulk_import_metrics_auto create_true behave.ini UDR_Bulk_Import.feature UDR_Bulk_Import_ixml_par t1.feature UDR_Bulk_Import_PolicyAn dVsa.feature UDR_ProvGw_Prov-Headers. feature diameter_only_metrics behave.ini Diameter_All_Entities.fe ature Diameter_Config.feature Diameter_Misc.feature Diameter_User_Data.featu re UDR_22_4_1_Diameter.fe ature UDR_23_1_0_Diameter.fe ature UDR_23_2_0_Diameter.fe ature UDR_All_Migration_State_ Entity_part1.feature UDR_Migration_tool_Addit ional_scenarios.feature	Multisite/ └─ Multi_Site └─ two_site └─ behave.ini UDR_Multisite.feature

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
UDR_Quota_Dq_Additional_Scenarios.feature	UDR_Migration_tool_part1.feature nrf_only_metrics	
UDR_Sm_Data_Dnn_Schema_Validation.feature	behave.ini UDR_NRFclient_DNS_Srv.feature	
UDR_Subscriber_Export_Tool.feature	UDR_NRFclient_Scenarios.feature UDR_User_AgentHeader_Nrf.feature oauth2_only_metrics behave.ini UDR_OAuth2_Validation.feature UDR_PCF_Prov_OAuth2_Validation.feature UDR_UDM_Prov_OAuth2_Validation.feature pcf_subscription_no_metrics behave.ini UDR_PCF_DELETE.feature UDR_PCF_SMDData_Patch.feature UDR_Subscriber_Policy_Data.feature UDR_UDM_API_SDM_Subscription.feature UDR_UDM_Subscription_Data_Subscription.feature vsa_no_metrics	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	 behave.ini UDR_22.4.x_PCF_VSA_GET_D ELETE.feature UDR_VSA_PATCH_Fix.featur e UDR_VSA_PCF_Validation.f eature Stage_2 bulk_import_autocreate_f alse behave.ini UDR_Bulk_Import_ixml_par t2.feature no_igw_dr_metrics behave.ini UDR_PCF_ETag_Notify_Vali dation.feature UDR_PCF_Notification_New _API.feature UDR_PCF_User_Agent_Heade r.feature UDR_Suppressed_Notificat ions.feature UDR_UDM_Notification.fea ture UDR_UDM_UserAgent_Header .feature UDR_User_AgentHeader_Not ification.feature Stage_3 provgw_no_igw_dr_metrics 	

Table 4-16 (Cont.) UDR ATS Stages and Groups

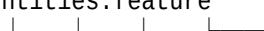
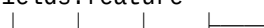
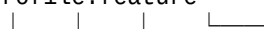
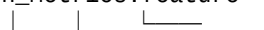
UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	23_1_1_ProvGw_fixes.feature  behave.ini  ProvGw_SOAP_Additional_s cenarios.feature  ProvGw_SOAP_CNCEsUDR82_m isc.feature  ProvGw_SOAP_ConvertToSec _strict.feature  ProvGw_SOAP_Custom_Entit ies.feature  ProvGw_SOAP_Custom_Field s.feature  ProvGw_SOAP_Format_Tags. feature  ProvGw_SOAP_GetSubscript ion.feature  ProvGw_SOAP_Misc.feature  ProvGw_SOAP_Reset_Quota. feature  ProvGw_SOAP_SSL.feature  ProvGw_SOAP_Subscriber_E ntities.feature  ProvGw_SOAP_Subscriber_E ntity_Fields.feature  ProvGw_SOAP_Subscriber_F ields.feature  ProvGw_SOAP_Subscriber_P rofile.feature  UDM_SUBS_Enhancements_No n_Metrics.feature  udr_no_igw_dr_metrics	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	behave.ini UDR_EGW_LCI_Header.feature UDR_LCI_Header_Via.feature UDR_SCP_Routing.feature UDR_SM_Data_Custom_Data_Enhancements.feature UDR_VSA_Notifications.feature Stage_4 sequential_execution behave.ini ProvGw_SOAP_AndUDR_VSA_New.feature ProvGw_SOAP_Bug_Fixes.feature UDM_SUBS_Enhancements_Metrics.feature UDR_23_2_0_Bug_fixes.feature UDR_All_Migration_State_Entity_rest_metric.feature UDR_AUTO_Enroll_policy_Data.feature UDR_Bulk_Import_All_metrics.feature UDR_Bulk_Import_Configuration_validation.feature UDR_Bulk_Import_ixml_bug_fixes.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_Bulk_Import_VSA.feature	
	UDR_Config_Import_Export.feature	
	UDR_Config_Service.feature	
	UDR_Custom_Fields_Profile.feature	
	UDR_Diameter_CER_Changes.feature	
	UDR_Diameter_Ondemand_Mig.feature	
	UDR_Egw_TLS.feature	
	UDR_IGW_LCI_Header.feature	
	UDR_Log_Level.feature	
	UDR_Mgm_Attribute_Mapping.feature	
	UDR_Migration_tool_part2.feature	
	UDR_Misc_Metrics_Alerts.feature	
	UDR_NEF_AM_Data.feature	
	UDR_NEF_MGM_APIs.feature	
	UDR_NEF_Misc.feature	
	UDR_NEF_Notification.feature	
	UDR_NEF_Prov.feature	
	UDR_NEF_Prov_OAuth2_Validation.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_NEF_SM_Data.feature UDR_NEF_Subscription.fe ature UDR_Network_Policy_Vali dation.feature UDR_New_Log_Level.featur e UDR_PCF_Auto_Enrollment_ New.feature UDR_PCF_ETag_Validation. feature UDR_PCF_MGM_APIs.feature UDR_PCF_New_API_AllData. feature UDR_PCF_New_API_AM.featu re UDR_PCF_New_API_SM.featu re UDR_PCF_New_API_Subscrip tion.feature UDR_PCF_New_API_UeData.f eature UDR_PCF_New_API_UMD.featu re UDR_PCF_New_Server_Heade r.feature UDR_PCF_Ondemand.feature UDR_PCF_Sbi_Correlation_ Header.feature UDR_PCF_User_Agent_Heade r_New.feature UDR_PCF_VSA_Diameter.fea	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	ture UDR_PCF_VSA_ETag_New.fea ture UDR_Profile_Data_New.fea ture UDR_Prov_Policy_OnDemand .feature UDR_Quota_Migration_OnDe mand_Misc.feature UDR_Quota_Migration_OnDe mand_NudrPolicy.feature UDR_Quota_Migration_OnDe mand_Sh.feature UDR_Quota_Migration_OnDe mand_SOAP_ProvPolicy.fea ture UDR_Quota_Migration_OnDe mand_VSA.feature UDR_Quota_Migration_Poli cy_Sh_Soap.feature UDR_Quota_Migration_Sh_i nterface.feature UDR_SMDData_Custom_Fields .feature UDR_Subscriber_Activity_ Logging.feature UDR_TLS_Enable.feature UDR_UDM_API_Additional_E rr.feature UDR_UDM_API_Authenticati on_Data.feature UDR_UDM_API_Context_Data .feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_UDM_API_EE_Profile_Data.feature	
	UDR_UDM_API_Identity_Data.feature	
	UDR_UDM_API_Operator_Determined_Barring_Data.feature	
	UDR_UDM_API_Operator_Specific_Data.feature	
	UDR_UDM_API_Provisioned_Data.feature	
	UDR_UDM_API_Provisioned_parameter_Data.feature	
	UDR_UDM_API_UE_Update_Confirmation_Data.feature	
	UDR_UDM_Mgm_Apis.feature	
	UDR_UDM_Prov_Authentication_Data.feature	
	UDR_UDM_Prov_Context_Data.feature	
	UDR_UDM_Prov_EE_Profile_Data.feature	
	UDR_UDM_Prov_Identity_Data.feature	
	UDR_UDM_Prov_Operator_Determined_Barring_Data.feature	
	UDR_UDM_Prov_Operator_Specific_Data.feature	
	UDR_UDM_Prov_Provisioned_Data.feature	
	UDR_UDM_Prov_Provisioned_Parameter_Data.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_UDM_Prov_UE_Update_Confirmation_Data.feature UDR_UDM_Sbi_Correlation_Header.feature UDR_VSA_Delete_All_Validation.feature UDR_Vsa_Ondemand.feature UDR_XFCC.feature VSA_SMPOLICY.feature BULK-IMPORT Stage_1 bulk_import_metrics_auto_create_true behave.ini UDR_Bulk_Import.feature UDR_Bulk_Import_ixml_part1.feature UDR_Bulk_Import_PolicyAndVsa.feature Stage_2 bulk_import_autocreate_false behave.ini UDR_Bulk_Import_ixml_part2.feature Stage_4 sequential_execution behave.ini UDR_Bulk_Import_All_metrics.feature UDR_Bulk_Import_ixml_bug	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	_fixes.feature UDR_Bulk_Import_VSA.feature CONFIG Stage_4 sequential_execution behave.ini UDR_Bulk_Import_Configuration_validation.feature UDR_Config_Import_Export.feature UDR_Config_Service.feature UDR_New_Log_Level.feature DIAMETER Stage_1 diameter_only_metrics behave.ini Diameter_All_Entities.feature Diameter_Config.feature Diameter_Misc.feature Diameter_User_Data.feature UDR_22_4_1_Diameter.feature UDR_23_1_0_Diameter.feature UDR_23_2_0_Diameter.feature MISCELLANEOUS Stage_1	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	 23_1_1_ProvGw_fixes.feature behave.ini ProvGw_SOAP_Additional_scenarios.feature ProvGw_SOAP_CNCEsUDR82_misc.feature ProvGw_SOAP_ConvertToSec_strict.feature ProvGw_SOAP_Custom_Entities.feature ProvGw_SOAP_Custom_Fields.feature ProvGw_SOAP_Format_Tags.feature ProvGw_SOAP_GetSubscription.feature ProvGw_SOAP_Misc.feature ProvGw_SOAP_Reset_Quota.feature ProvGw_SOAP_SSL.feature ProvGw_SOAP_SubscriberEntities.feature ProvGw_SOAP_Subscriber_Entity_Fields.feature ProvGw_SOAP_Subscriber_Fields.feature ProvGw_SOAP_Subscriber_Profile.feature udr_no_igw_dr_metrics behave.ini	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_EGW_LCI_Header.feature	
	UDR_LCI_Header_Via.feature	
	UDR_SCP_Routing.feature	
	Stage_4	
	sequential_execution	
	behave.ini	
	ProvGw_SOAP_AndUDR_VSA_New.feature	
	ProvGw_SOAP_Bug_Fixes.feature	
	UDR_23_2_0_Bug_fixes.feature	
	UDR_All_Migration_State_Entity_rest_metric.feature	
	UDR_Custom_Fields_Profile.feature	
	UDR_Diameter_CER_Changes.feature	
	UDR_Egw_TLS.feature	
	UDR_IGW_LCI_Header.feature	
	UDR_Log_Level.feature	
	UDR_Migration_tool_part2.feature	
	UDR_Misc_Metrics_Alerts.feature	
	UDR_Network_Policy_Validation.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_PCF_VSA_Diameter.feature	
	UDR_PCF_VSA_ETag_New.feature	
	UDR_Quota_Migration_OnDemand_Misc.feature	
	UDR_Quota_Migration_OnDemand_NudrPolicy.feature	
	UDR_Quota_Migration_OnDemand_Sh.feature	
	UDR_Quota_Migration_OnDemand_SOAP_ProvPolicy.feature	
	UDR_Quota_Migration_OnDemand_VSA.feature	
	UDR_Quota_Migration_Policy_Sh_Soap.feature	
	UDR_Quota_Migration_Sh_interface.feature	
	UDR_Subscriber_Activity_Logging.feature	
	UDR_UDM_Mgm_Apis.feature	
	UDR_VSA_Delete_All_Validation.feature	
	UDR_XFCC.feature	
	NEF	
	Stage_4	
	sequential_execution	
	behave.ini	
	UDR_NEF_AM_Data.feature	
	UDR_NEF_MGM_APIs.feature	
	UDR_NEF_Misc.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_NEF_Notification.feature UDR_NEF_Prov.feature UDR_NEF_Prov_OAuth2_Validation.feature UDR_NEF_SM_Data.feature UDR_NEF_Subscription.feature NRF-CLIENT Stage_1 nrf_only_metrics behave.ini UDR_NRFclient_DNS_Srv.feature UDR_NRFclient_Scenarios.feature ONDEMAND Stage_4 sequential_execution behave.ini UDR_Diameter_Ondemand_Mig.feature UDR_PCF_Ondemand.feature UDR_Prov_Policy_Ondemand.feature UDR_Vsa_Ondemand.feature PCF Stage_1 oauth2_only_metrics behave.ini UDR_PCF_Prov_OAuth2_Vali	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	dation.feature └── pcf_subscription_no_metr ics └── behave.ini └── UDR_PCF_DELETE.feature └── UDR_PCF_SMDData_Patch.fea ture └── UDR_Subscriber_Policy_Da ta.feature └── Stage_2 └── no_igw_dr_metrics └── behave.ini └── UDR_PCF_ETag_Notify_Vali dation.feature └── UDR_PCF_Notification_New _API.feature └── UDR_PCF_User_Agent_Heade r.feature └── Stage_3 └── udr_no_igw_dr_metrics └── behave.ini └── UDR_SM_Data_Custom_Data_ Enhancements.feature └── Stage_4 └── sequential_execution └── behave.ini └── UDR_AUTO_Enroll_policy_D ata.feature └── UDR_Mgm_Attribute_Mappin g.feature └── UDR_PCF_Auto_Enrollment_	

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	New.feature UDR_PCF_ETag_Validation.feature UDR_PCF_MGM_APIs.feature UDR_PCF_New_API_AllData.feature UDR_PCF_New_API_AM.feature UDR_PCF_New_API_SM.feature UDR_PCF_New_API_Subscription.feature UDR_PCF_New_API_UeData.feature UDR_PCF_New_API_UMD.feature UDR_PCF_New_Server_Header.feature UDR_PCF_Sbi_Correlation_Header.feature UDR_PCF_User_Agent_Header_New.feature UDR_Profile_Data_New.feature UDR_SMDData_Custom_Fields.feature UDR_TLS_Enable.feature UDM Stage_1 oauth2_only_metrics behave.ini UDR_UDM_Prov_OAuth2_Vali	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	dation.feature pcf_subscription_no_metr ics behave.ini UDR_UDM_API_SDM_Subscrip tion.feature UDR_UDM_Subscription_Dat a_Subscription.feature Stage_2 no_igw_dr_metrics behave.ini UDR_UDM_Notification.fea ture UDR_UDM_UserAgent_Header .feature Stage_3 provgw_no_igw_dr_metrics behave.ini UDM_SUBS_Enhancements_No n_Metrics.feature Stage_4 sequential_execution behave.ini UDM_SUBS_Enhancements_Me trics.feature UDR_UDM_API_Additional_E rr.feature UDR_UDM_API_Authenticati on_Data.feature UDR_UDM_API_Context_Data .feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_UDM_API_EE_Profile_Data.feature	
	UDR_UDM_API_Identity_Data.feature	
	UDR_UDM_API_Operator_Determined_Barring_Data.feature	
	UDR_UDM_API_Operator_Specific_Data.feature	
	UDR_UDM_API_Provisioned_Data.feature	
	UDR_UDM_API_Provisioned_parameter_Data.feature	
	UDR_UDM_API_UE_Update_Confirmation_Data.feature	
	UDR_UDM_Prov_Authentication_Data.feature	
	UDR_UDM_Prov_Context_Data.feature	
	UDR_UDM_Prov_EE_Profile_Data.feature	
	UDR_UDM_Prov_Identity_Data.feature	
	UDR_UDM_Prov_Operator_Determined_Barring_Data.feature	
	UDR_UDM_Prov_Operator_Specific_Data.feature	
	UDR_UDM_Prov_Provisioned_Data.feature	
	UDR_UDM_Prov_Provisioned_Parameter_Data.feature	
	UDR_UDM_Prov_UE_Update_Confirmation_Data.feature	

Table 4-16 (Cont.) UDR ATS Stages and Groups

UDR New Feature	UDR Regression (Single Site)	UDR Regression (Multiple Site)
	UDR_UDM_Sbi_Correlation_Header.feature VSA Stage_1 vsa_no_metrics behave.ini UDR_22.4.x_PCF_VSA_GET_DELETE.feature UDR_VSA_PATCH_Fix.feature UDR_VSA_PCF_Validation.feature Stage_3 udr_no_igw_dr_metrics behave.ini UDR_VSA_Notifications.feature Stage_4 sequential_execution behave.ini VSA_SMPOLICY.feature	

EIR ATS Stages and Groups

Following table provides different stages and groups for EIR ATS:

Table 4-17 EIR ATS Stages and Groups

EIR NewFeatures	EIR Regression
cust_newfeatures/ └─ Stage_1 └─ eir_basics └─ behave.ini └─ EIR_ErrorResponseDetail.feature	cust_regression/ └─ Stage_1 └─ bulk_import_basics └─ behave.ini └─ EIR_22_4_x_Patch_PDBI_Import.feature └─ EIR_CSV_Import.feature └─ EIR_PDBI_IMEISV_Import.feature └─ EIR_PDBI_Import.feature └─ Prometheus_bulk_import.feature └─ eir_basics └─ behave.ini └─ EIR_Provisioning_Err.feature └─ EIR_TLS_Provisioning.feature └─ EIR_TLS_Signalling.feature └─ EIR_Validation_Additional_Scenarios.f eature └─ export_tool_basics └─ behave.ini └─ EIR_Subscriber_Export_Tool.feature └─ EIR_Subscriber_Export_Tool_New.featur e

Table 4-17 (Cont.) EIR ATS Stages and Groups

EIR NewFeatures	EIR Regression
	— ingressgateway_features — behave.ini — EIR_OAuth2_Provisioning.feature — EIR_OAuth2_Signalling.feature — EIR_Server_Header.feature — EIR_XFCC.feature — nrf-client — behave.ini — EIR_Dns_Srv.feature — EIR_Nrf_Client_Scenarios.feature — EIR_User_AgentHeader_Nrf.feature — Prometheus_nrfclient.feature — Stage_2 — sequential_execution — behave.ini — EIR_22_4_x_Patch_PDBI_Import_part2.fe ature — EIR_AccessLog.feature — EIR_Config_Import_Export.feature — EIR_CSV_Import_part2.feature

Table 4-17 (Cont.) EIR ATS Stages and Groups

EIR NewFeatures	EIR Regression
	EIR_DefaultIMEINotFound_Response.feature
	EIR_Default_Response.feature
	EIR_Export_Configuration_validation.feature
	EIR_IMEISV_AccessLog.feature
	EIR_IMEISV_Default_Response.feature
	EIR_IMEISV.feature
	EIR_LogLevels.feature
	EIR_MGM_API.feature
	EIR_PDBI_IMEISV_Import_part2.feature
	EIR_PDBI_Import_part2.feature
	EIR_PDBI_Synchronous_Scenarios.feature
	EIR_Provisioning_Err_part2.feature
	EIR_Signalling.feature
	EIR_Subscriber_Prov.feature
	EIR_User_Agent_Header.feature
	EIR_Validation_Additional_Scenarios_part2.feature

Table 4-17 (Cont.) EIR ATS Stages and Groups

EIR NewFeatures	EIR Regression
	Prometheus_Scenarios.feature

Viewing Console Logs for Specific Group

When parallel test execution feature is enabled and ATS runs are triggered, you can view the progress of each stage and group on the corresponding pipeline dashboard.

Figure 4-85 Sample SLF Regression

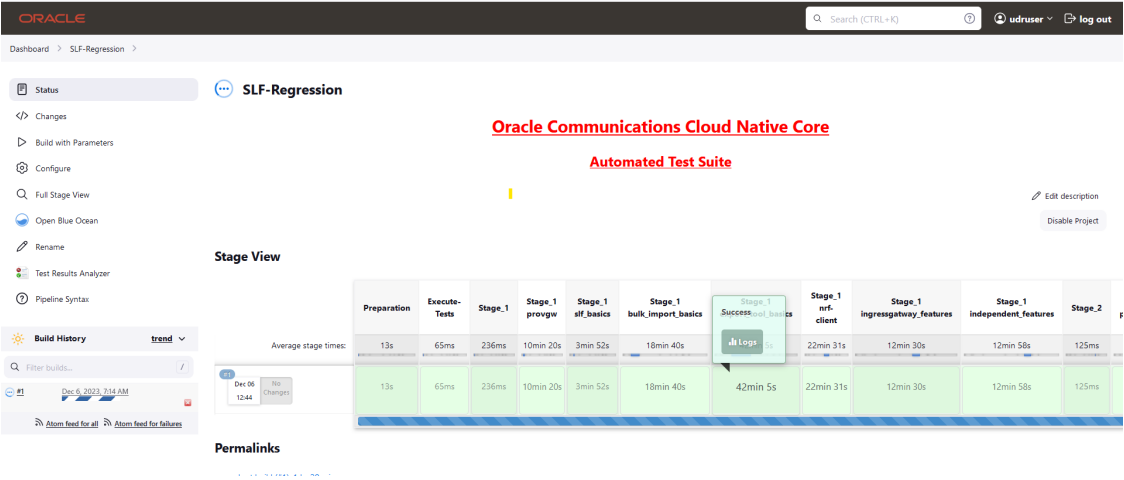
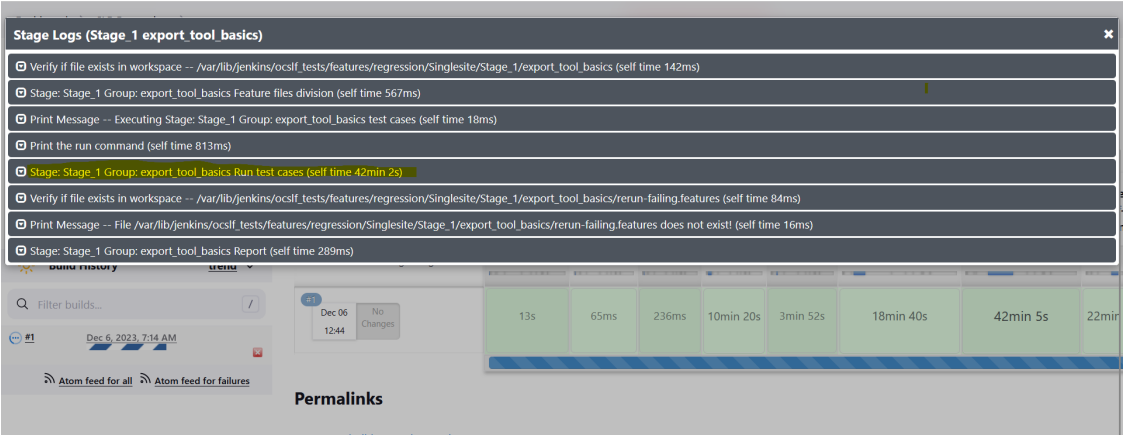


Figure 4-86 Sample Stage Logs



4.8.5 Logging into ATS

Before logging into ATS, it is important to know the nodeport of the "-ocats-UDR" service. To get the nodeport detail, run the following command:

```
kubectl get svc -n <slf_namespace>
```

Example: `kubectl get svc -n ocudr ocats-udr-ga`

Figure 4-87 UDR Nodeport

ats-ocats-udr	LoadBalancer	10.233.22.95	<pending>	8080:30156/TCP,5001:31175/TCP	57s

In the previous image, 31788 is the nodeport.

Run the following command to get the worker node IP address:

```
kubectl get nodes -o wide
```

To log in to ATS:

1. In the web browser, type **http://<Worker IP>:nodeport** and press **Enter**.

Example: `http://10.75.229.87:30156`

The log in page appears.

2. Enter the username and password. Click **Sign in**. A page with pre-configured pipelines appears.

Figure 4-88 SLF, EIR, and UDR Pre-configured Pipelines

Oracle Communications Cloud Native Core

Automated Test Suite

CNCATS EIR UDR SLF : 24.1.1

S	W	Name	Last Success	Last Failure	Last Duration	F
⊕	☀	SLF-HealthCheck	N/A	N/A	N/A	▶ ☆
⊕	☀	SLF-NewFeatures	N/A	N/A	N/A	▶ ☆
⊕	☀	SLF-Performance	N/A	N/A	N/A	▶ ☆
⊕	☀	SLF-Regression	N/A	N/A	N/A	▶ ☆
⊕	☀	UDR-HealthCheck	N/A	N/A	N/A	▶ ☆
⊕	☀	UDR-NewFeatures	N/A	N/A	N/A	▶ ☆
⊕	☀	UDR-Regression	N/A	N/A	N/A	▶ ☆

Figure 4-89 EIR ATS Pipeline

The screenshot shows the Oracle Cloud Native Core Automated Test Suite (ATS) interface. The top navigation bar includes the Oracle logo, a search bar, and user information (eiruser). The main content area displays the 'CNCATS EIR UDR SLF : 24.1.1' pipeline configuration. A table lists the pipeline components: EIR-HealthCheck, EIR-NewFeatures, and EIR-Regression. The table columns include S (Schedule), W (Week), Name, Last Success, Last Failure, Last Duration, and F (Frequency). The 'Build Queue' section shows 'No builds in the queue.' and the 'Build Executor Status' section shows '1 idle'.

S	W	Name	Last Success	Last Failure	Last Duration	F
☐	☐	EIR-HealthCheck	N/A	N/A	N/A	▶ ☆
☐	☐	EIR-NewFeatures	N/A	N/A	N/A	▶ ☆
☐	☐	EIR-Regression	N/A	N/A	N/A	▶ ☆

Note

To modify the default log in password, see [Modifying Login Password](#).

- **SLF-HealthCheck:** This pipeline does a test probe with SUT - trigger Helm test and provide the results in Console Logs.
- **SLF-NewFeatures:** This pipeline has all the test cases delivered as part of SLF ATS - 24.1.1 release.
- **SLF-Performance:** This pipeline provides a means to run 1K TPS of lightweight traffic.
- **SLF-Regression:** This pipeline has all the test cases of previous releases.
- **UDR-HealthCheck:** This pipeline does a test probe with SUT - trigger helm test and provide the results in Console Logs.
- **UDR-NewFeatures:** This pipeline has all the test cases delivered as part of UDR ATS - 24.1.1 release.
- **UDR-Regression:** This pipeline has all the UDR test cases of previous releases.
- **EIR-HealthCheck:** This pipeline does a test probe with SUT - trigger Helm test and provide the results in Console Logs.
- **EIR-NewFeatures:** This pipeline has all the test cases delivered as part of EIR ATS - 24.1.1 release.
- **EIR-Regression:** This pipeline has all the EIR test cases of previous releases.

4.8.6 Running SLF Pipelines

This section describes how to run SLF related pipelines. It includes:

- [Running SLF-NewFeatures Pipeline](#)
- [Running SLF Regression Pipeline - Single Site Deployment](#)
- [Running SLF Regression Pipeline - Multiple Site Deployment](#)
- [Running SLF HealthCheck Pipeline](#)
- [Running SLF-Performance Pipeline: Model 2: Single Site Deployment](#)

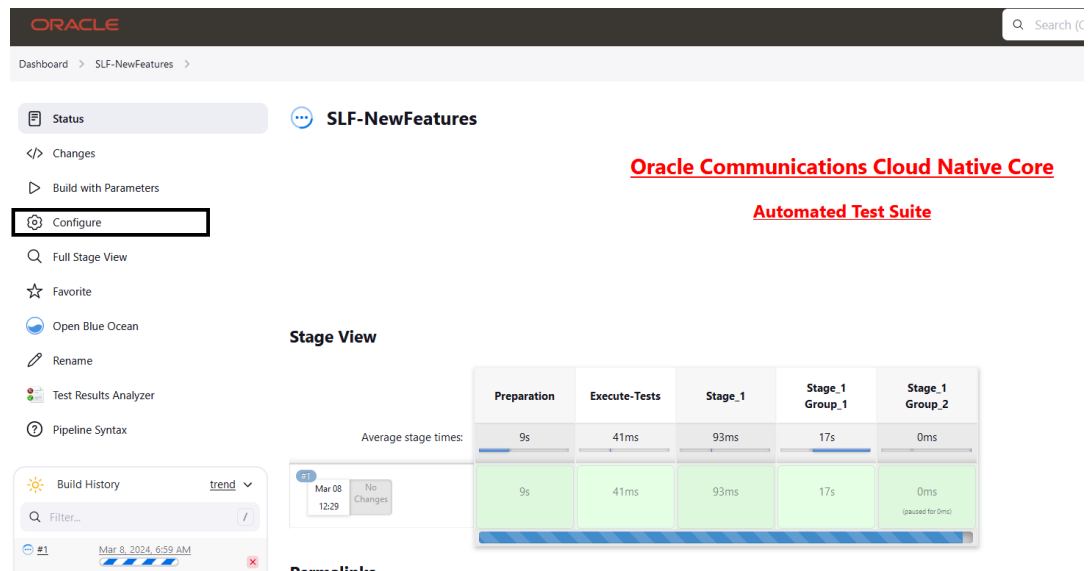
- [SLF NewFeatures Documentation - Single Site Scenarios](#)
- [SLF Regression Documentation - Single Site Scenarios](#)
- [SLF Regression Documentation - Multiple Site Scenarios](#)

4.8.6.1 Running SLF-NewFeatures Pipeline

To run the test cases:

1. Click **SLF-NewFeatures**. The following page appears:

Figure 4-90 SLF-NewFeatures Configure



2. Click **Configure** in the left navigation pane. The **General** tab appears. You must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there is large number of builds, the [Persistent Volume](#) may be utilized completely.

Figure 4-91 Discard Old Builds

The screenshot shows the 'Configure' page for 'SLF-NewFeatures' under the 'Configuration' tab. On the left, there are three tabs: 'General' (selected), 'Advanced Project Options', and 'Pipeline'. The main content area is titled 'Discard old builds' with a checked checkbox and a help icon. Below this, there is a 'Strategy' dropdown menu set to 'Log Rotation'. There are two input fields: 'Days to keep builds' (with a note: 'If not empty, build records are only kept up to this number of days') and 'Max # of builds to keep' (with a note: 'If not empty, only up to this number of build records are kept'). At the bottom, there is an 'Advanced' dropdown menu.

4. Click the **Advanced Project Options** tab. Scroll down to reach the **Pipeline** configuration section as follows:

! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Figure 4-92 SLF Configuration page

The screenshot shows the 'Configure' page for 'SLF' under the 'Configuration' tab. On the left, there are three tabs: 'General', 'Advanced Project Options', and 'Pipeline' (selected). The main content area is titled 'Pipeline' with a 'Definition' dropdown menu set to 'Pipeline script'. Below this, there is a 'Script' section with a text area containing a shell script. The script starts with a comment line 6, followed by variable assignments for 'PROVGM_Helm_Name', 'STUB_FQDN_PORT', 'STUB_METRICS_PORT', 'NRF_STUB_DEPLOY_NAME', 'PROVGM_HTTPS_Port', 'CAFILE', 'CERTFILE', 'KEYFILE', 'POD_DOWN_RETRIES', 'POD_UP_RETRIES', 'HTTP_CONNECT_RETRY', 'HTTP_RETRY_INTERVAL', 'MULTISITE_UDR2_FQDN', 'MULTISITE_UDR2_CONFIG_FQDN', 'CNE_VERSION_19', 'UDR1_PROV_FQDN', 'UDR2_PROV_FQDN', 'UDR3_PROV_FQDN', 'UDR4_PROV_FQDN', and 'MULTISITE_UDR2_PROV_FQDN'. The script then defines variables for 'SELECTED_NF', 'SLF_NAMESPACE', 'PROVGM_NAMESPACE', 'UDR1_FQDN_PORT', 'UDR2_FQDN_PORT', 'UDR3_FQDN_PORT', 'UDR4_FQDN_PORT', and 'PROVGM_FQDN_PORT'. At the bottom, there are 'Save' and 'Apply' buttons.

In the pipeline script, do not change any value other than from line number **66 to line 108**. You can change the parameters marked as **"b" - to - "R"** as per user requirement. The parameters details are provided as comments in line number **16 to 58**. The parameters description is as follows:

- a - Selected NF
- b - NameSpace in which UDR is Deployed (Mandatory)
- c - NameSpace in which ProvGw is Deployed (Mandatory)
- d - Preferred UDR1 FQDN and PORT of segment 1 in the format FQDN:PORT
- e - UDR2 FQDN and PORT of segment 1 in the format FQDN:PORT
- f - Preferred UDR3 FQDN and PORT of segment 2 in the format FQDN:PORT

- g - UDR4 FQDN and PORT of segment 2 in the format FQDN:PORT
- h - ProvGw FQDN and PORT in the format FQDN:PORT
- i - prometheus_ip and port of prometheus server in the format FQDN:PORT
- j - re-run count
- k - Name and Port of Kubernetes Host server in the format FQDN:PORT
- l - Mode of Communication (Can be either IP or fqdn)
- m - Helm Name for UDR1
- n - Helm Name for UDR2
- o - Helm Name for UDR3
- p - Helm Name for UDR4
- q - Helm Name for ProvGw
- r - FQDN:PORT format for primary nrf-stub
- s - primary nrf-stub metrics port
- t - nudr-config FQDN:PORT format
- u - prometheus uri apiName and apiVersion
- v - FQDN:PORT format for secondary nrf-stub
- w - secondary nrf-stub metrics port
- x - primary nrf-stub deployment name
- y - secondary nrf-stub deployment name
- z - ProvGw HTTPS port
- A - root certificate authority
- B - Server certificate signed with root CA private key
- C - private key in .pem format
- D - Number of retries to check if pod down is successful
- E - Number of retries to check if pod up is successful
- F - Number of retries when ATS detects tcp connection failures
- G - Time interval in seconds between tcp connection retries
- H - UDR2 FQDN:PORT for multi-site scenarios
- I - UDR2 nudr-config FQDN:PORT for multi-site scenarios
- J - ProvGateway Config service host and port in the format FQDN:PORT
- K - Bulk import helm name
- L - true if CNE VERSION is 1.9 or above or OSO version is 1.10.x or above, else false.
- M - Preferred UDR1 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- N - UDR2 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- O - Preferred UDR3 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- P - UDR4 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- Q - MultiSite UDR2 PROV FQDN and PORT in the format FQDN:PORT
- R - Helm Release name of SLF export tool

Note

Setting rerun count, number of retries to check pod status and segment details:

- Rerun count can be set to any desired value by changing option 'j' in the pipeline script.
- Number of retries to check if pod is down can be set by providing option 'D' in the pipeline script.
- Number of retries to check if pod is up can be set by providing option 'E' in the pipeline script.
- If ProvGw is deployed with segment details having UDR fqdn/IP:port, provide option 'I' as "IP". If ProvGw is deployed with segment details having only UDR fqdn/IP, provide option 'I' as "fqdn". If any value other than "IP" and "fqdn" is provided, ATS assumes that the segment details in ProvGw is in "IP" mode.
- Option 'u' shall take value '/<domain name>/prometheus/api/v1' for CNE version 1.9 or above or OSO version is 1.10.x or above. If domain name is not known, then domain name can be found after executing the command from ATS pod:

```
curl http://<prometheus fqdn>:<port>/
```

Example:

```
curl http://ocne-kube-prom-stack-kube-prometheus.ocne-infra/
Found
```

5. Click **Save**. The SLF-NewFeatures page appears.
6. Click **Build with Parameters**. The following page displays all the newly added features and total count of the features, counts of scenarios in each feature, name of each feature and scenarios in each feature:

Figure 4-93 SLF Build with Parameters

The screenshot displays the Oracle Communications Cloud Native Core Automated Test Suite - UDR interface. The page is titled "Pipeline SLF-NewFeatures" and includes a sidebar with navigation options: Status, Changes, Build with Parameters, Configure, Full Stage View, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area shows the "EXECUTION OPTIONS" section with the following details:

TestSuite	SUT	SUTSuite	Configuration_Type
NewFeatures	UDR-SLF	All	Product_Config

Below this, the "Log_Level" is set to "Not Applicable". The "FEATURES AND TESTCASES" section lists the following features:

- Stage: Stage_1 Group: Group_1
 - SLF_Error_Response_Enhancement
- Stage: Stage_1 Group: Group_2
 - ProvGw_Auditor_Enhancements
- ProvGw_Auditor_Prometheus_Scenarios

In the previous image,

- Select the **Configuration_Type**. For information about this feature, see [Parameterization](#).
- There are two **Select_Option(s)**, which are:
 - **All**: By default, it selects all the SLF test cases to run. Scroll-down and click **Build** to run all the test cases.
 - **Single/MultipleFeatures**: This option allows you to select the test cases you want to run. After selecting the test cases, scroll-down and click **Build**.

If the **Select_Option** is selected as 'All', the **TestCases** details are mapped to each feature and if the **Select_Option** is selected as 'Single/MultipleFeatures', only the TestCases details corresponding to those test cases appear. For more information, see [Support for Test Case Mapping and Count](#).

To view consolidated and detailed stack trace results in case of any failures, click **Test Results Analyzer** in the left navigation pane. The test results analyzer report appears. For more information, see [Test Results Analyzer](#).

A sample consolidated test report, when rerun is set to 0, is as follows:

Figure 4-94 Consolidated Test Report - Rerun is 0

```
*****Stage: Stage_1 Group: Group_1 Result*****
Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_1 Group: Group_2 Result)
+ set +x
*****Stage: Stage_1 Group: Group_2 Result*****
Final Result:-
Total-Features RUN 2, PASS 2, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_2 Group: Group_1 Result)
+ set +x
*****Stage: Stage_2 Group: Group_1 Result*****
Final Result:-
Total-Features RUN 0, PASS 0, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] sh (Overall Result)
+ set +x
*****Overall Result*****

Took 1h21m21s

Final Result:-
Total-Features RUN 3, PASS 3, FAIL 0
```

For more details on consolidated test report, see [Final Summary Report, Build Color and Application Logs](#).

4.8.6.2 Running SLF Regression Pipeline - Single Site Deployment

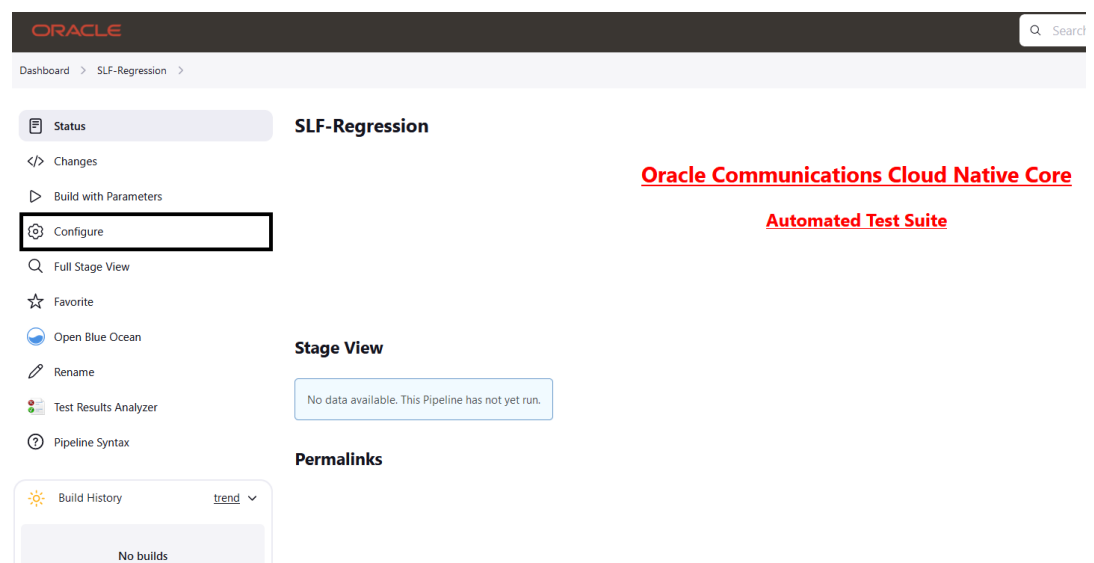
This pre-configured pipeline consists of all the test cases delivered till SLF ATS 24.1.1. However, some test cases are updated as per new implementation of SLF. To run the single site scenarios:

Note

Model 1 deployment is required for this pipeline. For more information, see [Prerequisites for SLF Pipelines](#).

1. Click **SLF-Regression** and then, click **Configure**.

Figure 4-95 SLF-Regression Pipeline



2. The **General** tab appears. The user must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there is a large number of builds, the [Persistent Volume](#) may be utilized completely.

4. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than from line number **66 to 108**. You can change the parameters marked as "**b**" - **to** - "**R**" as per user requirement. The parameters details are provided as comments in line number **16 to 58**. The parameters description is as follows:

- a - Selected NF
- b - NameSpace in which UDR is Deployed (Mandatory)
- c - NameSpace in which ProvGw is Deployed (Mandatory)
- d - Preferred UDR1 FQDN and PORT of segment 1 in the format FQDN:PORT
- e - UDR2 FQDN and PORT of segment 1 in the format FQDN:PORT
- f - Preferred UDR3 FQDN and PORT of segment 2 in the format FQDN:PORT
- g - UDR4 FQDN and PORT of segment 2 in the format FQDN:PORT
- h - ProvGw FQDN and PORT in the format FQDN:PORT
- i - prometheus_ip and port of prometheus server in the format FQDN:PORT
- j - re-run count
- k - Name and Port of Kubernetes Host server in the format FQDN:PORT
- l - Mode of Communication (Can be either IP or fqdn)
- m - Helm Name for UDR1
- n - Helm Name for UDR2
- o - Helm Name for UDR3
- p - Helm Name for UDR4
- q - Helm Name for ProvGw
- r - FQDN:PORT format for primary nrf-stub
- s - primary nrf-stub metrics port
- t - nudr-config FQDN:PORT format
- u - prometheus uri apiName and apiVersion
- v - FQDN:PORT format for secondary nrf-stub
- w - secondary nrf-stub metrics port
- x - primary nrf-stub deployment name
- y - secondary nrf-stub deployment name
- z - ProvGw HTTPS port
- A - root certificate authority
- B - Server certificate signed with root CA private key
- C - private key in .pem format
- D - Number of retries to check if pod down is successful
- E - Number of retries to check if pod up is successful
- F - Number of retries when ATS detects tcp connection failures
- G - Time interval in seconds between tcp connection retries
- H - UDR2 FQDN:PORT for multi-site scenarios
- I - UDR2 nudr-config FQDN:PORT for multi-site scenarios

- J - ProvGateway Config service host and port in the format FQDN:PORT
- K - Bulk import helm name
- L - true if CNE VERSION is 1.9 or above or OSO version is 1.10.x or above, else false.
- M - Preferred UDR1 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- N - UDR2 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- O - Preferred UDR3 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- P - UDR4 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- Q - MultiSite UDR2 PROV FQDN and PORT in the format FQDN:PORT
- R - Helm Release name of SLF export tool

Note

Setting rerun count, number of retries to check pod status and segment details:

- Rerun count can be set to any desired value by changing option 'j' in the pipeline script.
- Number of retries to check if pod is down can be set by providing option 'D' in the pipeline script.
- Number of retries to check if pod is up can be set by providing option 'E' in the pipeline script.
- If ProvGw is deployed with segment details having UDR fqdn/IP:port, provide option 'l' as "IP". If ProvGw is deployed with segment details having only UDR fqdn/IP, provide option 'l' as "fqdn". If any value other than "IP" and "fqdn" is provided, ATS assumes that the segment details in ProvGw is in "IP" mode.
- Option 'u' shall take value '/<domain name>/prometheus/api/v1' for CNE version 1.9 or above or OSO version is 1.10.x or above. If domain name is not known, then domain name can be found after executing the command from ATS pod:

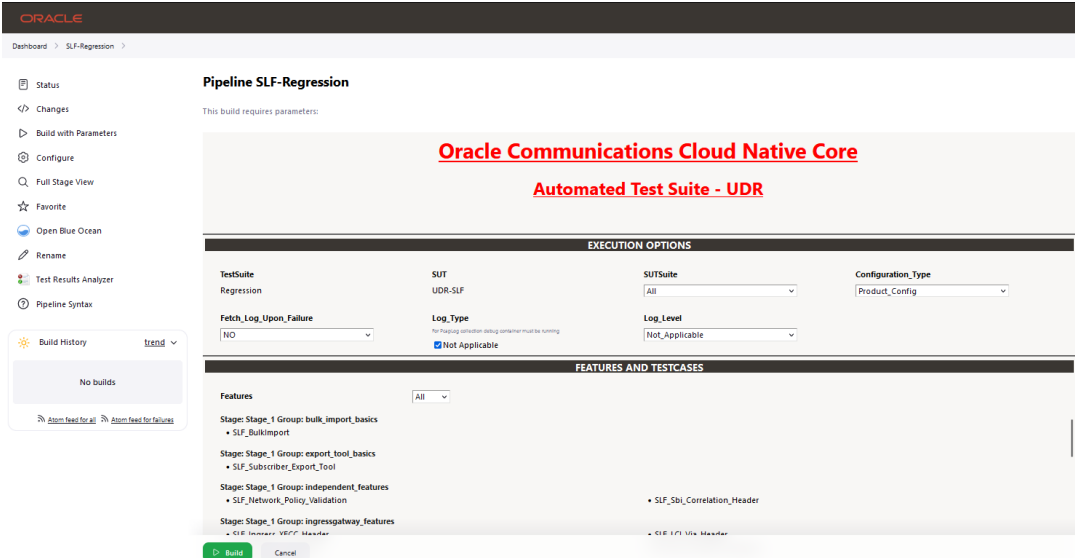
```
curl http://<prometheus fqdn>:<port>/
```

Example:

```
curl http://occne-kube-prom-stack-kube-prometheus.occne-infra/
<a href="/udr-cne-190/prometheus">Found</a>
```

5. Click **Save**. The **Pipeline SLF-Regression** page appears.
6. Click **Build with Parameters**.

Figure 4-96 Build with Parameters



- 7. Select the Configuration_Type. For more information about this feature, see [Parameterization](#).
- 8. Select **SUTSuite** as 'All'. This lists all regression features and displays the total count of the features, counts of scenarios in each feature, name of each feature, and scenarios in each feature.
- 9. Click **Build**.
For more details on consolidated test report, see [Final Summary Report, Build Color and Application Logs](#).

A sample consolidated test report, when rerun is set to 0, is as follows:

Figure 4-97 Consolidated Test Report Rerun

```

Dashboard > SLF-Regression > #2

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_2 Group: provgw_config Result)
+ set +x
*****Stage: Stage_2 Group: provgw_config Result*****
Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_2 Group: slf_config Result)
+ set +x
*****Stage: Stage_2 Group: slf_config Result*****
Final Result:-
Total-Features RUN 3, PASS 3, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_3 Group: sequential_execution Result)
+ set +x
*****Stage: Stage_3 Group: sequential_execution Result*****
Final Result:-
Total-Features RUN 16, PASS 16, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] sh (Overall Result)
+ set +x
*****Overall Result*****

Took 3h33m13s

Final Result:-
Total-Features RUN 41, PASS 41, FAIL 0

```

4.8.6.3 Running SLF Regression Pipeline - Multiple Site Deployment

This pre-configured pipeline consists of all the test cases delivered till SLF ATS 24.1.1. However, some test cases are updated as per new implementation of SLF. To run the multiple site scenarios:

Note

Model 3 deployment is required for this pipeline. For more information, see [Prerequisites for SLF Pipelines](#).

1. Click **SLF-Regression** and then, click **Configure**.
2. The **General** tab appears. The user must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there are large number of builds, the [Persistent Volume](#) may be utilized completely.

4. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than from **line number 16 to line 58**. You can change the parameters marked as "**b**" - **to** - "**R**" as per user requirement. The parameters details are provided as comments in line number 14 to 56. The parameters description is as follows:

- a - Selected NF
- b - NameSpace in which UDR is Deployed (Mandatory)
- c - NameSpace in which ProvGw is Deployed (Mandatory)
- d - Preferred UDR1 FQDN and PORT of segment 1 in the format FQDN:PORT
- e - UDR2 FQDN and PORT of segment 1 in the format FQDN:PORT
- f - Preferred UDR3 FQDN and PORT of segment 2 in the format FQDN:PORT
- g - UDR4 FQDN and PORT of segment 2 in the format FQDN:PORT
- h - ProvGw FQDN and PORT in the format FQDN:PORT
- i - prometheus_ip and port of prometheus server in the format FQDN:PORT
- j - re-run count
- k - Name and Port of Kubernetes Host server in the format FQDN:PORT
- l - Mode of Communication (Can be either IP or fqdn)
- m - Helm Name for UDR1
- n - Helm Name for UDR2
- o - Helm Name for UDR3
- p - Helm Name for UDR4
- q - Helm Name for ProvGw
- r - FQDN:PORT format for primary nrf-stub
- s - primary nrf-stub metrics port
- t - nudr-config FQDN:PORT format
- u - prometheus uri apiName and apiVersion
- v - FQDN:PORT format for secondary nrf-stub
- w - secondary nrf-stub metrics port
- x - primary nrf-stub deployment name
- y - secondary nrf-stub deployment name
- z - ProvGw HTTPS port
- A - root certificate authority
- B - Server certificate signed with root CA private key
- C - private key in .pem format
- D - Number of retries to check if pod down is successful
- E - Number of retries to check if pod up is successful

- F - Number of retries when ATS detects tcp connection failures
- G - Time interval in seconds between tcp connection retries
- H - UDR2 FQDN:PORT for multi-site scenarios
- I - UDR2 nudr-config FQDN:PORT for multi-site scenarios
- J - ProvGateway Config service host and port in the format FQDN:PORT
- K - Bulk import helm name
- L - true if CNE VERSION is 1.9 or above or OSO version is 1.10.x or above, else false.
- M - Preferred UDR1 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- N - UDR2 PROV FQDN and PORT of segment 1 in the format FQDN:PORT
- O - Preferred UDR3 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- P - UDR4 PROV FQDN and PORT of segment 2 in the format FQDN:PORT
- Q - MultiSite UDR2 PROV FQDN and PORT in the format FQDN:PORT
- R - Helm Release name of SLF export tool

Note

Setting rerun count, number of retries to check pod status, and segment details:

- Rerun count can be set to any desired value by changing option 'j' in the pipeline script.
- Number of retries to check if pod is down can be set by providing option 'D' in the pipeline script.
- Number of retries to check if pod is up can be set by providing option 'E' in the pipeline script.
- If ProvGw is deployed with segment details having UDR fqdn/IP:port, provide option 'I' as "IP". If ProvGw is deployed with segment details having only UDR fqdn/IP, provide option 'I' as "fqdn". If any value other than "IP" and "fqdn" is provided, ATS assumes that the segment details in ProvGw is in "IP" mode.
- Option 'u' shall take value '/<domain name>/prometheus/api/v1' for CNE version 1.9 or above or OSO version is 1.10.x or above. If domain name is not known, then domain name can be found after executing the command from ATS pod:

```
curl http://<prometheus fqdn>:<port>/
```

```
curl http://occne-kube-prom-stack-kube-prometheus.occne-infra/
<a href="/udr-cne-190/prometheus">Found</a>
```

5. Click **Save**. The **Pipeline SLF-Regression** page appears.
6. Click **Build with Parameters**. Select the **Configuration_Type**.

Note

For information about the Configuration_Type, see [Parameterization](#).

7. Select **SUTSuite** as 'Multi_Site'. This lists all the regression features and displays the total count of the features, counts of scenarios in each feature, name of each feature, and scenarios in each feature.
8. Click **Build**.
A sample consolidated test report, when rerun is set to 0 is as follows:

Figure 4-98 SLF- Regression to run Multiple Site Scenarios - Consolidated Test Report - Rerun is 0

```

3 features passed, 0 failed, 0 skipped
21 scenarios passed, 0 failed, 0 skipped
794 steps passed, 0 failed, 0 skipped, 0 undefined
Took 17m58.722s
[Pipeline] sh
+ cd /var/lib/jenkins/ocslf_tests
++ cat /var/lib/jenkins/ocslf_tests/enviro.sh
++ grep RERUN
++ cut -d= -f2
++ cut '-d;' -f1
+ rerun=0
+ [[ 0 -ne 0 ]]
+ sh re-run.sh 0
1
Scenario passed

```

For more details on consolidated test report, see [Final Summary Report, Build Color and Application Logs](#).

4.8.6.4 Running SLF HealthCheck Pipeline

To run health check on SLF or Provisioning Gateway:

1. Navigate to **SLF-HealthCheck** pipeline and click **Configure**.

Figure 4-99 UDR - SLF-HealthCheck Pipeline

The screenshot shows the Oracle UDR - SLF-HealthCheck Pipeline configuration page. The sidebar on the left contains navigation options: Status, Changes, Build Now, Configure (highlighted with a red box), Full Stage View, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area displays the 'SLF-HealthCheck' pipeline status as 'Success' with a green checkmark. Below this, there's a 'Stage View' section showing a table of stage times. The table has columns for 'Helm-Test-Init' and 'Validate-Helm-Test', with values '2min 51s' and '215ms' respectively. A 'Build History' section at the bottom shows a build from Mar 09 12:00 with 'No Changes'.

2. The **General** tab appears. The user must wait for the page to load completely.
3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

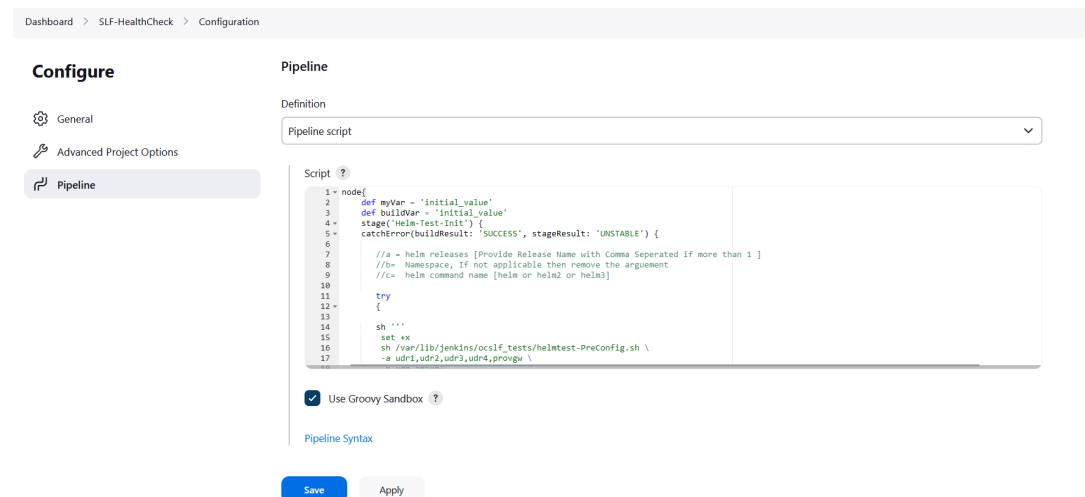
! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than **line numbers 16 and 17**. You can change the parameters marked as "**a**" "**b**" and "**c**" as per user requirement. The parameters details are provided as comments in **line number 7 and 9**. The parameters description is as follows:

- a - helm releases [Provide Release Name with Comma Separated if more than 1]
- b - namespace of SUT.
- c - helm command alias that is being used on the CNE (helm or helm2 or helm3)

Figure 4-100 SLF-HealthCheck Pipeline configuration



4. Click **Save**. The **Pipeline SLF-HealthCheck** page appears.
5. Click **Build Now**. This triggers health check for SUT.

Figure 4-101 SLF-HealthCheck Pipeline Build

The screenshot shows the Oracle Communications Cloud Native Core Automated Test Suite (ATS) interface. The top navigation bar includes the Oracle logo and a search bar. The main header displays 'SLF-HealthCheck' with a green checkmark icon. A sidebar on the left lists various actions: Status (selected), Changes, Build Now (highlighted with a red box), Configure, Full Stage View, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area shows the 'Stage View' for the 'SLF-HealthCheck' pipeline. It includes a table with stage names and their durations:

Stage Name	Duration
Helm-Test-Init	2min 51s
Validate-Helm-Test	215ms

Below the table, there is a 'Build History' section with a 'trend' dropdown and a 'Filter...' input. The history shows a build on Mar 09 at 12:00 with 'No Changes'.

Figure 4-102 Sample Output - SLF Health Check

The screenshot shows the sample output of the SLF Health Check pipeline. The output is displayed in a terminal window with a light gray background. The output text is as follows:

```

Your release is named udr4 , Release Revision: 1.
To learn more about the release, try:

$ helm status udr4
$ helm get udr4
[0m
2024-03-09 06:32:42,735[32m INFO LOG.HELM-TEST:176 [0m] [32mhelm test provgw -n udr-arjun[0m
2024-03-09 06:33:10,827[32m INFO LOG.HELM-TEST:191 [0m] [32m
NAME: provgw
LAST DEPLOYED: Fri Mar 8 12:34:26 2024
NAMESPACE: udr-arjun
STATUS: deployed
REVISION: 1
TEST SUITE: provgw-test
Last Started: Sat Mar 9 06:32:43 2024
Last Completed: Sat Mar 9 06:33:10 2024
Phase: Succeeded
NOTES:
# Copyright 2019 (C), Oracle and/or its affiliates. All rights reserved.

Thank you for installing provgw.

Your release is named provgw , Release Revision: 1.
To learn more about the release, try:

$ helm status provgw
$ helm get provgw
[0m

[Pipeline] }
[Pipeline] // catchError
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

4.8.6.5 Running SLF-Performance Pipeline: Model 2: Single Site Deployment

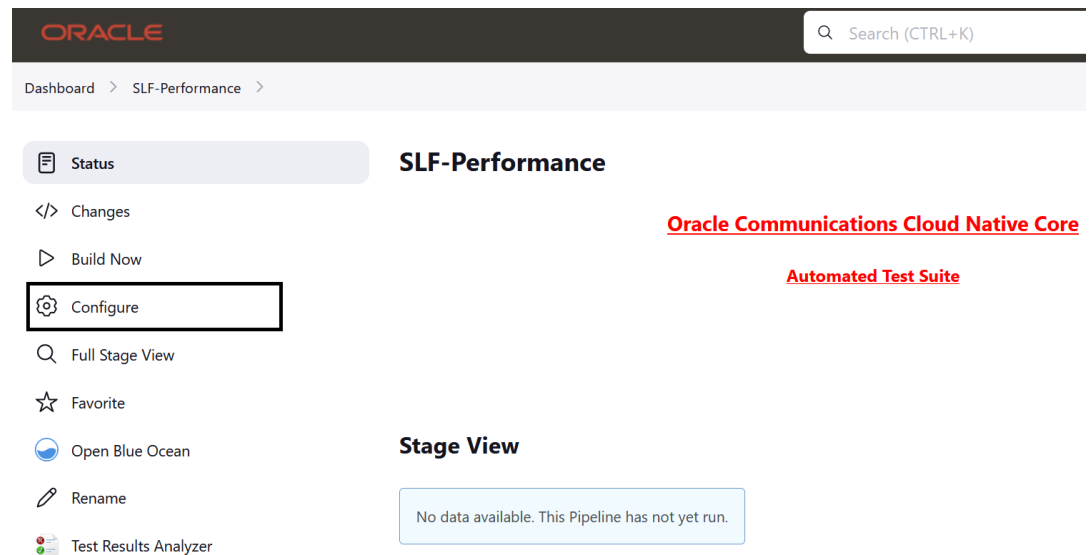
To run light weight performance scenarios:

Note

Model 2 deployment is required for this pipeline. For more information, see [Prerequisites for SLF Pipelines](#).

1. Navigate to the **SLF-Performance** pipeline and click **Configure**.

Figure 4-103 SLF Single Site Performance Model 2



2. The **General** tab appears. The user must wait for the page to load completely.
3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than line number 22 and line number 31. You can change the parameters marked as "b" and "k" as per user requirement. The parameters details are provided as comments in line number 7 and 21. The parameters description is as follows:

- a - Selected NF (should always be SLF)
- b - UDR ingressgateway provisioning FQDN and port for performance testing (ocudr-ingressgateway-prov.ocudr:80)
- c - UDR ingressgateway signalling FQDN and port for performance testing (ocudr-ingressgateway-sig.ocudr:80)
- d - Prometheus Server FQDN and port (occne-prometheus-server.occne-infra:80)
- e - Prometheus URI (/prometheus/api/v1)
- f - Namespace in which UDR is deployed for performance testing (ocudr)

- g - Helm Name of UDR (ocudr)
- h - Kubernetes host FQDN (kubernetes.default)
- i - Kubernetes host port number (443)
- j - Duration for which performance needs to be tested (in seconds) (900)
- k - Whether scenario with ramping up of TPS from 1K to 2K and back, needs to be run (false)

Figure 4-104 SLF-Performance Pipeline Configuration

Dashboard > SLF-Performance > Configuration

Configure

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

Script ?

```
1 = node {'built-in':{
2   //a = SELECTED_NF      b = PROV_INGRESS_EXT_IP   c = SIG_INGRESS_EXT_IP   d = PROMSVCIIP_PORT
3   //e = PROMSVCIURI      f = SLF_Namespace         g = SLF_Helm_Name       h = kube_host
4   //i = kube_port        j = Duration              k = IsRampingRequired
5
6   //Description of Variables:
7   //SELECTED_NF = SLF
8   //PROV_INGRESS_EXT_IP = UDR Provisioning ingressgateway FQDN and port for performance testing (ocudr-ingressgateway-prov.ocudr:80)
9   //SIG_INGRESS_EXT_IP = UDR Signalling ingressgateway FQDN and port for performance testing (ocudr-ingressgateway-sig.ocudr:80)
10  //PROMSVCIIP_PORT = Prometheus Server FQDN and PORT (ocne-prometheus-server-ocne-infra:80)
11  //PROMSVCIURI = Prometheus URI (/prometheus/api/v1)
12  //SLF_Namespace = Namespace in which UDR is deployed for performance testing (ocudr)
13  //SLF_Helm_Name = Helm Name of UDR (ocudr)
14  //kube_host = Kubernetes host FQDN (kubernetes.default)
15  //kube_port = Kubernetes host port number (443)
16  //Duration: Duration for which performance needs to be tested (in seconds) (900)
17  //IsRampingRequired: Whether scenario with ramping up of TPS from 1K to 2K and back, needs to be run (false)
```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save

Apply

- Click **Save**. The **Pipeline SLF-Performance** page appears.
- Click **Build Now**. This triggers lightweight traffic on UDR-SLF.

Figure 4-105 Sample Output - SLF Performance

```
1 feature passed, 0 failed, 0 skipped
1 scenario passed, 0 failed, 1 skipped
30 steps passed, 0 failed, 29 skipped, 0 undefined
Took 21m47.747s
```

4.8.6.6 SLF NewFeatures Documentation - Single Site Scenarios

To view the documentation of SLF-NewFeatures, click the **Documentation** link in the left navigation pane (present inside the build), as follows:

Figure 4-106 SLF-New Features Documentation

ORACLE

Search

Dashboard > SLF-NewFeatures >

Status

SLF-NewFeatures

</> Changes

> Build with Parameters

Configure

Full Stage View

Documentation

HTML Report

Favorite

Open Blue Ocean

Rename

Test Results Analyzer

Oracle Communications Cloud Native Core

Automated Test Suite

Last Successful Artifacts

TESTS-features.newfeatures.Singlesite.Stage_1.Group_1.SLF_Error_Response_Enhancement.xml	7.45 KiB	view	33
TESTS-features.newfeatures.Singlesite.Stage_1.Group_2.ProvGw_Auditor_Enhancements.xml	277.75 KiB	view	30
TESTS-features.newfeatures.Singlesite.Stage_1.Group_2.ProvGw_Auditor_Prometheus_Scenarios.xml	31.17 KiB	view	27
			24
			21
			18
			16.5

The page lists all the features. Click any feature to view description of that test case.

Back to SLF-NewFeatures

pages

My Project

Related Pages

Here is a list of all related documentation pages:

ProvGw_Auditor_Enhancements	
ProvGw_Auditor_Prometheus_Scenarios	
SLF_Error_Response_Enhancement	

The SLF new features list along with its description is as follows:

Table 4-18 SLF New Features

Feature	Description
SLF_Error_Response_Enhancement.feature	Validation of error detail in response received by SLF.
ProvGw_Auditor_Enhancements.feature	Validate the Provisioning Gateway auditor scenarios with the latest enhancements.
ProvGw_Auditor_Prometheus_Scenarios.feature	Validate the metrics on OSO and the alerts for Provisioning Gateway auditor scenarios with the latest enhancements.

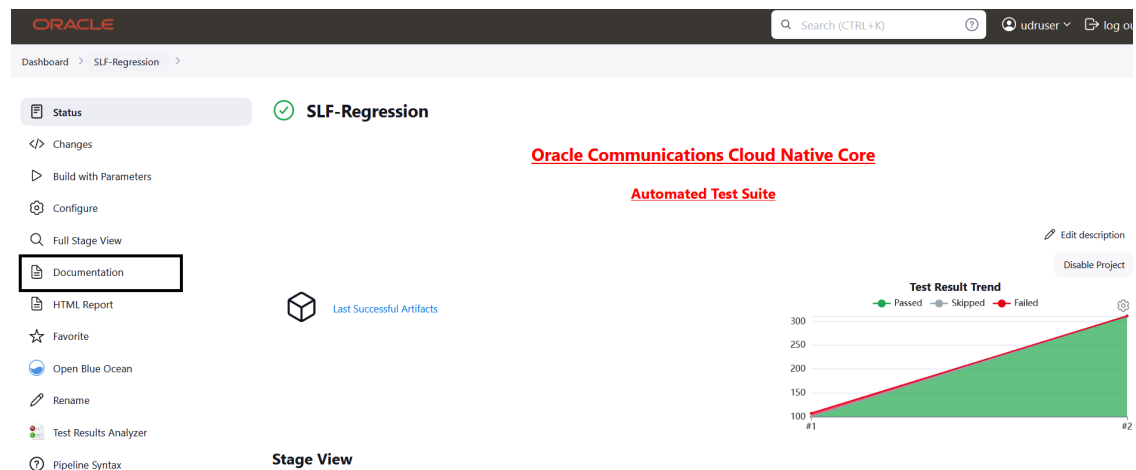
Note

The Documentation option appears only if you have run the SLF-NewFeatures pipeline test cases atleast once.

Based on the functionalities covered under Documentation, the **Build Requires Parameters** page displays test cases. To navigate back to the Pipeline SLF-NewFeatures page, click **Back to SLF-NewFeatures** link available on top left corner of the page.

4.8.6.7 SLF Regression Documentation - Single Site Scenarios

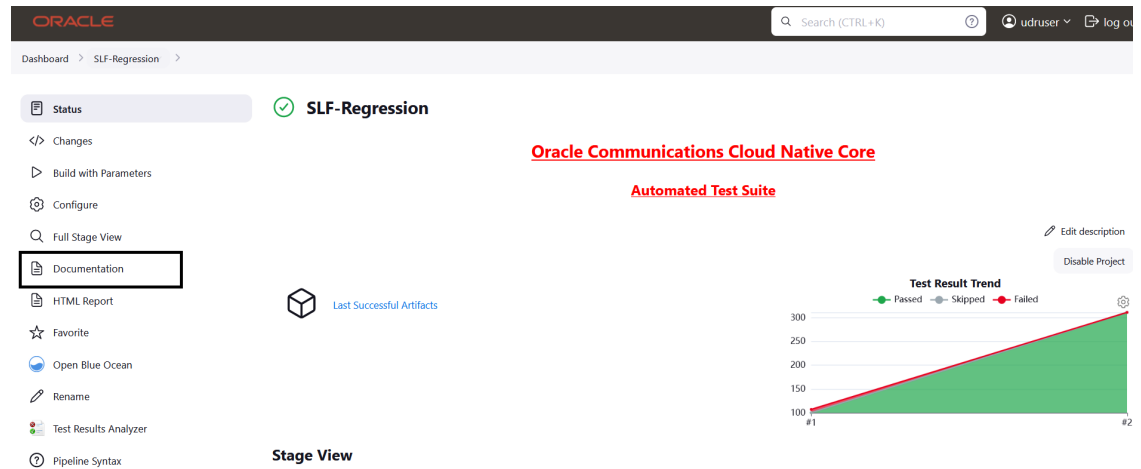
Click **Documentation** in the left navigation pane of the SLF-Regression pipeline to view all the test cases.

Figure 4-107 SLF-Regression Documentation

Click any feature to see the description of that test case.

4.8.6.8 SLF Regression Documentation - Multiple Site Scenarios

Click **Documentation** in the left navigation pane of the SLF-Regression pipeline to view all the test cases provided.

Figure 4-108 SLF-Regression Documentation

Click any feature to view its details.

4.8.7 Running UDR Pipelines

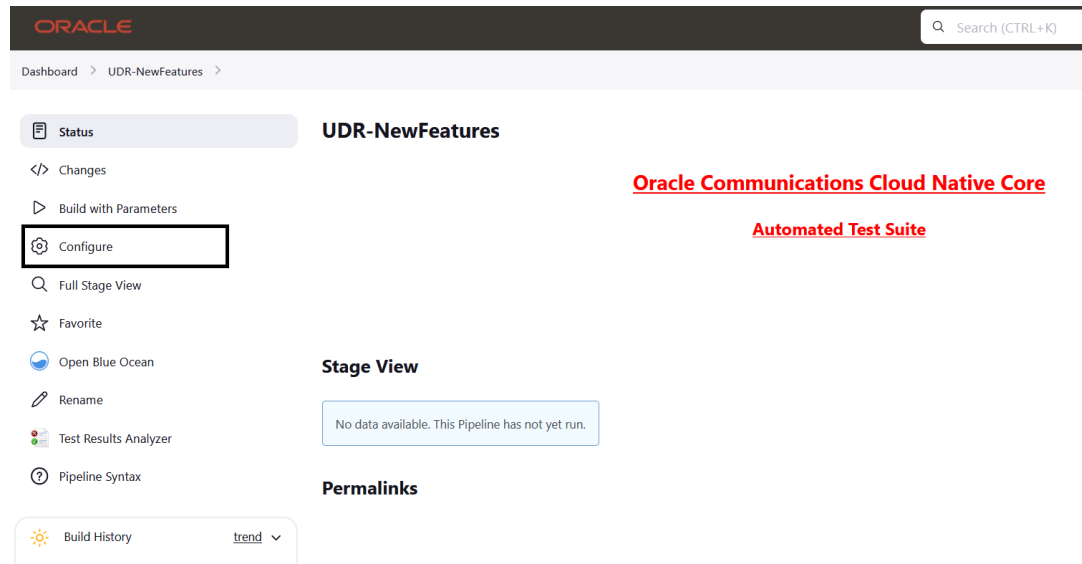
This section describes how to run UDR related pipelines. It includes:

- [Running UDR NewFeatures Pipeline](#)
- [Running UDR Regression Pipeline](#)
- [Running UDR HealthCheck Pipeline](#)
- [UDR NewFeatures Documentation](#)
- [UDR Regression Documentation](#)

4.8.7.1 Running UDR NewFeatures Pipeline

To run the UDR-NewFeatures pipeline test cases:

1. Click **UDR-NewFeatures**. Click **Configure** in the left navigation pane. The **General** tab appears. The user must wait for the page to load completely.

Figure 4-109 UDR NewFeatures Pipeline

2. Once the page loads completely, select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there is large number of build, the [Persistent Volume](#) may be utilized completely.

3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

You should not change any other value apart from **line number 63 to 103**. It means you can change the parameters marked as "**b**" - to - "**P**" as per user requirement. The parameters details are provided as comments in **line number 14 to 55**. The parameters description is as follows:

- a - Name of the NF to be tested in capital (UDR).
- b - Namespace in which the udr is deployed.
- c - Name of signalling UDR1_ingressgateway_service.namespace and port (ocudr-ingressgateway-sig.ocudr:80).
- d - Helm release name of UDR (ocudr)
- e - FQDN and Port UDR's nudr-config in the format fqdn:port (ocudr-nudr-config.ocudr:5001)
- f - Name_of_Prometheus_service.namespace and port(occne-prometheus-server.occne-infra:80).

- g - If this option is not provided, Prometheus URI shall be /api/v1 (Default: /prometheus/api/v1) (if CNE 1.9+ /<cne-cluster-name>/prometheus/api/v1)
- h - Number of times the re-run of failed case is allowed (default as 2).
- i - Name of Kubernetes Host server and port (kubernetes.default:443)
- j - diameter endpoint fqdn and port in the form fqdn:port (ocudr-nudr-diam-gateway.ocudr:3868)
- k - fqdn or IP of http2-server in the format <ip/fqdn> (fqdn) (notify-stub-service.ocudr:80)
- l - fqdn or IP for primary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server.ocudr:8080)
- m - fqdn or IP for secondary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server2.ocudr:8080)
- n - primary stub deployment name (nrfstub-ocstub-py)
- o - secondary stub deployment name (nrfstub2-ocstub-py)
- p - HTTPS port for UDR (443)
- q - root certificate authority (CA) (caroot.cer)
- r - Server certificate signed with root CA private key (apigatewayrsa.cer)
- s - private key to .pem format (rsa_private_key_pkcs1.pem)
- t - fqdn and port for 2nd diameter tool endpoint
- u - MYSQL SERVICE ENDPOINT
- v - NOTIFY STUB deployment name
- w - Name of signaling in the format fqdn:port (ocudr-s2-ingressgateway-sig.ocudr-s2:80)
- x - fqdn and port for UDR2 nudr-config (ocudr2-nudr-config.ocudr:5001)
- y - fqdn and port for UDR2 DIAMETER_ENDPOINT (fqdn:port) (ocudr-s2-nudr-diam-gateway.ocudr-s2:3868)
- z - FQDN and port for SCP stub (scp-stub-service:8080)
- A - Deployment name for SCP stub (scpstub-notify-ocstub-py)
- B - Bulk import helm name
- C - ProvGw namespace (ocudr)
- D - ProvGw endpoint in format fqdn:port (provgw-prov-ingressgateway.ocudr:80)
- E - ProvGw endpoint https port(443)
- F - ProvGw helm name
- G - ProvGw config host in the format fqdn:port(provgw-provgw-config.ocudr:5001)
- H - true if OCCNE is 1.9.x and above, else false (true)
- I - fqdn and port for fourg stub in the form (fqdn:port)
- J - fqdn and port for 1st diameter tool endpoint
- K - port for notify stub for TLS scenarios (8443)
- L - Service name of UDR1 provisioning ingressgateway in the format FQDN:PORT(ocudr-ingressgateway-prov.ocudr:80)

- M - Service name of site2 UDR provisioning ingressgateway in the format FQDN:PORT(ocudr2-ingressgateway-prov.ocudr-s2:80)
- N - Diameter Stub Tool1 Deploy Name (diamtoola-diam-stub)
- O - Diameter Stub Tool2 Deploy Name (diamtoolb-diam-stub)
- P - Migration tool Helm release name (migration-tool)

Figure 4-110 UDR NewFeatures Pipeline Configuration

Dashboard > UDR-NewFeatures > Configuration

Configure

- General
- Advanced Project Options
- Pipeline**

Pipeline

Definition

Pipeline script

```

1 node {
2   //a = NF
3   //e = NUDR_CONFIG_ENDPOINT
4   //l = KUBERNETES_HOST_PORT
5   //m = NMF_STUB_DATA
6   //q = CAPFILE
7   //u = DR_SVC_ENDPOINT
8   //y = UDR2_DIAM_ENDPOINT
9   //c = PROVDW_NAMESPACE
10  //g = PROVDW_CONFIG_HOST_PORT
11  //k = TLS_STUB_HOST_PORT
12  //o = DIAM_STUB2_DEPLOY_NAME
13  //Description of Variables:
14  //SELECTED_NF : UDR
15  //SLF_NAMESPACE : UDR Namespace
16  //UDR1_SIG_FQDN_PORT : Name of signalling UDR ingressgateway_service.namespace and port in the format FQDN:PORT(ocudr-ingressgateway-sig.ocudr)
17  //UDR1_Helm_Name : Helm Name for UDR1 (ocudr)
18  //b = UDR_NAMESPACE
19  //f = PROVDW_FQDN_PORT
20  //j = DIAM_ENDPOINT
21  //n = NMF_STUB_DEPLOY_NAME
22  //r = CERTFILE
23  //v = HTTP_STUB_DEPLOY_NAME
24  //z = SCP_STUB_FQDN_PORT
25  //D = PROVDW_FQDN_PORT
26  //H = CHE_VERSION_19
27  //L = UDR1_PROVDW_FQDN_PORT
28  //P = MIGRATION_TOOL_RELEASE_NAME
29  //c = UDR1_SIG_FQDN_PORT
30  //g = PROVDW_SVC_URI
31  //k = HTTP_STUB_FQDN_PORT
32  //o = NMF_STUB_DEPLOY_NAME
33  //s = KEYFILE
34  //u = UDR2_SIG_FQDN_PORT
35  //w = SCP_STUB_DEPLOY_NAME
36  //E = Provdw_HTTPS_Port
37  //I = FQDN_MIG_STUB_ENDPOINT
38  //H = UDR2_PROVDW_FQDN_PORT
39  //d = UDR1_Helm_Name
40  //h = RERUN_COUNT
41  //l = NMF_STUB_DATA
42  //p = UDR1_HTTPS_PORT
43  //t = DIAM_TOOL2_ENDPOINT
44  //x = UDR2_NUDR_CONFIG_ENDPOINT
45  //B = BULK_HELM_NAME
46  //F = Provdw_Helm_Name
47  //3 = DIAM_TOOL1_ENDPOINT
48  //N = DIAM_STUB1_DEPLOY_NAME

```

☒ Use Groovy Sandbox

Pipeline Syntax

Save Apply

4. Click **Save** after making necessary changes. The **Pipeline UDR-NewFeatures** page appears.
5. Click **Build with Parameters**. The page lists all the newly added features and displays the total number of features and the scenarios in each feature, name of each feature, and scenarios in each feature.

Figure 4-111 UDR NewFeatures Pipeline Build with Parameter

Dashboard > UDR-NewFeatures >

Pipeline UDR-NewFeatures

This build requires parameters:

Oracle Communications Cloud Native Core

Automated Test Suite - UDR

EXECUTION OPTIONS			
TestSuite NewFeatures	SUT UDR	SUTSuite All	Configuration_Type Product_Config
Include_Regression NO	Fetch_Log_Upon_Failure NO	Log_Type Not Applicable	Log_Level Not Applicable

FEATURES AND TESTCASES

Features: All

Stage: Stage_1

Group: sequential_execution

- UDR_Subscriber_Export_ToolAdditional_Scenarios
- UDR_Subscriber_ActivityLogging_New
- UDR_Suppressed_notification_to_the_initiator
- UDR_ErrorResponseDetail
- UDR_23_4_0_Bug_fixes

Build Cancel

- Select the Configuration_Type. For more information, see [Parameterization](#).
 - Choose the **Select_Option(s)**, which are:
 - **All:** By default, it selects all the UDR test cases to run. Scroll-down and click **Build** to run all the test cases.
 - **Single/MultipleFeatures:** This option allows you to select any number of test cases you want to run from the list of total test cases available for execution.
6. After selecting the test cases, scroll-down and click **Build** to run the selected UDR test cases.

Figure 4-112 Consolidated Test Report

```
Dashboard > UDR-NewFeatures > #2

[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_1 Group: sequential_execution Result)
+ set +X
*****Stage: Stage_1 Group: sequential_execution Result*****
Final Result:-
Total-Features RUN 4, PASS 3, FAIL 1

[Pipeline] }
[Pipeline] // catchError
[Pipeline] sh (Overall Result)
+ set +X
*****Overall Result*****
Failing scenarios:
Stage:Stage_1 Group:sequential_execution
  UDR_23_4_0_Bug_fixes.feature:281 Sequential_update_entity_03

Took 0h12m49s

Final Result:-
Total-Features RUN 4, PASS 3, FAIL 1
```

4.8.7.2 Running UDR Regression Pipeline

To run the UDR-Regression pipeline test cases:

Running UDR Regression Pipeline: Model 1: Single Site Deployment

1. Click **UDR-Regression** and then, click **Configure**.

Figure 4-113 UDR Regression Pipeline

2. The **General** tab appears. The user must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there are large number of builds, the [Persistent Volume](#) may be utilized completely.

4. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than from **line number 63 to 103**. You can change the parameters marked as "**b**" - to - "**P**" as per user requirement. The parameters details are provided as comments in **line number 14 to 55**. The parameters description is as follows:

- a - Name of the NF to be tested in capital (UDR).
- b - Namespace in which the udr is deployed.
- c - Name of signalling UDR1_ingressgateway_service.namespace and port (ocudr-ingressgateway-sig.ocudr:80).
- d - Helm release name of UDR (ocudr)
- e - FQDN and Port UDR's nudr-config in the format fqdn:port (ocudr-nudr-config.ocudr:5001)
- f - Name_of_Prometheus_service.namespace and port(occne-prometheus-server.occne-infra:80).

- g - If this option is not provided, Prometheus URI shall be /api/v1 (Default: /prometheus/api/v1) (if CNE 1.9+ /<cne-cluster-name>/prometheus/api/v1)
- h - Number of times the re-run of failed case is allowed (default as 1).
- i - Name and port of Kubernetes Host server in the format FQDN:PORT(kubernetes.default:443).
- j - fqdn and port for DIAMETER_ENDPOINT in the format FQDN:PORT (ocudr-nudr-diam-gateway:3868)
- k - fqdn or IP of http2-server in the format <ip/fqdn> (fqdn) (notify-stub-service.ocudr:80)
- l - fqdn or IP for primary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server.ocudr:8080)
- m - fqdn or IP for secondary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server2.ocudr:8080)
- n - primary stub deployment name (nrfstub-ocstub-py)
- o - secondary stub deployment name (nrfstub2-ocstub-py)
- p - HTTPS port for UDR (443)
- q - root certificate authority (CA) (caroot.cer)
- r - Server certificate signed with root CA private key (apigatewayrsa.cer)
- s - private key to .pem format (rsa_private_key_pkcs1.pem)
- t - fqdn and port for 2nd diameter tool endpoint
- u - MYSQL SERVICE ENDPOINT
- v - NOTIFY STUB deployment name
- w - Name of signaling in the format fqdn:port (ocudr-s2-ingressgateway-sig.ocudr-s2:80)
- x - fqdn and port for UDR2 nudr-config (ocudr2-nudr-config.ocudr:5001)
- y - fqdn and port for UDR2 DIAMETER_ENDPOINT (fqdn:port) (ocudr-s2-nudr-diam-gateway.ocudr-s2:3868)
- z - FQDN and port for SCP stub (scp-stub-service:8080)
- A - Deployment name for SCP stub (scpstub-notify-ocstub-py)
- B - Bulk import helm name
- C - ProvGw namespace (ocudr)
- D - ProvGw endpoint in format fqdn:port (provgw-prov-ingressgateway.ocudr:80)
- E - ProvGw endpoint https port(443)
- F - ProvGw helm name
- G - ProvGw config host in the format fqdn:port(provgw-provgw-config.ocudr:5001)
- H - true if OCCNE is 1.9.x and above, else false (true)
- I - fqdn and port for fourg stub in the form (fqdn:port)
- J - fqdn and port for 1st diameter tool endpoint
- K - port for notify stub for TLS scenarios (8443)
- L - Service name of UDR1 provisioning ingressgateway in the format FQDN:PORT(ocudr-ingressgateway-prov.ocudr:80)

- M - Service name of site2 UDR provisioning ingressgateway in the format FQDN:PORT(ocudr2-ingressgateway-prov.ocudr-s2:80)
 - N - Diameter Stub Tool1 Deploy Name (diamtoola-diam-stub)
 - O - Diameter Stub Tool2 Deploy Name (diamtoolb-diam-stub)
 - P - Migration tool helm release name (migration-tool)
5. Click **Save**. The **Pipeline UDR-Regression** page appears.
 6. Click **Build with Parameters**. A page listing all the regression features and displays the total count of the features and displays the total count of the features, count of scenarios in each feature, name of each feature, and scenarios in each feature.

Figure 4-114 UDR Regression Pipeline Site Deployment

The screenshot displays the 'Pipeline UDR-Regression' page in the Oracle Cloud Native Core Automated Test Suite. The page is titled 'Oracle Communications Cloud Native Core Automated Test Suite - UDR'. It features a sidebar with navigation options: Status, Changes, Build with Parameters, Configure, Full Stage View, Favorite, Open Blue Ocean, Rename, Test Results Analyzer, and Pipeline Syntax. The main content area is divided into two sections: 'EXECUTION OPTIONS' and 'FEATURES AND TESTCASES'. The 'EXECUTION OPTIONS' section includes fields for TestSuite (Regression), SUT (UDR), SUTSuite (All), Configuration_Type (Product_Config), Include_NewFeatures (NO), Fetch_Log_Upon_Failure (NO), Log_Type (Not Applicable), and Log_Level (Not Applicable). The 'FEATURES AND TESTCASES' section shows a list of features grouped by Stage. Stage 1 Group: bulk_import_metrics_autocreate_true includes UDR_Bulk_Import and UDR_Bulk_Import_xml_part1. Stage 1 Group: diameter_only_metrics includes Diameter_All_Entities. The page also has a 'Build History' section showing 'No builds' and a 'Build' button at the bottom.

7. Click **Build** to run the test cases.

Figure 4-115 Consolidated Test Report

```

[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_3 Group: provgw_no_igw_dr_metrics Result)
+ set +x
*****Stage: Stage_3 Group: provgw_no_igw_dr_metrics Result*****
Final Result:-
Total-Features RUN 15, PASS 15, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_3 Group: udr_no_igw_dr_metrics Result)
+ set +x
*****Stage: Stage_3 Group: udr_no_igw_dr_metrics Result*****
Final Result:-
Total-Features RUN 5, PASS 5, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] catchError
[Pipeline] {
[Pipeline] sh (Stage: Stage_4 Group: sequential_execution Result)
+ set +x
*****Stage: Stage_4 Group: sequential_execution Result*****

```

Note

- For information about Configuration_Type, see [Parameterization](#).
- The **SUITSuite** option selected as 'ALL' will run all suites except multiple site scenarios.

Running UDR Regression Pipeline: Model 2: Multiple Site Deployment

1. Click **UDR-Regression** and then, click **Configure**.

Figure 4-116 UDR Regression Pipeline

2. The **General** tab appears. The user must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there are large number of builds, the [Persistent Volume](#) may be utilized completely.

4. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than from **line number 63 to 103**. You can change the parameters marked as "**b**" - to - "**P**" as per user requirement. The parameters details are provided as comments in line number **14 to 55**. The parameters description is as follows:

- a - Name of the NF to be tested in capital (UDR).
- b - Namespace in which the udr is deployed.
- c - Name of signalling UDR1_ingressgateway_service.namespace and port (ocudr-ingressgateway-sig.ocudr:80).
- d - Helm release name of UDR (ocudr)
- e - FQDN and Port UDR's nudr-config in the format fqdn:port (ocudr-nudr-config.ocudr:5001)
- f - Name_of_Prometheus_service.namespace and port(occne-prometheus-server.occne-infra:80).

- g - If this option is not provided, Prometheus URI shall be /api/v1 (Default: / prometheus/api/v1) (if CNE 1.9+ /<cne-cluster-name>/prometheus/api/v1)
- h - Number of times the re-run of failed case is allowed (default as 1).
- i - Name and port of Kubernetes Host server in the format FQDN:PORT(kubernetes.default:443).
- j - fqdn and port for DIAMETER_ENDPOINT in the format FQDN:PORT (ocudr-nudr-diam-gateway:3868)
- k - fqdn or IP of http2-server in the format <ip/fqdn> (fqdn) (notify-stub-service.ocudr:80)
- l - fqdn or IP for primary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server.ocudr:8080)
- m - fqdn or IP for secondary nrf stub and port in the format <ip/fqdn>:<port> (nrf-stub-server2.ocudr:8080)
- n - primary stub deployment name (nrfstub-ocstub-py)
- o - secondary stub deployment name (nrfstub2-ocstub-py)
- p - HTTPS port for UDR (443)
- q - root certificate authority (CA) (caroot.cer)
- r - Server certificate signed with root CA private key (apigatewayrsa.cer)
- s - private key to .pem format (rsa_private_key_pkcs1.pem)
- t - fqdn and port for 2nd diameter tool endpoint
- u - MYSQL SERVICE ENDPOINT
- v - NOTIFY STUB deployment name
- w - Name of signaling in the format fqdn:port (ocudr-s2-ingressgateway-sig.ocudr-s2:80)
- x - fqdn and port for UDR2 nudr-config (ocudr2-nudr-config.ocudr:5001)
- y - fqdn and port for UDR2 DIAMETER_ENDPOINT (fqdn:port) (ocudr-s2-nudr-diam-gateway.ocudr-s2:3868)
- z - FQDN and port for SCP stub (scp-stub-service:8080)
- A - Deployment name for SCP stub (scpstub-notify-ocstub-py)
- B - Bulk import helm name
- C - ProvGw namespace (ocudr)
- D - ProvGw endpoint in format fqdn:port (provgw-prov-ingressgateway.ocudr:80)
- E - ProvGw endpoint https port(443)
- F - ProvGw helm name
- G - ProvGw config host in the format fqdn:port(provgw-provgw-config.ocudr:5001)
- H - true if OCCNE is 1.9.x and above, else false (true)
- I - fqdn and port for fourg stub in the form (fqdn:port)
- J - fqdn and port for 1st diameter tool endpoint
- K - port for notify stub for TLS scenarios (8443)
- L - Service name of UDR1 provisioning ingressgateway in the format FQDN:PORT(ocudr-ingressgateway-prov.ocudr:80)

- M - Service name of site2 UDR provisioning ingressgateway in the format FQDN:PORT(ocudr2-ingressgateway-prov.ocudr-s2:80)
 - N - Diameter Stub Tool1 Deploy Name (diamtoola-diam-stub)
 - O - Diameter Stub Tool2 Deploy Name (diamtoolb-diam-stub)
 - P - Migration tool helm release name (migration-tool)
5. Click **Save**. The **Pipeline UDR-Regression** page appears.
 6. Click **Build with Parameters**. A page listing all the regression features and displays the total count of the features and displays the total count of the features, count of scenarios in each feature, name of each feature, and scenarios in each feature.

Figure 4-117 UDR Regression Pipeline Multiple Site Deployment

Oracle Communications Cloud Native Core
Automated Test Suite - UDR

EXECUTION OPTIONS

TestSuite Regression	SUT UDR	SUTSuite Multi_Site	Configuration_Type Product_Config
Include_NewFeatures NO	Fetch_Log_Upon_Failure NO	Log_Type Not Applicable	Log_Level Not Applicable

FEATURES AND TESTCASES

Features
All

Stage: Multi_Site Group: two_site
• UDR_Multisite

Build Cancel

7. Click **Build** to run the test cases.

Note

- For information about Configuration_Type, see [Parameterization](#).
- The **SUTSuite** option selected as 'MULTISITE' will run multiple site scenarios only.

Figure 4-118 Consolidated Test Report

```
*****Overall Result*****

Took 0h7m14s

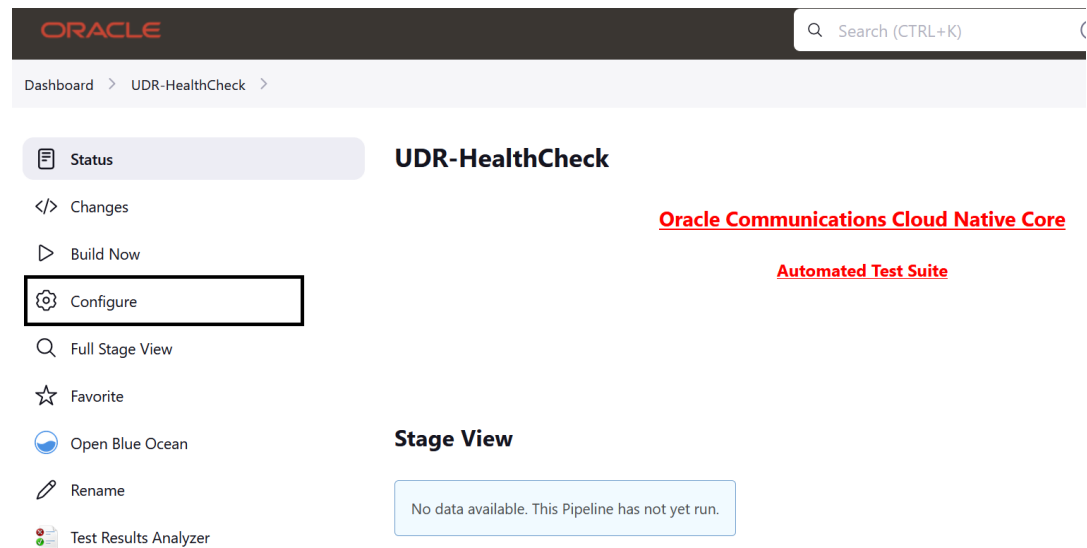
Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0
```


4.8.7.3 Running UDR HealthCheck Pipeline

To run health check on UDR:

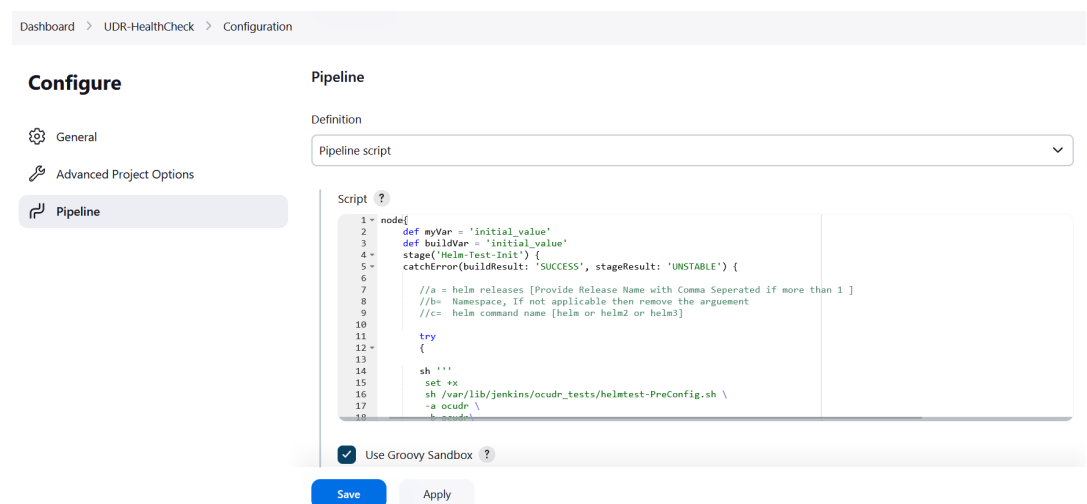
1. Navigate to **UDR-HealthCheck** pipeline and click **Configure**.

Figure 4-119 UDR-HealthCheck Pipeline



2. The **General** tab appears. The user must wait for the page to load completely.
3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

Figure 4-120 UDR-HealthCheck Configuration



! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than **line numbers 17 and 18**. You can change the parameters marked as "**a**", "**b**" and "**c**" as per user requirement. The parameters details are provided as comments in **line number 7 and 9**. The parameters description is as follows:

- a - helm releases [Provide Release Name with Comma Separated if more than 1]
 - b - namespace of SUT
 - c - helm command alias (helm or helm2 or helm3)
4. Click **Save**. The **Pipeline UDR-HealthCheck** page appears.
 5. Click **Build Now**. This triggers health check for SUT.

Figure 4-121 UDR-HealthCheck Build Now

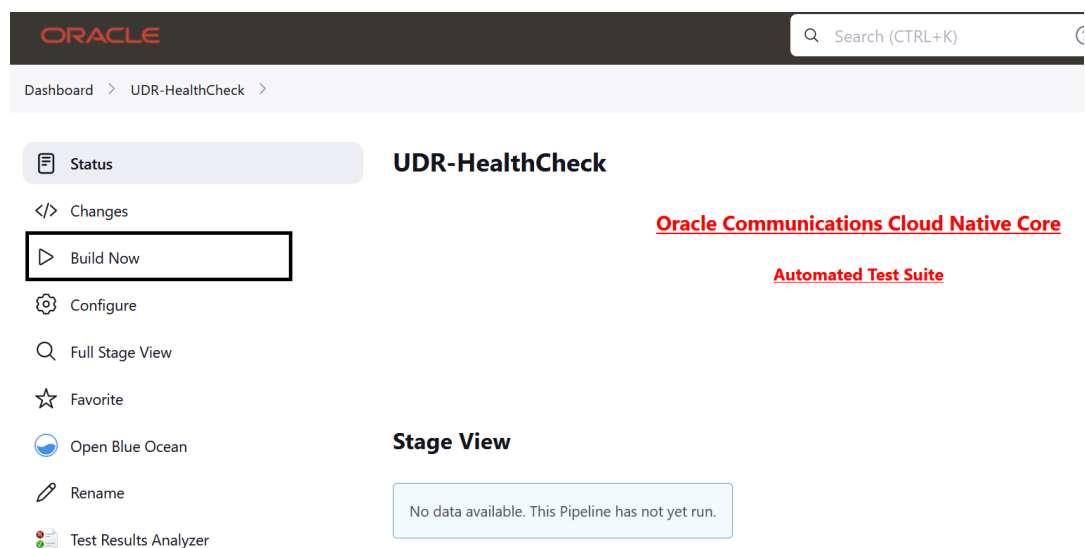


Figure 4-122 Sample Output

```

NAME: ocudr-ravi
LAST DEPLOYED: Sun Oct 15 19:15:05 2023
NAMESPACE: ats-ravi
STATUS: deployed
REVISION: 1
TEST SUITE:   ocudr-ravi-test
Last Started: Wed Oct 18 17:47:13 2023
Last Completed: Wed Oct 18 17:47:38 2023
Phase:        Succeeded
NOTES:
# Copyright 2019 (C), Oracle and/or its affiliates. All rights reserved.

Thank you for installing ocudr.

Your release is named ocudr-ravi , Release Revision: 1.
To learn more about the release, try:

$ helm status ocudr-ravi
$ helm get ocudr-ravi
[0m

[Pipeline] }
[Pipeline] // catchError
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS

```

4.8.7.4 UDR NewFeatures Documentation

To view the documentation of UDR-NewFeatures, click **Documentation** in the left navigation pane of the UDR-NewFeatures pipeline to view all the UDR new features test cases.

Figure 4-123 UDR-NewFeatures Documentation

The screenshot displays the Oracle Cloud Native Core Automated Testing Suite (ATS) interface. At the top, there is a search bar with the text "Search (CTRL+K)". Below the search bar, the breadcrumb navigation shows "Dashboard > UDR-NewFeatures > #2". The main content area is titled "Build #2 (Mar 8, 2024, 1:52:12 PM)" and includes a red "X" icon. On the left, a sidebar lists various options: "Status", "Changes", "Console Output", "View as plain text", "View Build Information", "Parameters", "Documentation" (highlighted with a red box), "HTML Report", "Test Result", "Open Blue Ocean", and "Replay from State". The main content area shows "Build Artifacts" with a list of files and their sizes: "TESTS-features.newfeatures.Singlesite.AllStage_1.sequential_execution.UDR_23_4_0_Bug_fixes.xml" (196.77 KIB), "TESTS-features.newfeatures.Singlesite.AllStage_1.sequential_execution.UDR_ErrorResponseDetail.xml" (16.06 KIB), "TESTS-features.newfeatures.Singlesite.AllStage_1.sequential_execution.UDR_Subscriber_Activity_Logging_New.xml" (4.41 KIB), and "TESTS-features.newfeatures.Singlesite.AllStage_1.sequential_execution.UDR_Subscriber_Export_Tool_Additional_Scenarios.xml" (11.90 KIB). Below the artifacts, it shows "Started by user udruser" and "Replayed #1 (diff)". At the bottom, there is a "Test Result" section indicating "1 failure / ±0".

UDR new features list along with its description is as follows:

Table 4-19 UDR New Features List

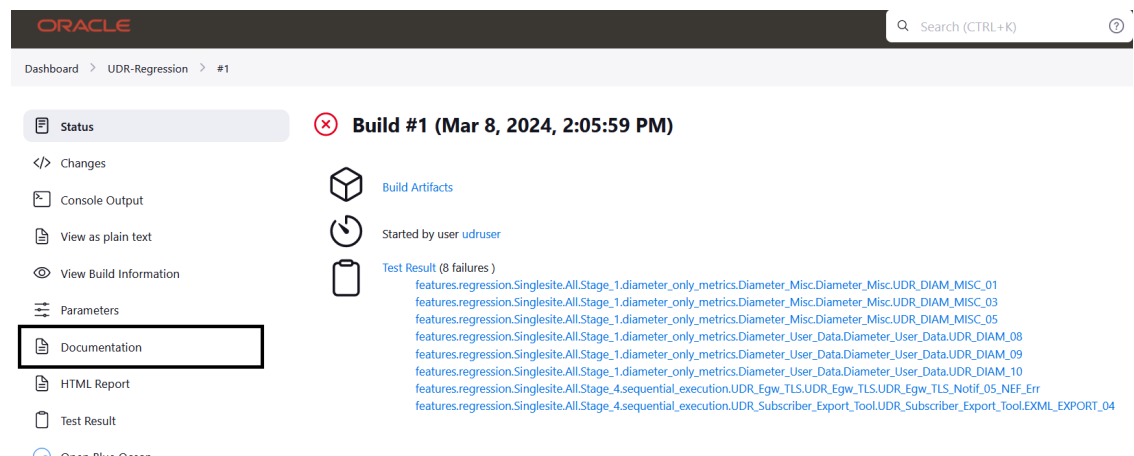
Feature	Description
UDR_Subscriber_Export_Tool_Additional_Scenarios.feature	Validates the additional scenarios for export of UDR subscribers.
UDR_Subscriber_Activity_Logging_New.feature	Validates subscriber activity logging for VSA POST requests.
UDR_ErrorResponseDetail.feature	Validates the error details in response received from UDR.
UDR_23_4_0_Bug_fixes.feature	Validates the additional scenarios similar to the bug fixes made in 23.4.0.
UDR_Suppressed_notification_to_the_initiator	Validates the scenarios where notifications to the subscribed initiating node is suppressed.

Click any feature to view its description.

4.8.7.5 UDR Regression Documentation

Click **Documentation** in the left navigation pane of the UDR-Regression pipeline to view all the UDR regression test cases.

Figure 4-124 UDR-Regression Documentation



Click any feature to view its description.

4.8.8 Running EIR Pipelines

This section describes how to run EIR related pipelines. It includes:

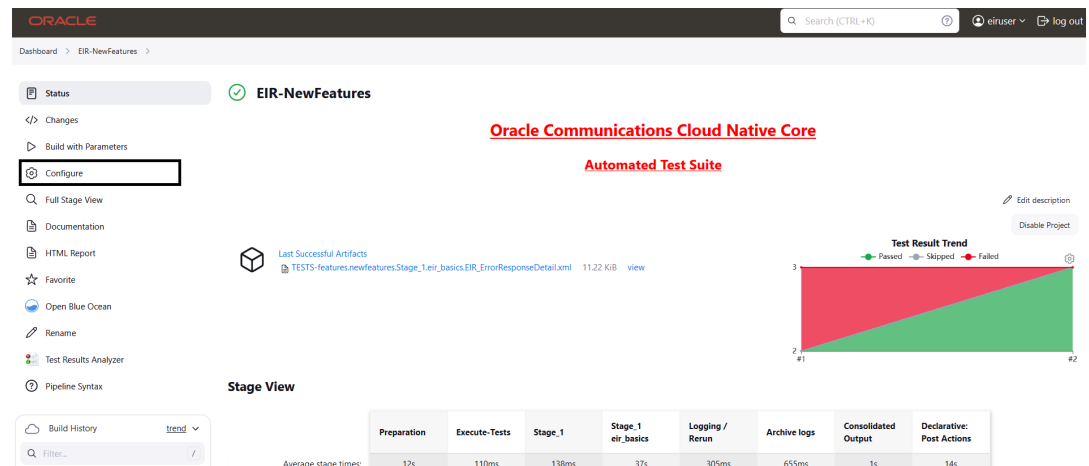
- [Running EIR NewFeatures Pipeline](#)
- [Running EIR HealthCheck Pipeline](#)
- [EIR NewFeatures Documentation](#)

4.8.8.1 Running EIR NewFeatures Pipeline

To run the EIR-NewFeatures pipeline test cases:

1. Click **EIR-NewFeatures**. Click **Configure** in the left navigation pane. The **General** tab appears. The user must wait for the page to load completely.

Figure 4-125 EIR NewFeatures Pipeline



2. Once the page loads completely, select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there is large number of build, the [Persistent Volume](#) may be utilized completely.

3. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration as follows:

Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

You should not change any other value apart from **line number 45 to 74**. It means you can change the parameters marked as "**b**" - to - "**C**" as per user requirement. The parameters details are provided as comments in **line number 11 to 39**. The parameters description is as follows:

- a - Selected NF.
- b - NameSpace in which UDR is Deployed.
- c - EIR signaling FQDN and PORT in the format FQDN:PORT.
- d - EIR provisioning FQDN and PORT in the format FQDN:PORT.
- e - EIR config service FQDN and PORT in the format FQDN:PORT.

- f - FQDN:PORT format for primary nrf-stub.
- g - FQDN:PORT format for secondary nrf-stub.
- h - FQDN and PORT of Kubernetes Host server in the format FQDN:PORT.
- i - Helm name of EIR.
- j - primary nrf-stub deployment name.
- k - secondary nrf-stub deployment name.
- l - primary nrf-stub metric port
- m - secondary nrf-stub metric port
- n - bulk import Helm name
- o - Number of retries to check if pod down is successful.
- p - Number of retries to check if pod up is successful.
- q - Number of retries when ATS detects tcp connection failures.
- r - Time interval in seconds between tcp connection retries.
- s - re-run count.
- t - FQDN and PORT of Prometheus server in the format FQDN:PORT.
- u - prometheus uri apiName and apiVersion.
- v - EIR HTTPS port.
- w - root certificate authority for signalling ingressgateway.
- x - Server certificate signed with root CA private key for signalling ingressgateway.
- y - private key in .pem format for signalling ingressgateway.
- z - root certificate authority for provisioning ingressgateway.
- A - Server certificate signed with root CA private key for provisioning ingressgateway.
- B - Private key in .pem format for provisioning ingressgateway.
- C - True if HA prometheus else false (default: true)

Figure 4-126 EIR NewFeatures Pipeline Configuration

Dashboard > EIR-NewFeatures > Configuration

Configure

General

Advanced Project Options

Pipeline

Pipeline

Definition

Pipeline script

```

1 = node ('built-in') {
2   //a = SELECTED_IP
3   //a = EIR_CONF_FQDN_PORT
4   //l = EIR_HELP_NAME
5   //n = STUB2_METRICS_PORT
6   //q = HTTP_CONNECT_RETRY
7   //u = PROMVCURTI
8   //y = KEYFILE
9   //C = HA_Prom
10  //Description of Variables:
11  //SELECTED_IP : EIR
12  //EIR_NAMESPACE : EIR Namespace (ocelr)
13  //EIR_SIG_FQDN_PORT : EIR signaling fqdn and port in the format FQDN:PORT (ocelr-ingressgateway-sig.ocelr:88)
14  //EIR_PROV_FQDN_PORT : EIR provisioning fqdn and port in the format FQDN:PORT (ocelr-ingressgateway-sig.ocelr:88)
15  //EIR_CONF_FQDN_PORT : EIR nuds-config fqdn and port in the format FQDN:PORT (ocelr-nuds-config.ocelr:5881)
16  //STUB1_FQDN_PORT : Primary NRF Stub fqdn and port in the format FQDN:PORT (nrf-stub-service1.ocelr:8888)
17  //STUB2_FQDN_PORT : Secondary NRF Stub fqdn and port in the format FQDN:PORT (nrf-stub-service2.ocelr:8888)

```

☒ Use Groovy Sandbox ?

[Pipeline Syntax](#)

Save Apply

Note

Setting rerun count, number of retries to check pod status and segment details:

- Rerun count can be set to any desired value by changing option 's' in the pipeline script.
- Number of retries to check if pod is down can be set by providing option 'o' in the pipeline script.
- Number of retries to check if pod is up can be set by providing option 'p' in the pipeline script.
- If ProvGw is deployed with segment details having UDR fqdn and port, provide option 'l' as port.
- Option 'u' will take the value '<domain name>/prometheus/api/v1' for CNE version 1.9 and above or OSO version 1.10.x and above. Run the below command, If domain name is not known. The domain name can be found after executing the command from ATS pod:
`curl http://<prometheus fqdn>:<port>/`

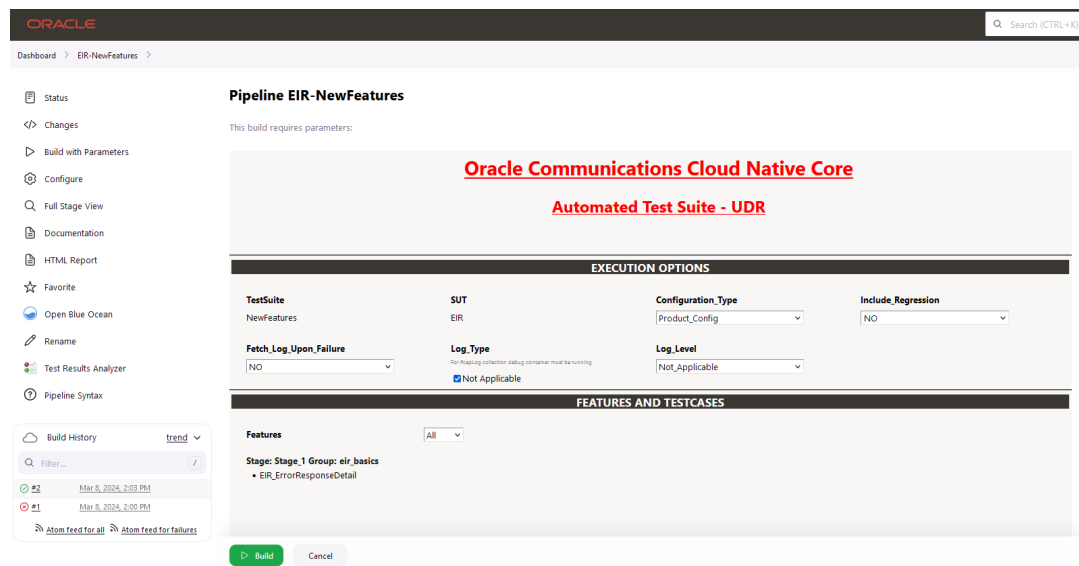
Example:

```
curl http://ocne-kube-prom-stack-kube-prometheus.ocne-infra/
Found
```

4. Click **Save** after making necessary changes. The **Pipeline EIR-NewFeatures** page appears.

Figure 4-127 EIR Build with Parameters





5. Click **Build with Parameters**. The page lists all the newly added features and displays the total count of features, counts of scenarios in each feature, name of each feature, and scenarios in each feature.
6. To view consolidated and detailed stack trace results in case of any failures, click Test Results Analyzer in the left navigation pane. The test results analyzer report appears. For more information, see [Test Results Analyzer](#).

Figure 4-128 EIR-NewFeatures Pipeline Consolidated Automation Suite Test Report

```

The below are the consolidated output of all the features that are run
[Pipeline] catchError
[Pipeline] { (hide)
[Pipeline] sh (Stage: Stage_1 Group: eir_basics Result)
+ set +x
*****Stage: Stage_1 Group: eir_basics Result*****
Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0

[Pipeline] }
[Pipeline] // catchError
[Pipeline] sh (Overall Result)
+ set +x
*****Overall Result*****

Took 0h1m3s

Final Result:-
Total-Features RUN 1, PASS 1, FAIL 0

```

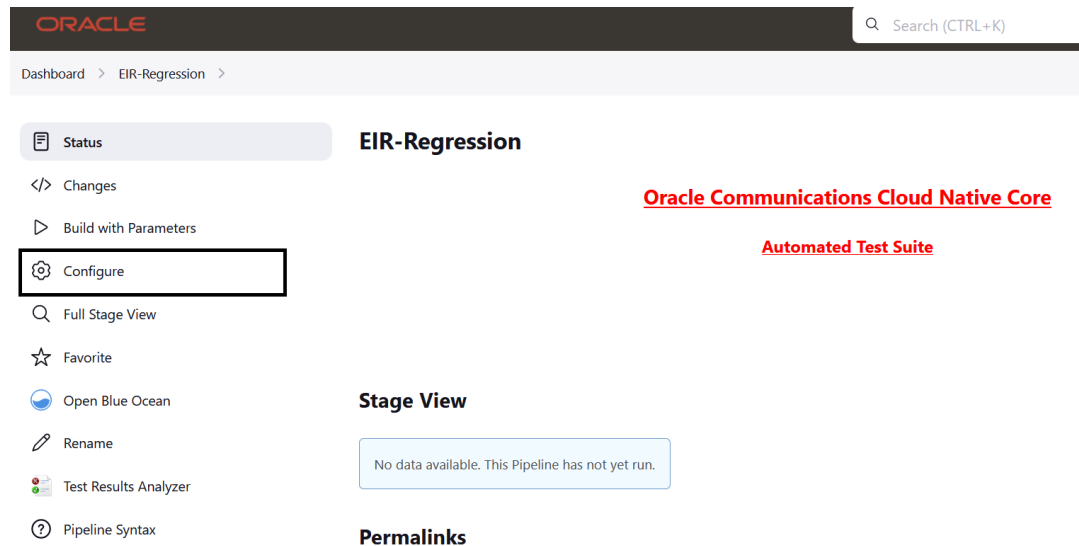
For more details on consolidated test report, see [Final Summary Report, Build Color and Application Logs](#).

4.8.8.2 Running EIR Regression Pipeline

Perform the below steps to run the EIR-Regression pipeline test cases:

1. Click **EIR-Regression** and then, click **Configure**.

Figure 4-129 EIR Regression Pipeline

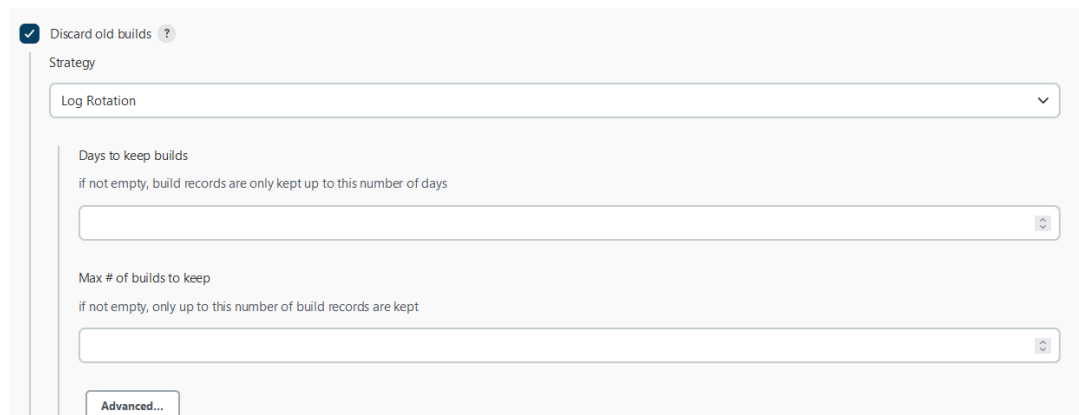


2. The **General** tab appears. The user must wait for the page to load completely.
3. Select the **Discard old Builds**. This option allows you to configure the number of builds you want to retain in the persistent volume.

Note

It is recommended to configure this option. If there are large number of builds, the [Persistent Volume](#) may be utilized completely.

Figure 4-130 Discard Builds



4. Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than from **line number 47 to 74**. You can change the parameters marked as "**b**" - **to** - "**C**" as per user requirement. The parameters details are provided as comments in **line number 11 to 39**. The parameters description is as follows:

- a - Selected NF
- b - NameSpace in which UDR is Deployed
- c - EIR signaling FQDN and PORT in the format FQDN:PORT
- d - EIR provisioning FQDN and PORT in the format FQDN:PORT
- e - EIR config service FQDN and PORT in the format FQDN:PORT
- f - FQDN:PORT format for primary nrf-stub
- g - FQDN:PORT format for secondary nrf-stub
- h - FQDN and PORT of Kubernetes Host server in the format FQDN:PORT
- i - helm name of EIR
- j - primary nrf-stub deployment name
- k - secondary nrf-stub deployment name
- l - primary nrf-stub metric port
- m - secondary nrf-stub metric port
- n - bulk import Helm name
- o - Number of retries to check if pod down is successful
- p - Number of retries to check if pod up is successful
- q - Number of retries when ATS detects tcp connection failures
- r - Time interval in seconds between tcp connection retries
- s - re-run count
- t - FQDN and PORT of Prometheus server in the format FQDN:PORT
- u - prometheus uri apiName and apiVersion
- v - EIR HTTPS port
- w - root certificate authority for signalling ingressgateway
- x - Server certificate signed with root CA private key for signalling ingressgateway
- y - private key in .pem format for signalling ingressgateway
- z - root certificate authority for provisioning ingressgateway
- A - Server certificate signed with root CA private key for provisioning ingressgateway
- B - Private key in .pem format for provisioning ingressgateway
- C - True if HA prometheus else false(default: true)

5. Click **Save**. The **Pipeline EIR-Regression** page appears.

Figure 4-131 EIR Regression Configuration

Dashboard > EIR-Regression > Configuration

Configure

- General
- Advanced Project Options
- Pipeline**

Pipeline

Definition

Pipeline script

```

1 = node ('built-in'){
2   //a = SELECTED_NF      b = EIR_NAMESPACE      c = EIR_SIG_FQDN_PORT      d = EIR_PROV_FQDN_PORT
3   //m = EIR_CONF_FQDN_PORT      f = STUB1_FQDN_PORT      g = STUB2_FQDN_PORT      h = KUBERNETES_FQDN_PORT
4   //s = EIR_HELM_NAME      j = STUB1_DEPLOY_NAME      k = STUB2_DEPLOY_NAME      l = STUB1_METRICS_PORT
5   //n = STUB2_METRICS_PORT      n = BULK_IMPORT_DEP_NAME      o = POD_DOWN_RETRIES      p = POD_UP_RETRIES
6   //q = HTTP_CONNECT_RETRY      r = HTTP_RETRY_INTERVAL      s = RERUN_COUNT      t = PROM_FQDN_PORT
7   //u = PROMSCURL      v = EIR_HTTPS_PORT      w = CAFILE      x = CERTFILE
8   //y = KEYFILE      z = CAFILE_PROV      A = CERTFILE_PROV      B = KEYFILE_PROV
9   //C = HA_Prom
10  //Description of Variables:
11  //SELECTED_NF : EIR
12  //EIR_NAMESPACE : EIR Namespace (ocair)
13  //EIR_SIG_FQDN_PORT : EIR signaling fqdn and port in the format FQDN:PORT (ocair-ingressgateway-sig.ocair:80)
14  //EIR_PROV_FQDN_PORT : EIR signaling fqdn and port in the format FQDN:PORT (ocair-ingressgateway-sig.ocair:80)
15  //EIR_CONF_FQDN_PORT : EIR nudr-config fqdn and port in the format FQDN:PORT (ocair-nudr-config.ocair:5001)
16  //STUB1_FQDN_PORT : Primary NRF Stub fqdn and port in the format FQDN:PORT (nrf-stub-service1.ocair:8080)
17  //STUB2_FQDN_PORT : Secondary NRF Stub fqdn and port in the format FQDN:PORT (nrf-stub-service2.ocair:8080)
18

```

☒ Use Groovy Sandbox

Save Apply

Note

Setting rerun count and the number of retries to check the pod status and segment details is given below:

- Rerun count can be set to any desired value by changing option 's' in the pipeline script.
- Number of retries to check if pod is down can be set by providing option 'o' in the pipeline script.
- Number of retries to check if pod is up can be set by providing option 'p' in the pipeline script.
- If ProvGw is deployed with segment details having UDR fqdn/IP:port, provide option 'i' as "IP". If ProvGw is deployed with segment details having only UDR fqdn/IP, provide option 'i' as "fqdn". If any value other than "IP" and "fqdn" is provided, ATS assumes that the segment details in ProvGw is in "IP" mode.
- Option 'u' shall take value '/<domain name>/prometheus/api/v1 for CNE version 1.9 or above or OSO version is 1.10.x or above. Run the below command from ATS pod to find the domain name, if the domain name is not known:

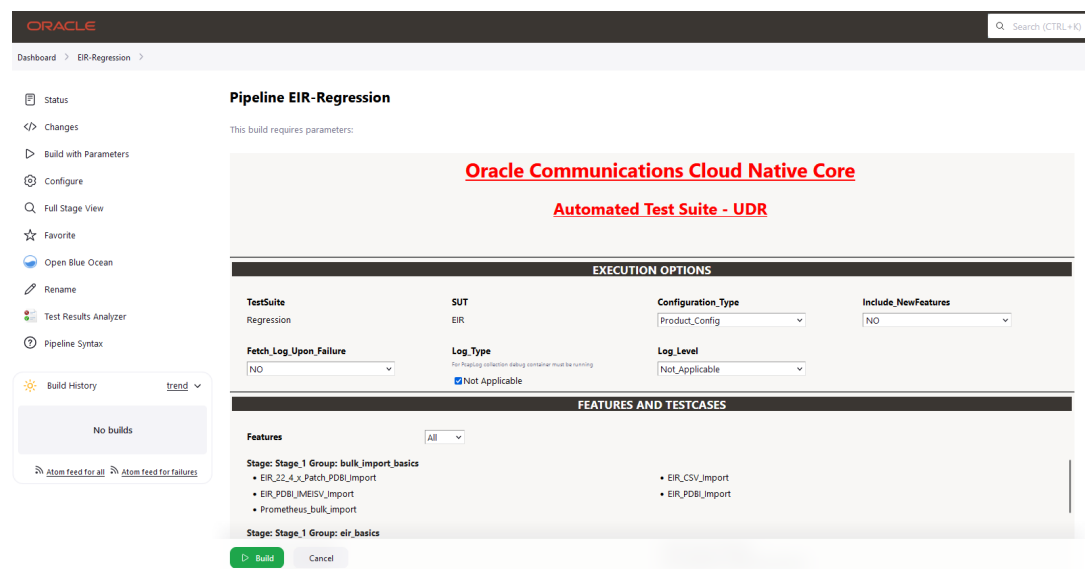
```
curl http://<prometheus fqdn>:<port>/
```

Example:

```
curl http://occne-kube-prom-stack-kube-prometheus.occne-infra/
<a href="/udr-cne/prometheus">Found</a>
```

- Click **Save**. The **Pipeline EIR-Regression** page appears.
- Click **Build with Parameters**. A page listing all the regression features and displays the total count of the features and displays the total count of the features, count of scenarios in each feature, name of each feature, and scenarios in each feature.

Figure 4-132 EIR Regression Build with Parameters



8. A sample consolidated automation suite test report, when re-run is set to 0 is as follows

Figure 4-133 EIR-Regression Consolidated Automation Suite Test Report

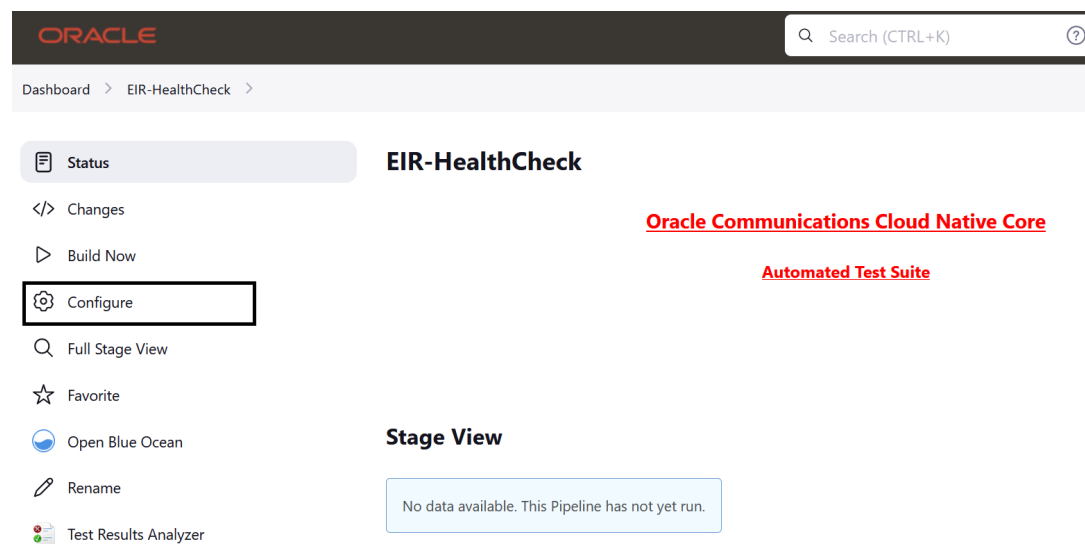
```
31 features passed, 0 failed, 0 skipped
262 scenarios passed, 0 failed, 0 skipped
6690 steps passed, 0 failed, 0 skipped, 0 undefined
Took 232m49.448s
```

4.8.8.3 Running EIR HealthCheck Pipeline

To run health check on EIR:

1. Navigate to **EIR-HealthCheck** pipeline and click **Configure**.

Figure 4-134 EIR HealthCheck Pipeline



- The **General** tab appears. The user must wait for the page to load completely.
- Click the **Advanced Project Options** tab. Scroll-down to reach the **Pipeline** configuration section as follows:

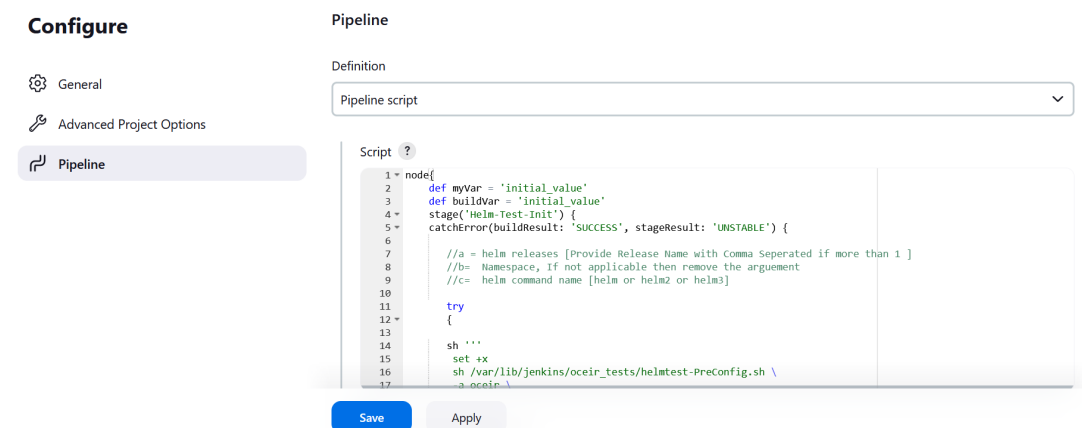
! Important

Make sure the page loads completely before you perform any action on it. Do not modify any configuration except the one discussed below.

Do not change any value other than **line numbers 16 and 17**. You can change the parameters marked as "a" "b" and "c" as per user requirement. The parameters details are provided as comments in **line number 7 and 9**. The parameters description is as follows:

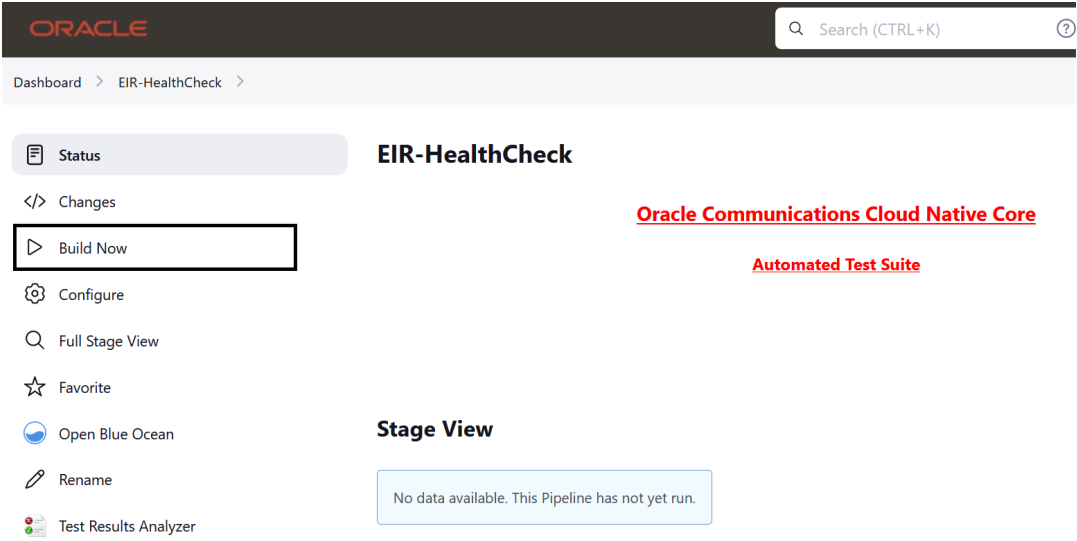
- a - helm releases [Provide Release Name with Comma Separated if more than 1]
- b - namespace of SUT.
- c - helm command alias (helm, helm2, or helm3)

Figure 4-135 EIR-HealthCheck Pipeline configuration



- Click **Save**. The **Pipeline EIR-HealthCheck** page appears.
- Click **Build Now**. This triggers health check for SUT.

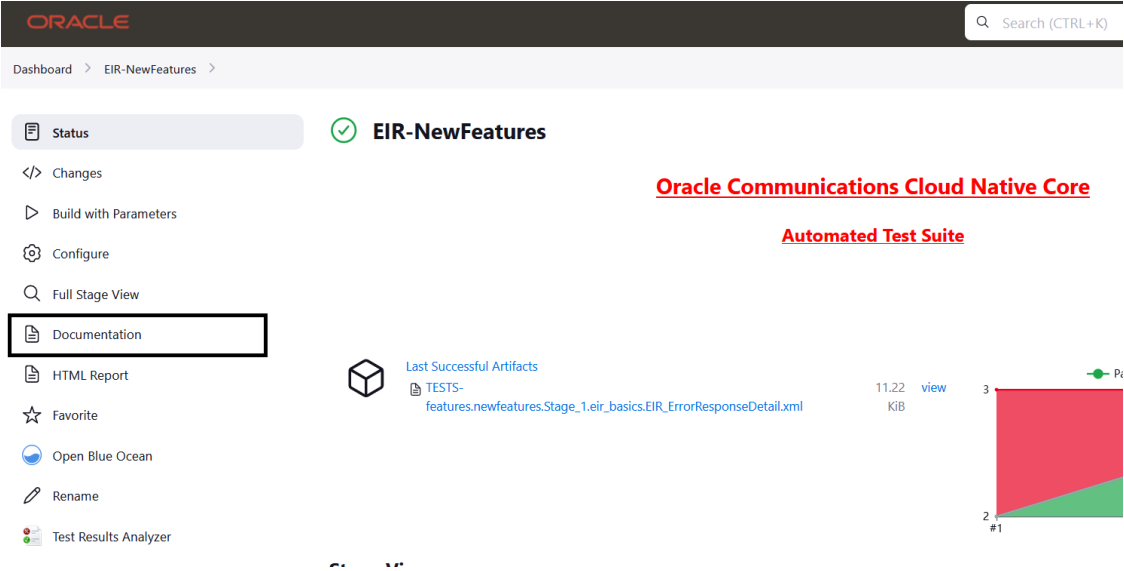
Figure 4-136 EIR-HealthCheck Build Now



4.8.8.4 EIR NewFeatures Documentation

To view the documentation of EIR-NewFeatures, click the **Documentation** link in the left navigation pane (present inside the build), as follows:

Figure 4-137 EIR-New Features Documentation



The page lists all the features. Click any feature to view description of that test case.
EIR new features list along with its description is as follows:

Table 4-20 EIR New Features List

Feature	Description
EIR_ErrorResponseDetail.feature	Validates the error detail in response received from EIR.

Note

The Documentation option appears only if you have run the EIR-NewFeatures pipeline test cases atleast once.

Based on the functionalities covered under **Documentation**, the **Build Requires Parameters** page displays test cases. To navigate back to the Pipeline EIR-NewFeatures page, click **Back to EIR-NewFeatures** link available on top left corner of the page.

4.8.8.5 EIR Regression Documentation

Click **Documentation** in the left navigation pane of the EIR Regression pipeline to view all the test cases provided.

Figure 4-138 EIR Regression Documentation

The screenshot shows the Oracle Cloud Native Core Automated Testing Suite (ATS) interface. The top navigation bar includes the Oracle logo and the text 'Dashboard > EIR-Regression > #1'. The main content area is divided into two sections. On the left, there is a sidebar with a list of navigation options: Status, Changes, Console Output, View as plain text, View Build Information, Parameters, Documentation (highlighted with a black border), HTML Report, Test Result, Open Blue Ocean, and Restart from Stage. On the right, there is a section titled 'Build #1 (Mar 8, 2024, 2:05:00 PM)' with a red 'X' icon. Below this title, there are three items: 'Build Artifacts', 'Started by user eiruser', and 'Test Result (3 failures)'. The 'Test Result' item is expanded, showing a list of test cases: 'features.regression.Stage_1.bulk_import_basics.Prc', 'features.regression.Stage_1.nrf-client.Prometheus_', and 'features.regression.Stage_2.sequential_execution.E'.

Click any feature to see the description of that test case.

