

Oracle® Communications

Cloud Native Core, Converged Policy Benchmarking Guide



Release 23.4.0
F91862-01
March 2024

ORACLE®

Copyright © 2023, 2023, 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	Purpose and Scope	1
1.2	References	1
2	Deployment Environment	
2.1	Deployed Components	1
2.2	Deployment Diagram	2
3	Benchmarking Policy Call Models	
3.1	PCRF Call Model 1	1
3.1.1	Test Scenario 1: PCRF Data Call Model on Four-Site GeoRedundant setup, with 7.5K Transaction Per Second (TPS) on each site and ASM disabled	1
3.1.1.1	Test Case and Setup Details	2
3.1.1.2	CPU Utilization	5
3.1.1.3	Results	6
3.1.2	Test Scenario 2: PCRF Voice Call Model on Two-Sites of Four-Site GeoRedundant setup, with 15K Transaction Per Second (TPS) on each site and ASM disabled	6
3.1.2.1	Test Case and Setup Details	6
3.1.2.2	CPU Utilization	9
3.1.2.3	Results	10
3.2	PCF Call Model 2	10
3.2.1	Test Scenario: PCF Call Model on Two-Site GeoRedundant setup, with 15K TPS each for AM/UE and ASM enabled.	10
3.2.1.1	Test Case and Setup Details	10
3.2.1.2	CPU Utilization	13
3.2.1.3	Results	14
3.2.2	Test Scenario: PCF AM/UE Call Model on Two-Site GeoRedundant setup, with each site handling 25K TPS traffic and ASM enabled	15
3.2.2.1	Test Case and Setup Details	15
3.2.2.2	CPU Utilization	21
3.2.2.3	Results	22

3.2.3	Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with each site handling 43K TPS traffic and ASM Enabled	22
3.2.3.1	Test Case and Setup Details	22
3.2.3.2	CPU Utilization	27
3.2.3.3	Results	28
3.2.4	Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with each site handling 30K TPS traffic and ASM Enabled	28
3.2.4.1	Test Case and Setup Details	29
3.2.4.2	CPU Utilization	35
3.2.4.3	Results	36
3.2.5	Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with 41K TPS Traffic on Site-1 and ASM Enabled	36
3.2.5.1	Test Case and Setup Details	36
3.2.5.2	CPU Utilization	41
3.2.5.3	Results	41
3.2.6	Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Each Site Handling 30K TPS Traffic and ASM Enabled	42
3.2.6.1	Test Case and Setup Details	42
3.2.6.2	CPU Utilization	47
3.2.6.3	Results	49
3.2.7	Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Single-Site Handling 60K TPS Traffic and ASM Enabled	50
3.2.7.1	Test Case and Setup Details	50
3.2.7.2	CPU Utilization	56
3.2.7.3	Results	58
3.2.8	Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Single-Site Handling 75K TPS Traffic and ASM Enabled	58
3.2.8.1	Test Case and Setup Details	59
3.2.8.2	CPU Utilization	65
3.2.8.3	Results	66
3.3	Policy Call Model 3	66
3.3.1	Test Scenario: Policy Voice Call Model on Four-Site Georedundant Setup, with 7.5K TPS Traffic on Each Site and ASM Disabled	66
3.3.1.1	Test Case and Setup Details	66
3.3.1.2	CPU Utilization	69
3.3.1.3	Results	70
3.3.2	Test Scenario: Policy Voice Call Model on Four-Site Georedundant Setup, with 15K TPS Traffic on Two Sites and No Traffic on Other Two Sites	70
3.3.2.1	Test Case and Setup Details	71
3.3.2.2	CPU Utilization	73
3.3.2.3	Results	75
3.4	Policy Call Model 4	75
3.4.1	Test Scenario: Policy Call Model on Four-Site Georedundant Setup, with 7.5K TPS Traffic on Each Site and ASM Disabled	75

3.4.1.1	Test Case and Setup Details	75
3.4.1.2	CPU Utilization	78
3.4.1.3	Results	79
3.4.2	Test Scenario: Policy Call Model on Four-Site Georedundant Setup, with 15K TPS Traffic on Two Sites and No Traffic on Other Two Sites	79
3.4.2.1	Test Case and Setup Details	80
3.4.2.2	CPU Utilization	83
3.4.2.3	Results	84
3.4.3	Test Scenario: Policy Call Model on Two-Site Georedundant Setup, with 15K TPS Traffic on Two Sites	84
3.4.3.1	Test Case and Setup Details	84
3.4.3.2	CPU Utilization	87
3.4.3.3	Results	88
3.5	PCF Call Model 5	88
3.5.1	Test Scenario: PCF Call Model on Single-Site Setup, Handling 30K TPS Traffic with Binding Feature Enabled	88
3.5.1.1	Test Case and Setup Details	88
3.5.1.2	CPU Utilization	92
3.5.1.3	Results	93
3.5.2	Test Scenario: PCF Call Model on Single-Site Setup, Handling 30K TPS Traffic with Binding Feature Disabled	93
3.5.2.1	Test Case and Setup Details	93
3.5.2.2	CPU Utilization	97
3.5.2.3	Results	98
3.6	PCF Call Model 6	98
3.6.1	Test Scenario: 10K TPS Diameter Ingress Gateway and 17K TPS Egress Gateway TPS Traffic with Usage Monitoring Enabled	98
3.6.1.1	Test Case and Setup Details	99
3.6.1.2	CPU Utilization	102
3.6.1.3	Results	103

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 in the following list 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 provides information about the acronyms and the terminology used in the document.

Table Acronyms and Terminologies

Acronym	Description
ASM	Aspen Service Mesh
AMF	Access and Mobility Management Function
AAR	Authorization Authentication Request
BSF	Oracle Communications Cloud Native Core, Binding Support Function
CPS	Call Per Second
CHF	Charging Function
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment
CPU	Central Processing Unit
DNN	Data Network Name
HTTP	Hypertext Transfer Protocol
LDAP	Lightweight Directory Access Protocol
MPS	Messages Per Second
NF	Network Function
NRF	Oracle Communications Cloud Native Core, Network Repository Function
OCS	Online Charging System
PER	Policy Event Record
PCF	Oracle Communications Cloud Native Core, Policy Control Function
PCRF	Policy and Charging Rules Function
PV	Persistent Volume
RAM	Random Access Memory
RAR	Re-Authorization Request
SAL	Subscriber Activity Log
SSV	Subscriber State Variable
STR	Session Termination Request
SM	Session Management
Sy	Diameter Sy reference point
TPS	Transactions Per Second
UDR	Oracle Communications Cloud Native Core, Unified Data Repository
vCNE	Virtual Cloud Native Environment

What's New in this Guide

This section introduces the documentation updates for Release 23.4.x in *Oracle Communications Cloud Native Core, Converged Policy Benchmarking Guide*.

Release 23.4.0 - F91862-01, March 2024

Updated the [Deployed Components](#) section.

Updated the document to include following test scenarios:

- [Test Scenario: PCF AM/UE Call Model on Two-Site GeoRedundant setup, with each site handling 25K TPS traffic and ASM enabled](#) section.
- [Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with each site handling 30K TPS traffic and ASM Enabled](#)

1

Introduction

Oracle Communications Cloud Native Core, Converged Policy (Policy) is a key component of the 5G Service Based Architecture (SBA). It is a cloud native solution consisting of both, a 4G Policy and Charging Rules Function (PCRF) and a 5G Policy Control Function (PCF) as a unified framework. It provides a flexible, secure, and scalable policy designing solution.

Policy interacts with other Network Functions (NF) through Network Repository Function (NRF) to provide a unified communication platform for the NFs to interact with each other. It helps operators to design, test, and deploy different network policies supporting 5G deployments. Policy solution supports deployments into cloud native environment, including containers on bare metal managed by Kubernetes or VMs managed by OpenStack.

Note

The performance and capacity of the Policy system may vary based on the Call model, Feature/Interface configurations, underlying CNE and hardware environment, including but not limited to the complexity of deployed policies, policy table size, object expression, and custom json usage in policy design.

For more information about Policy architecture, see *Oracle Communications Cloud Native Core, Converged Policy User Guide*.

1.1 Purpose and Scope

This document is designed to help operators in measuring the performance and capacity of Policy, Policy microservices, and deployment environment setup software such as Cloud Native Environment (CNE) and cnDBTier.

It is recommended that Policy is run through a benchmark on the target cloud native infrastructure to determine the capacity and performance in the target infrastructure. This information can be used to adjust the initial deployment resources for Policy. These recommendations are just guidelines, since the actual performance of the Policy can vary significantly based on the details of the infrastructure.

1.2 References

- *Oracle Communications Cloud Native Core, Converged Policy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Converged Policy User Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, cnDBTier Installation, Upgrade, and Fault Recovery Guide*

2

Deployment Environment

This section provides information about the cloud native platform infrastructure details for deploying Oracle Communications Cloud Native Core, Converged Policy.

Note

The performance and capacity of the Policy system may vary based on the Call model, Feature/Interface configurations, underlying CNE and hardware environment, including but not limited to the complexity of deployed policies, policy table size, object expression, and custom json usage in policy design.

2.1 Deployed Components

This section provides details about the deployed components.

Deployment Platform

Oracle Communications Cloud Native Environment (CNE) 23.3.3 and BareMetal can be used for performing benchmark tests.

Table 2-1 Observability Services

Service Name	Version
OpenSearch	2.3.0
Fluentd	1.16.2
Prometheus	2.44.0
Grafana	9.5.3
Jaeger	1.45.0

Cloud Native Orchestrator

Kubernetes 1.27.x is used to manage application pods across the cluster.

cnDBTier

cnDBTier 23.2.2 is used for performing benchmark tests.

Policy Infrastructure Details

Infrastructure used for benchmarking Policy performance run is described in this section.

Table 2-2 Hardware Details

Hardware	Details
Environment	BareMetal
Server	Oracle Server X8-2

Table 2-2 (Cont.) Hardware Details

Hardware	Details
Model	Intel(R) Xeon(R) Platinum 8260
Clock Speed	2.400 GHz
Total Cores	24
Memory Size	32 GB
Type	DDR4 SDRAM
Installed DIMMs	18
Maximum DIMMs	24
Installed Memory	576 GB

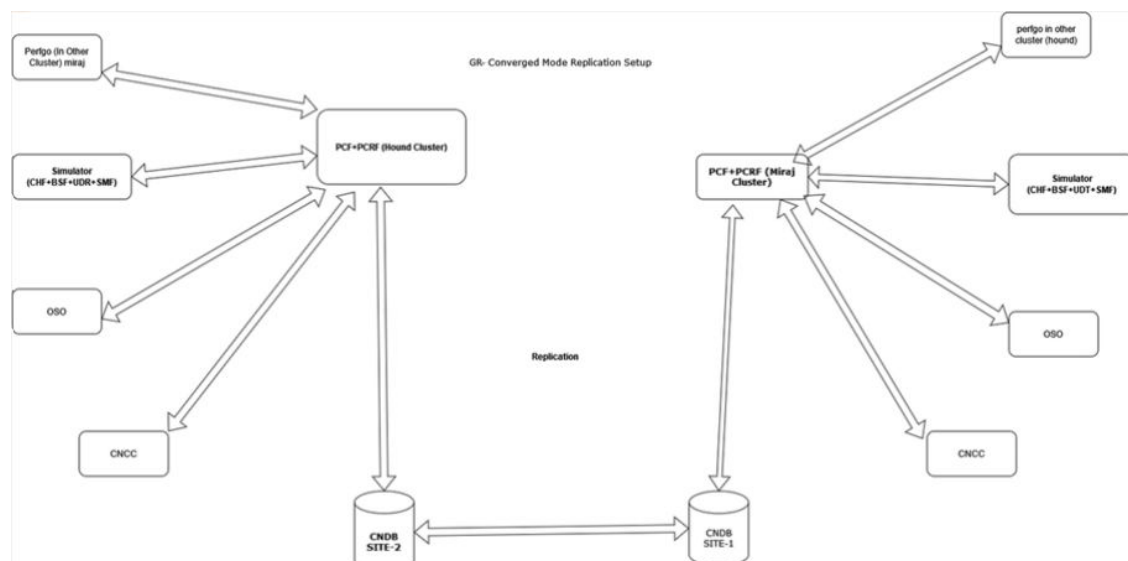
Table 2-3 Software Details

Applications	Version
PCF	23.4.0
cnDBTier	23.2.2
OSO	23.3.2
CNE	23.3.3

For more information about Policy Installation details, see *Oracle Communications Cloud Native Core, Converged Policy Installation, Upgrade, and Fault Recovery Guide*.

2.2 Deployment Diagram

The following diagram describes the Policy application deployed in converged mode on a Two-Site Georedundant replication setup.

Figure 2-1 Policy Deployment in Two-Site GR Setup

3

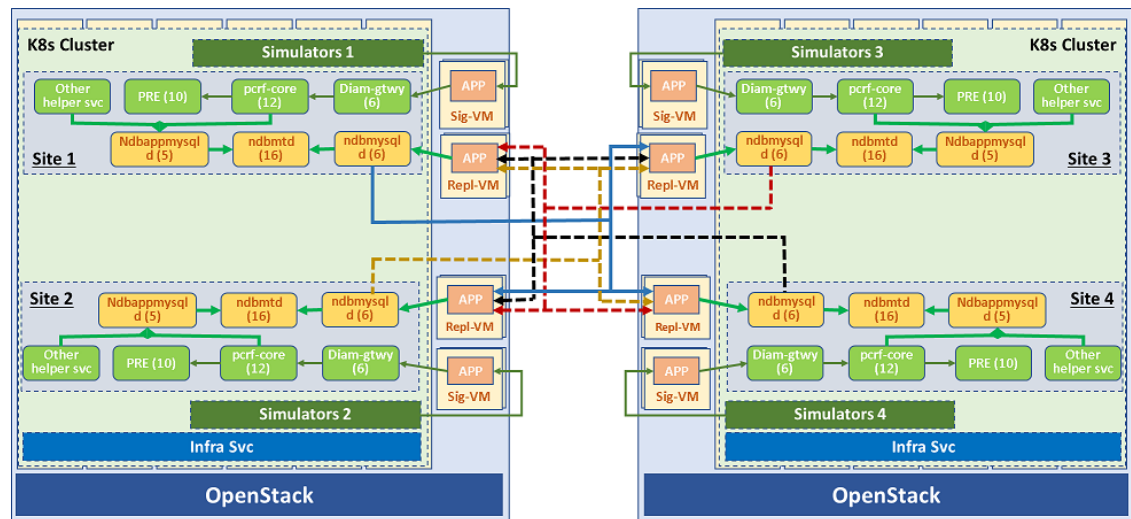
Benchmarking Policy Call Models

This section describes different Policy call models and the performance test scenarios which were run using these call model.

3.1 PCRF Call Model 1

The following diagram describes the architecture for a multisite PCRF deployment.

Figure 3-1 PCRF 4 Site GR Deployment Architecture



To test this PCRF call model, the Policy application is deployed in converged mode on a four-site georedundant site. The cnDBTier database and PCRF application are replicated on all the four-sites. The database replication is used to perform data synchronization between databases over the replication channels.

3.1.1 Test Scenario 1: PCRF Data Call Model on Four-Site GeoRedundant setup, with 7.5K Transaction Per Second (TPS) on each site and ASM disabled

This test run benchmarks the performance and capacity of PCRF data call model that is deployed in converged mode on a four-site georedundant setup. Each site in the setup handles an incoming traffic of 7.5K TPS. Aspen Service Mesh (ASM) is disabled.

3.1.1.1 Test Case and Setup Details

Table 3-1 Test Case Parmeters

Parameters	Values
Call Rate	30K TPS (7.5K TPS on each site)
Execution Time	12 Hours
ASM	Disable

Table 3-2 Call Model Data

Messages	Total CPS Instance-1	sy Traffic	Ldap Traffic	Total TPS
CCR-I	320	320	320	960
CCR-U	320	0	0	320
CCR-T	320	320	0	640
Total Messages	960	640	320	1920

Table 3-3 PCRF Configurations

Service Name	Status
Binding Service	Disable
Policy Event Record (PER)	Disable
Subscriber Activity Log (SAL)	Enable
LDAP	Enable
Online Charging System (OCS)	Enable

Table 3-4 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Disable
N36 UDR subscription (N7/N15-Nudr)	Disable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Disable
BSF (N7-Nbsf)	Disable
AMF on demand nrf discovery	Disable
LDAP (Gx-LDAP)	Enable
Sy (PCF N7-Sy)	Enable

Table 3-5 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	Enable
Sd (Gx-Sd)	Disable
Gx UDR query (Gx-Nudr)	Disable

Table 3-5 (Cont.) PCRF Interfaces

Feature Name	Status
Gx UDR subscription (Gx-Nudr	Disable
CHF enabled (AM)	Disable
Usage Monitoring (Gx)	Disable
Subscriber HTTP Notifier (Gx)	Disable

Table 3-6 Configuring cnDBTier Helm Parameters

Helm Parameter	New Value
ndb_batch_size	2G
TimeBetweenEpochs	100
NoOfFragmentLogFiles	50
FragmentLogFileSize	256M
RedoBuffer	1024M
ndbappmysqld Pods Memory	19/20 Gi
ndbmttd pods CPU	8/8
ndb_report_thresh_binlog_epoch_slip	50
ndb_eventbuffer_max_alloc	19G
ndb_log_update_minimal	1
ndbmysqld Pods Memory	25/25 Gi
replicationskiperrors	enable: true
replica_skip_errors	'1007,1008,1050,1051,1022'
numOfEmptyApiSlots	4

Table 3-7 Policy Microservices Resource

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ocpcf-appinfo	1	1	0.5	1	1
ocpcf-oc-binding	5	6	1	8	15
ocpcf-oc-diam-connector	3	4	1	2	8
ocpcf-oc-diam-gateway	3	4	1	2	7
ocpcf-occnf-config-server	2	4	0.5	2	1
ocpcf-occnf-egress-gateway	3	4	4	6	2
ocpcf-ocpm-ldap-gateway	3	4	1	2	10
ocpcf-occnf-ingress-gateway	3	4	4	6	2

Table 3-7 (Cont.) Policy Microservices Resource

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ocpcf-occnp-nrf-client-nfdiscovery	3	4	0.5	2	2
ocpcf-occnp-nrf-client-nfmanagement	1	1	1	1	2
ocpcf-ocpm-audit-service	1	2	1	1	1
ocpcf-ocpm-cm-service	2	4	0.5	2	1
ocpcf-ocpm-policyds	5	6	1	4	25
ocpcf-ocpm-pre	5	5	0.5	4	25
ocpcf-ocpm-queryservice	1	2	1	1	1
ocpcf-pcf-smsservice	7	8	1	4	2
ocpcf-pcrf-core	7	8	8	8	30
ocpcf-performance	1	1	0.5	1	2

Note

Min Replica = Max Replica

Table 3-8 cnDBTier Services Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ndbappmysqld	8	8	19	20	5
ndbmcmd	2	2	9	11	2
ndbmtl	8	8	73	83	8
ndbmysqld	4	4	19	20	12

Note

Min Replica = Max Replica

3.1.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the Pod).

Table 3-9 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y)- Site 2	CPU(X/Y) - Site 3	CPU(X/Y) - Site 4
ocpcf-alternate-route	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-appinfo	1%/80%	2%/80%	2%/80%	3%/80%
ocpcf-occnf-config-server	10%/80%	11%/80%	12%/80%	12%/80%
ocpcf-oc-diam-connector	10%/40%	11%/40%	10%/40%	10%/40%
ocpcf-occnf-egress-gateway	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-occnf-ingress-gateway	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-ocpm-ldap-gateway	4%/60%	4%/60%	5%/60%	4%/60%
ocpcf-occnf-nrf-client-nfdiscovery	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-occnf-nrf-client-nfmanagement	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-oc-binding	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-occnf-chf-connector	0%/50%	0%/50%	0%/50%	0%/50%
ocpcf-occnf-udr-connector	0%/50%	0%/50%	0%/50%	0%/50%
ocpcf-ocpm-audit-service	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-ocpm-policyds	11%/60%	11%/60%	11%/60%	11%/60%
ocpcf-ocpm-soapconnector	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-ocpm-pre	13%/80%	13%/80%	13%/80%	13%/80%
ocpcf-pcf-smsservice	0%/50%	0%/50%	0%/50%	0%/50%
ocpcf-pcrf-core	7%/40%	7%/40%	7%/40%	7%/40%
ocpcf-ocpm-queryservice	0%/80%	0%/80%	0%/80%	0%/80%

Table 3-10 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2	CPU (X/Y) - Site 3	CPU (X/Y) - Site 4
ndbappmysqld	35%/80%	36%/80%	35%/80%	35%/80%
ndbmcmd	1%/80%	1%/80%	0%/80%	0%/80%

Table 3-10 (Cont.) cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2	CPU (X/Y) - Site 3	CPU (X/Y) - Site 4
ndbmttd	15%/80%	15%/80%	18%/80%	17%/80%
ndbmysqld	5%/80%	5%/80%	5%/80%	5%/80%

3.1.1.3 Results

The following table provides observation data for the performance test that can be used for benchmark testing:

Table 3-11 Result and Observations

Parameter	Values
Test Duration	12 Hours
TPS Achieved	30K TPS (7.5KTPS on each site)

It was observed that on a four-site GR setup, handling an incoming traffic of 7.5K TPS on each site, the call model was working successfully without any replication delay and traffic drop.

3.1.2 Test Scenario 2: PCRF Voice Call Model on Two-Sites of Four-Site GeoRedundant setup, with 15K Transaction Per Second (TPS) on each site and ASM disabled

This test run benchmarks the performance and capacity of PCRF voice call model that is deployed in converged mode on a two-site of a four-site georedundant setup. Each site in the setup handles an incoming traffic of 15K TPS, and with Aspen Service Mesh (ASM) disabled.

3.1.2.1 Test Case and Setup Details

Table 3-12 Test Case Parameters

Parameters	Values
Call Rate	30K TPS (15K TPS on each site)
Execution Time	10 Hours
ASM	Disable

Table 3-13 Call Model Data

Command	Messages per call
CCRI (Single APN)	9.08%
CCRU (Single APN)	18.18%
CCRT (Single APN)	9.09 %
Gx RAR	18.18%
AARI	9.09 %
AARU	9.09 %

Table 3-13 (Cont.) Call Model Data

Command	Messages per call
Rx RAR	18.18%
STR	9.09%

Table 3-14 PCRF Configurations

Service Name	Status
Binding Service	Enable
Policy Event Record (PER)	Disable
Subscriber Activity Logging (SAL)	Enable
LDAP	Disable
Online Charging System (OCS)	Disable

Table 3-15 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Disable
N36 UDR subscription (N7/N15-Nudr)	Disable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Disable
BSF (N7-Nbsf)	Disable
AMF on demand nrf discovery	Disable
LDAP (Gx-LDAP)	Disable
Sy (PCF N7-Sy)	Disable

Table 3-16 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	Disable
Sd (Gx-Sd)	Disable
Gx UDR query (Gx-Nudr)	Disable
Gx UDR subscription (Gx-Nudr)	Disable
CHF enabled (AM)	Disable
Usage Monitoring (Gx)	Disable
Subscriber HTTP Notifier (Gx)	Disable

Table 3-17 Configuring cnDBTier Helm Parameters

Helm Parameter	Value
ndb_batch_size	2G
TimeBetweenEpochs	100
NoOfFragmentLogFiles	50
FragmentLogFileSize	256M

Table 3-17 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value
RedoBuffer	1024M
ndbappmysqld Pods Memory	19/20 Gi
ndbmtld pods CPU	8/8
ndb_report_thresh_binlog_epoch_slip	50
ndb_eventbuffer_max_alloc	19G
ndb_log_update_minimal	1
ndbmysqld Pods Memory	25/25 Gi
replicationskiperrors	enable: true
replica_skip_errors	'1007,1008,1050,1051,1022'
numOfEmptyApiSlots	4

Table 3-18 Policy Microservices Resource

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ocpcf-appinfo	1	1	0.5	1	1
ocpcf-oc-binding	5	6	1	8	18
ocpcf-oc-diam-connector	3	4	1	2	8
ocpcf-oc-diam-gateway	3	4	1	2	9
ocpcf-occnf-config-server	2	4	0.5	2	2
ocpcf-occnf-egress-gateway	3	4	4	6	1
ocpcf-ocpm-ldap-gateway	3	4	1	2	0
ocpcf-occnf-ingress-gateway	3	4	4	6	2
ocpcf-occnf-nrf-client-nfdiscovery	3	4	0.5	2	1
ocpcf-occnf-nrf-client-nfmanagement	1	1	1	1	1
ocpcf-ocpm-audit-service	1	2	1	1	1
ocpcf-ocpm-cm-service	2	4	0.5	2	1
ocpcf-ocpm-policyds	5	6	1	4	2
ocpcf-ocpm-pre	5	5	0.5	4	15
ocpcf-ocpm-queryservice	1	2	1	1	1

Table 3-18 (Cont.) Policy Microservices Resource

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ocpcf-pcf-smsservice	7	8	1	4	2
ocpcf-pcrf-core	7	8	8	8	24
ocpcf-performance	1	1	0.5	1	2

Note

Min Replica = Max Replica

Table 3-19 cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ndbappmysqld	8	8	19	20	5
ndbmcmd	2	2	9	11	3
ndbmtd	8	8	73	83	8
ndbmysqld	4	4	19	20	6

Note

Min Replica = Max Replica

3.1.2.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-20 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ocpcf-appinfo	2%/80%	1%/80%
ocpcf-occn-p-config-server	8%/80%	8%/80%
ocpcf-oc-diam-connector	0%/40%	0%/40%
ocpcf-occn-p-egress-gateway	0%/80%	0%/80%
ocpcf-occn-p-ingress-gateway	0%/80%	1%/80%
ocpcf-occn-p-nrf-client-nfdiscovery	0%/80%	0%/80%
ocpcf-occn-p-nrf-client-nfmanagement	0%/80%	0%/80%
ocpcf-oc-binding	12%/60%	0%/60%

Table 3-20 (Cont.) Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ocpcf-ocpm-audit-service	0%/60%	0%/60%
ocpcf-ocpm-policyds	0%/60%	0%/60%
ocpcf-ocpm-pre	13%/80%	0%/80%
ocpcf-pcf-smsservice	0%/50%	0%/50%
ocpcf-pcrf-core	25%/40%	0%/40%
ocpcf-ocpm-queryservice	0%/80%	0%/80%

Table 3-21 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ndbappmysqld	75%/80%	76%/80%
ndbmngmd	0%/80%	0%/80%
ndbmtl	19%/80%	6%/80%
ndbmysqld	8%/80%	3%/80%

3.1.2.3 Results

In this four site geo-redundant setup, it was observed that,

- each of the two sites handles traffic of 15k TPS successfully, and
- in the event of two site failure, the system failover to the two redundant sites quickly.

3.2 PCF Call Model 2

3.2.1 Test Scenario: PCF Call Model on Two-Site GeoRedundant setup, with 15K TPS each for AM/UE and ASM enabled.

This test run benchmarks the performance and capacity of Policy data call model that is deployed in PCF mode. The PCF application handles an incoming traffic of 30K TPS, with 15K TPS each for AM and UE services. For this setup Aspen Service Mesh (ASM) was enabled.

3.2.1.1 Test Case and Setup Details

Table 3-22 Test Case Parmeters

Parameters	Values
Call Rate	30K TPS on Single site
Execution Time	17 Hours
ASM	Enable
Traffic Ratio	1:0:1 (AM/UE Create: AM/UE Update: AM/UE delete)
Active Subscribers	~10000000

Table 3-23 Call Model

Service Name	AM Service			UE Service			Total MPS	Total TPS
	Ingress	Egress	Total MPS	Ingress	Egress	Total MPS		
Ingress	3600	3600	7200	3600	3600	7200	14400	7200
PRE	3600	0	3600	3600	0	3600	7200	3600
PDS	9000	9000	18000	8100	6300	14400	34200	17100
Egress	9900	9900	19800	13500	13500	27000	46800	23,400
Nrf Discovery	1800	1800	3600	1800	1800	3600	7200	3600
UDR Connector	6300	8100	14400	6300	6300	12600	27000	13500
CHF Connector	3600	3600	7200	0	0	0	7200	3600
AM	3600	18900	22500	0	0	0	22500	11250
UE	0	0	0	3600	20700	24300	24300	12150
Bulwark	7200	0	7200	7200	0	7200	14400	7200

Table 3-24 PCF Configuration

Service Name	Status
Bulwark Service	Enable
Binding Service	Disable
Subscriber State Variable (SSV)	Enable
Validate_user	Disable
Alternate Route Service	Disable
Audit Service	Enable
Binlog	Enable

Table 3-25 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Disable
BSF (N7-Nbsf)	Disable
AMF on demand nrf discovery	Disable
LDAP (Gx-LDAP)	Disable
Sy (PCF N7-Sy)	Enable

Table 3-26 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	Disable
Sd (Gx-Sd)	Disable
Gx UDR query (Gx-Nudr)	Disable
Gx UDR subscription (Gx-Nudr)	Disable
CHF enabled (AM)	Disable
Usage Monitoring (Gx)	Disable
Subscriber HTTP Notifier (Gx)	Disable

Table 3-27 Configuring cnDBTier Helm Parameters

Helm Parameter	Value
restartSQLNodesIfBinlogThreadStalled	true
binlog_cache_size	65536
ndbsql node memory	54Gi
NoOfFragmentLogFiles	96
ndb_allow_copying_alter_table	1

Table 3-28 Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
Appinfo	1	1	0.5	1	2
Audit Service	1	2	1	1	2
CM Service	2	4	0.5	2	2
Config Service	2	4	0.5	2	2
Egress Gateway	4	4	4	6	13
Ingress Gateway	4	4	4	6	4
Nrf Client Management	1	1	1	1	2
Diameter Gateway	4	4	1	2	0
Diameter Connector	4	4	1	2	0
AM Service	8	8	1	4	9
UE Service	8	8	1	4	11
Nrf Client Discovery	4	4	0.5	2	4
Query Service	1	2	1	1	2
PCRF Core Service	8	8	8	8	0
Performance	1	1	0.5	1	2
PRE Service	4	4	0.5	2	6

Table 3-28 (Cont.) Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
SM Service	8	8	1	4	0
PDS	6	6	1	4	17
UDR Connector	6	6	1	4	7
CHF Connector	6	6	1	4	2
LDAP Gateway Service	3	4	1	2	0
Binding Service	5	6	1	8	0
SOAP Connector	2	4	4	4	0
Alternate Route Service	2	2	2	4	4
Bulwark Service	8	8	1	4	3

Note

Min Replica = Max Replica

Table 3-29 cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replica
ndbappmysql	15	15	18	18	6
ndbmcmd	3	3	10	10	2
ndbmt	12	12	96	96	12
ndbmysql	4	4	54	54	2

Note

Min Replica = Max Replica

3.2.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-30 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site1
ocpcf-alternate-route	0%/80%

Table 3-30 (Cont.) Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site1
ocpcf-appinfo	0%/80%
ocpcf-bulwark	0%/60%
ocpcf-occnf-config-server	9%/80%
ocpcf-occnf-egress-gateway	46%/80%
ocpcf-occnf-ingress-gateway	38%/80%
ocpcf-occnf-nrf-client-nfdiscovery	38%/80%
ocpcf-occnf-nrf-client-nfmanagement	15%/80%
ocpcf-oc-binding	0%/60%
ocpcf-occnf-chf-connector	0%/50%
ocpcf-occnf-udr-connector	46%/50%
ocpcf-ocpm-audit-service	0%/60%
ocpcf-ocpm-policyds	32%/60%
ocpcf-ocpm-pre	18%/80%
ocpcf-pcf-amservice	21%/30%
ocpcf-pcf-ueservice	33%/30%
ocpcf-ocpm-queryservice	0%/80%

The following table provides

Table 3-31 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site1
ndbappmysql	31%/80%
ndbmcmd	0%/80%
ndbmtd	43%/80%
ndbmysql	9%/80%

3.2.1.3 Results

Table 3-32 Latency Observations

NF	Procedure	NF Processing Time - (Average/50%) ms	NF Processing Time - (99%) ms
AM-PCF	AM-Create (simulator)	56.2	47.6
	AM-Delete (simulator)	50.2	44.6
UE-PCF	AM-Create (simulator)	78.6	63.3
	AM-Delete (simulator)	7.6	6.3

Latency Observations for Policy Services:

Table 3-33 Latency Observations for Policy Services:

Services	Average Latency (ms)
Ingress	45.6
PDS	26.9
UDR	7.60
NrfClient Discovery - OnDemand	6.39
Egress	0.914

- Able to achieve 30K TPS with AM (15K) and UE (15K) with constant approximate run of 17 Hours.
- Latency was constant through out the call model run, with
 - approximate of 46ms for Ingress, and
 - approximate of <=20ms for rest of the PCF services

3.2.2 Test Scenario: PCF AM/UE Call Model on Two-Site GeoRedundant setup, with each site handling 25K TPS traffic and ASM enabled

This test run benchmarks the performance and capacity of Policy AM/UE data call model that is deployed in PCF mode. The PCF application handles a total (Ingress + Egress) traffic of 50K TPS, with each site handling a traffic of 25K TPS. For this setup Aspen Service Mesh (ASM) was enabled.

In this test setup, the Georedundant (GR) mode was enabled in cnDBTier and it was configured for 3 channel replication.

3.2.2.1 Test Case and Setup Details

Table 3-34 Test Case Parmeters

Parameters	Values
Call Rate (Ingress + Egress)	50K TPS on Single site
Execution Time	94 Hours
ASM	Enable
Traffic Ratio	1:0:1 (AM/UE Create: AM/UE Update: AM/UE delete)
Active Subscribers	12591141

Table 3-35 TPS Distribution

TPS Distribution	Site1	Site2
AM Ingress	6.12K	0
AM Egress	18.88K	0
UE Ingress	6.12K	0
UE Egress	18.88K	0
Total TPS	50K	0

Table 3-36 Call Model

Service Name	AM Service			UE Service			Total MPS	Total TPS
	Ingress	Egress	Total MPS	Ingress	Egress	Total MPS		
Ingress	6250	6250	12500	6250	6250	12500	25000	12500
PRE	6250	0	6250	6250	0	6250	12500	6250
PDS	9375	9375	18750	9375	9375	18750	37500	18750
Egress	12500	12500	25000	25000	25000	50000	75000	37500
Nrf Discovery	3125	3125	6250	6250	6250	12500	18750	9375
UDR Connector	9375	12500	21875	9375	12500	21875	43750	21875
CHF Connector	0	0	0	0	0	0	0	0
AM	6250	15625	21875	0	0	0	21875	10937.5
UE	0	0	0	6250	28125	34375	34375	17187.5
Bulwark	0	0	0	0	0	0	0	0

Table 3-37 PCF Configuration

Service Name	Status
Bulwark Service	Disable
Binding Service	NA
Subscriber State Variable (SSV)	Enable
Validate_user	Disable
Alternate Route Service	Disable
Audit Service	Enable
Binlog	Enable

Table 3-38 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Enable
CHF (SM-Nchf)	Disable
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	Enable
LDAP (Gx-LDAP)	Disable
Sy (PCF N7-Sy)	Disable

Table 3-39 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	Enable
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Table 3-40 Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
restartSQLNodesIfBinlogThreadStalled	true	<pre>db-monitor-svc: restartSQLNodesIfBinlogThreadStalled: true</pre>
binlog_cache_size	10485760	<pre>additionalndbconfigurations: mysqld: binlog_cache_size: '10485760'</pre>
ConnectCheckIntervalDelay	500	<pre>additionalndbconfigurations: ndb: ConnectCheckIntervalDelay: 500</pre>

Table 3-40 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
NoOfFragmentLogFiles	32	<pre># values for configuration files, cnf ndbconfigurations: ndb: NoOfFragmentLogFiles: 32</pre>
NoOfFragmentLogParts	6	<pre>ndbconfigurations: ndb: NoOfFragmentLogParts: 6</pre>
MaxNoOfExecutionThreads	14	<pre>ndbconfigurations: ndb: MaxNoOfExecutionThreads: 14</pre>
FragmentLogFileSize	128M	<pre>additionalndbconfigurations: ndb: FragmentLogFileSize: 128M</pre>
binlogthreadstore.capacity	5	<pre>db-monitor-svc: binlogthreadstore: capacity: 5</pre>

Table 3-40 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
ndb_allow_copying_alter_table	ON	<pre> additionalndbconfigurations: mysqld: # use replica_skip_errors as slave-skip-errors/ slave_skip_errors is deprecated ndb_allow_copying_alter_ table: 'ON' </pre>

Note

The cnDBTier customized parameters values remains same for both site1 and site2.

Table 3-41 Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Isito CPU	Request/ Limit Isito Memory
Appinfo	1	1	0.5	1	2	2	2	2
Audit Service	2	2	4	4	2	2	2	2
CM Service	2	4	0.5	2	2	2	2	2
Config Service	4	4	0.5	2	2	2	2	2
Egress Gateway	4	4	6	6	2	27	2	2
Ingress Gateway	5	5	6	6	2	8	2.5	2
Nrf Client Management	1	1	1	1	2	2	2	2
Diameter Gateway	4	4	1	2	0	0	2	2
Diameter Connector	4	4	1	2	0	0	2	2
AM Service	8	8	1	4	2	6	2	2

Table 3-41 (Cont.) Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Istio CPU	Request/ Limit Istio Memory
UE Service	8	8	1	4	2	16	2	2
Nrf Client Discovery	4	4	0.5	2	2	7	2	2
Query Service	1	2	1	1	2	2	2	2
PCRF Core Service	8	8	8	8	0	0	2	2
Performance	1	1	0.5	1	2	2	2	2
PRE Service	4	4	4	4	2	4	1.5	2
SM Service	7	7	10	10	0	0	2.5	2
PDS	7	7	8	8	2	22	2.5	4
UDR Connector	6	6	4	4	2	14	2	2
CHF Connector	6	6	4	4	0	0	2	2
LDAP Gateway Service	3	4	1	2	0	0	2	2
Binding Service	6	6	8	8	2	0	2.5	2
SOAP Connector	2	4	4	4	0	0	2	2
Alternate Route Service	2	2	2	4	2	5	2	2
Bulwark Service	8	8	6	6	0	0	2.5	2

Table 3-42 cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Request/ Limit Istio CPU	Request/ Limit Istio Memory
ndbappmysqld		12		20	12	5	5
ndbmcmd		3		10	2	2	2
ndbmtl		12		129	10	6	6

Table 3-42 (Cont.) cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Request/ Limit Istio CPU	Request/ Limit Istio Memory
ndbmysqld		4		54	6	4	4

3.2.2.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The average CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-43 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site1
ocpcf-alternate-route	0%/80%
ocpcf-appinfo	0%/80%
ocpcf-bulwark	0%/60%
ocpcf-occnf-config-server	16%/80%
ocpcf-occnf-egress-gateway	60%/80%
ocpcf-occnf-ingress-gateway	55%/80%
ocpcf-occnf-nrf-client-nfdiscovery	43%/80%
ocpcf-occnf-nrf-client-nfmanagement	0%/80%
ocpcf-oc-binding	0%/60%
ocpcf-occnf-chf-connector	0%/50%
ocpcf-occnf-udr-connector	48%/50%
ocpcf-ocpm-audit-service	0%/60%
ocpcf-ocpm-policyds	49%/60%
ocpcf-ocpm-pre	25%/80%
ocpcf-pcf-amservice	32%/30%
ocpcf-pcf-ueservice	54%/30%
ocpcf-ocpm-queryservice	0%/80%

Table 3-44 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site1	CPU (X/Y) - Site2
ndbappmysqld	26%/80%	20%/80%
ndbmcmd	0%/80%	0%/80%
ndbmtd	63%/80%	60%/80%
ndbmysqld	6%/80%	1%/80%

3.2.2.3 Results

Table 3-45 Latency Observations

AM/UE Flow	50th Percentile (mean)	95th Percentile (mean)
AM Create	48ms	96ms
AM Delete	50ms	93ms
UE Create	72ms	125ms
UE Delete	7ms	11ms

3.2.3 Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with each site handling 43K TPS traffic and ASM Enabled

This test run benchmarks the performance and capacity of Policy SM data call model that is deployed in PCF mode on a two-site georedundant setup. The PCF application handles a total (Ingress + Egress) traffic of 60K TPS, with each site handling a traffic of 21.5K TPS. For this setup Aspen Service Mesh (ASM) was enabled.

In this test setup, the Georedundant (GR) mode was enabled in cnDBTier and it was configured for 3 channel replication.

3.2.3.1 Test Case and Setup Details

Table 3-46 Test Case Parameters

Parameters	Values
Call Rate (Ingress + Egress)	21.5K TPS on Site1, 21.5K TPS on Site2
ASM	Enable
Traffic Ratio	Internet:- 1 SM Create : 74 SM Updates : 1 SM Delete IMS:- 1 SM Create : 8 SM Updates : 1 SM Delete APP:- 1 SM Create : 0 SM Updates : 1 SM Delete ADMIN:- 1 SM Create : 0 SM Updates : 1 SM Delete IMS Rx:- 1 Create : 1 STR
Active Subscribers	10000000 subscribers and 20000000 sessions

Policy Project Details:

The Policy design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was High.

Policy Project's Complexity Level Definition:

- Low– No Usage of Loops in Blockly logic, No JSON operations, No complex Java Script code in Object Expression /Statement Expression.
- Medium - Usage of Loops in Blockly logic, Policy Table Wildcard match <= 3 fields, MatchList < 3, 3 < RegEx match < 6
- High - JSON Operations – Custom, complex Java Script code in Object Expression / Statement Expression, Policy Table Wildcard match > 3 fields, MatchLists >= 3, RegEx mat >= 6

Table 3-47 PCF Configuration

Name	Status
Bulwark Service	Enable
Binding Service	Enable
Subscriber State Variable (SSV)	Enable
Validate_user	Disable
Alternate Route	Disable
Audit Service	Enable
Enable Custom JSON	Enable

Table 3-48 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Enable
BSF (N7-Nbsf)	Enable
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA

Table 3-49 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Table 3-50 Configuring Policy Helm Parameters

Service Name	Policy Helm Configuration
Ingress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>

Table 3-50 (Cont.) Configuring Policy Helm Parameters

Service Name	Policy Helm Configuration
Egress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>

Note

The Policy customized parameters values remains same for both site1 and site2.

Table 3-51 Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
binlog_cache_size	10485760	<pre> additionalndbconfigurations: mysqld: binlog_cache_size: '10485760' </pre>
ConnectCheckIntervalDelay	500	<pre> additionalndbconfigurations: ndb: ConnectCheckIntervalDelay: 500 </pre>
NoOfFragmentLogFiles	32	<pre> # values for configuration files, cnf ndbconfigurations: ndb: NoOfFragmentLogFiles: 32 </pre>

Table 3-51 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
NoOfFragmentLogParts	4	<pre> ndbconfigurations: ndb: NoOfFragmentLogParts: 4 </pre>
MaxNoOfExecutionThreads	11	<pre> ndbconfigurations: ndb: MaxNoOfExecutionThreads: 11 </pre>
FragmentLogFileSize	128M	<pre> additionalndbconfigurations: ndb: FragmentLogFileSize: 128M </pre>
binlogthreadstore.capacity	5	<pre> db-monitor-svc: binlogthreadstore: capacity: 5 </pre>
ndb_allow_copying_alter_table	ON	<pre> additionalndbconfigurations: mysqld: # use replica_skip_errors as slave-skip-errors/ slave_skip_errors is deprecated ndb_allow_copying_alter_ table: 'ON' </pre>

Note

The cnDBTier customized parameters values remains same for both site1 and site2.

Table 3-52 Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Isito CPU	Request/ Limit Isito Memory
Appinfo	1	1	0.5	1	2	2	2	2
Audit Service	2	2	4	4	2	2	2	2
CM Service	2	4	0.5	2	2	2	2	2
Config Service	4	4	0.5	2	2	2	2	2
Egress Gateway	4	4	6	6	2	6	2	2
Ingress Gateway	5	5	6	6	2	27	2.5	2
NRF Client Management	1	1	1	1	2	2	2	2
Diameter Gateway	4	4	1	2	2	2	2	2
Diameter Connector	4	4	1	2	2	2	2	2
AM Service	8	8	1	4	0	0	2	2
UE Service	8	8	1	4	0	0	2	2
NRF Client Discovery	4	4	2	2	2	2	2	2
Query Service	1	2	1	1	2	2	2	2
PCRF Core Service	8	8	8	8	0	0	2	2
Performance	1	1	0.5	1	2	2	2	2
PRE Service	4	4	4	4	2	55	1.5	2
SM Service	7	7	10	10	2	76	2	2
PDS Service	7	7	8	8	2	21	2.5	4
UDR Connector	6	6	4	2	2	2	2	2

Table 3-52 (Cont.) Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Isito CPU	Request/ Limit Isito Memory
CHF Connector	6	6	4	4	2	2	2	2
LDAP Gateway Service	3	4	1	2	0	0	2	2
Binding Service	6	6	8	8	2	3	2.5	2
SOAP Connector	2	4	4	4	0	0	2	2
Alternate Route Service	2	2	2	4	2	2	2	2
Bulwark Service	8	8	6	6	2	19	2.5	2

Table 3-53 cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
ndbappmysqld	12	12	18	18	18
ndbmngmd	3	3	8	8	2
ndbmtl	10	10	132	132	10
ndbmysqld	4	4	54	54	12

Note

Min Replica = Max Replica

3.2.3.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The average CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-54 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ocpcf-occnp-alternate route	0.10%%/9.56%	0.10%%/9.97%
ocpcf-appinfo	4.40%/25.78%	4.50%/25.34%
ocpcf-bulwark	17.55%/17.13%	0.04%/14.53%

Table 3-54 (Cont.) Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ocpcf-occn-p-config-server	6.17%/42.65%	3.70%/40.19%
ocpcf-occn-p-egress-gateway	19.48%/21.97%	0.04%/20.34%
ocpcf-occn-p-ingress-gateway	16.50%/32.03%	0.54%/25.63%
ocpcf-occn-p-nrf-client-nfdiscovery	7.94%/51.84%	0.07%/38.38%
ocpcf-occn-p-nrf-client-nfmanagement	1.75%/50.29%	0.35%/48.73%
ocpcf-oc-binding	12.36%/17.44%	0.05%/12.41%
ocpcf-occn-p-chf-connector	11.87%/22.10%	0.05%/18.97%
ocpcf-occn-p-udr-connector	14.83%/23.34%	0.06%/17.67%
ocpcf-occpm-audit-service	0.22%/16.35%	0.10%/12.41%
ocpcf-occpm-policyds	21.13%/22.16%	0.03%/18.47%
ocpcf-occpm-pre	21.64%/47.43%	0.21%/12.82%
ocpcf-pcf-sm-service	22.38%/25.81%	0.04%/18.15%
ocpcf-occpm-queryservice	0.05%/23.54%	0.05%/24.12%

Table 3-55 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site1	CPU (X/Y) - Site2
ndbappmysqld	28.57%/41.04%	0.31%/32.17%
ndbmngmd	0.22%/25.38%	0.22%/25.41%
ndbmttd	55.88%/46.89%	9.32%/46.90%

3.2.3.3 Results

Table 3-56 Latency Observations

Services	Average Latency (ms)
Ingress	34.8
SM	30.7
PDS	13.8
UDR	3.97
NRFClient Discovery	2.60
CHF	2.80
Binding	16.0
Diam-connector	1.36
Egress	8.39

3.2.4 Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with each site handling 30K TPS traffic and ASM Enabled

This test run benchmarks the performance and capacity of Policy SM data call model that is deployed in PCF mode on a two-site georedundant setup. The PCF application handles a total

(Ingress + Egress) traffic of 60K TPS, with each site handling a traffic of 30K TPS. For this setup Aspen Service Mesh (ASM) was enabled.

In this test setup, the Georedundant (GR) mode was enabled in cnDBTier and it was configured for 3 channel replication.

3.2.4.1 Test Case and Setup Details

Table 3-57 Test Case Parmeters

Parameters	Values
Call Rate (Ingress + Egress)	30K TPS on Site1, 30K TPS on Site2
ASM	Enable
Traffic Ratio	Internet:- 1 SM Create : 74 SM Updates : 1 SM Delete IMS Rx:- 1 Create : 1 Update : 1 STR
Active Subscribers	393590 (Site1) + 393589 (Site2) = 787179

Policy Project Details:

The Policy design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was High.

Policy Project's Complexity Level Definition:

- Low– No Usage of Loops in Blockly logic, No JSON operations, No complex Java Script code in Object Expression /Statement Expression.
- Medium - Usage of Loops in Blockly logic, Policy Table Wildcard match <= 3 fields, MatchList < 3, 3 < RegEx match < 6
- High - JSON Operations – Custom, complex Java Script code in Object Expression / Statement Expression, Policy Table Wildcard match > 3 fields, MatchLists >= 3, RegEx mat >= 6

Table 3-58 Call Model

Service Name	DNN1 SM Service (MPS)		DNN2 SM Service and Rx Interface (MPS)				Total MPS
	Inbound Message	Outbound Message	Inbound Message	Outbound Message	Inbound Message	Outbound Message	
Ingress Gateway	49000	49000	1520	1520	0	0	101040
SM Service	49654	209036	1526	10739	2533	7094	280590
PRE Service	49000	0	1520	0	1520	0	52040
PDS Service	58114	3924	3623	525	3040	0	69230
Egress Gateway	4578	4578	1545	1545	1520	1520	15290
NRF Discovery	654	654	6	6	0	0	1320

Table 3-58 (Cont.) Call Model

Service Name	DNN1 SM Service (MPS)		DNN2 SM Service and Rx Interface (MPS)				Total MPS
	Inbound Message	Outbound Message	Inbound Message	Outbound Message	Inbound Message	Outbound Message	
UDR Connector	1962	2616	513	519	0	0	5610
CHF Connector	1308	1308	6	6	0	0	2630
Binding Service	1307	0	2027	1014	0	0	4350
Diameter Connector	0	0	507	507	1520	2533	5070
Diameter Gateway	0	0	507	507	1520	1520	4060
Bulwark Service	99308	0	3052	0	1013	0	103380

Table 3-59 PCF Configuration

Name	Status
Bulwark Service	Enable
Binding Service	Enable
Subscriber State Variable (SSV)	Enable
Validate_user	Disable
Alternate Route	Disable
Audit Service	Enable
Enable Custom JSON	Enable

Table 3-60 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Enable
BSF (N7-Nbsf)	Enable
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA

Table 3-61 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA

Table 3-61 (Cont.) PCRF Interfaces

Feature Name	Status
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Table 3-62 Configuring Policy Helm Parameters

Service Name	Policy Helm Configuration
Ingress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 16 maxPoolSize: 16 queueCapacity: 3300 </pre>
Egress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 16 maxPoolSize: 16 queueCapacity: 3300 </pre>

Note

The Policy customized parameters values remains same for both site1 and site2.

Table 3-63 Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
binlog_cache_size	10485760	<pre> additionalnldbconfigurations: mysqld: binlog_cache_size: '10485760' </pre>

Table 3-63 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
ConnectCheckIntervalDelay	500	<pre> additionalndbconfigurations: ndb: ConnectCheckIntervalDelay: 500 </pre>
NoOfFragmentLogFiles	32	<pre> # values for configuration files, cnf ndbconfigurations: ndb: NoOfFragmentLogFiles: 32 </pre>
NoOfFragmentLogParts	4	<pre> ndbconfigurations: ndb: NoOfFragmentLogParts: 4 </pre>
MaxNoOfExecutionThreads	11	<pre> ndbconfigurations: ndb: MaxNoOfExecutionThreads: 11 </pre>
FragmentLogFileSize	128M	<pre> additionalndbconfigurations: ndb: FragmentLogFileSize: 128M </pre>

Table 3-63 (Cont.) Configuring cnDBTier Helm Parameters

Helm Parameter	Value	cnDBTier Helm Configuration
binlogthreadstore.capacity	5	<pre>db-monitor-svc: binlogthreadstore: capacity: 5</pre>
ndb_allow_copying_alter_table	ON	<pre>additionalndbconfigurations: mysqld: # use replica_skip_errors as slave-skip-errors/ slave_skip_errors is deprecated ndb_allow_copying_alter_ table: 'ON'</pre>

Note

The cnDBTier customized parameters values remains same for both site1 and site2.

Table 3-64 Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Isito CPU	Request/ Limit Isito Memory
Appinfo	1	1	0.5	1	2	2	2	2
Audit Service	2	2	4	4	2	2	2	2
CM Service	2	4	0.5	2	2	2	2	2
Config Service	4	4	0.5	2	2	2	2	2
Egress Gateway	4	4	6	6	2	6	2	2
Ingress Gateway	5	5	6	6	2	27	2.5	2
NRF Client Management	1	1	1	1	2	2	2	2

Table 3-64 (Cont.) Policy Microservices Resources

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Min Replicas	Max Replicas	Request/ Limit Isito CPU	Request/ Limit Isito Memory
Diameter Gateway	4	4	1	2	2	2	2	2
Diameter Connector	4	4	1	2	2	2	2	2
AM Service	8	8	1	4	0	0	2	2
UE Service	8	8	1	4	0	0	2	2
NRF Client Discovery	4	4	2	2	2	2	2	2
Query Service	1	2	1	1	2	2	2	2
PCRF Core Service	8	8	8	8	0	0	2	2
Performance	1	1	0.5	1	2	2	2	2
PRE Service	4	4	4	4	2	55	1.5	2
SM Service	7	7	10	10	2	76	2.5	2
PDS Service	7	7	8	8	2	21	2.5	4
UDR Connector	6	6	4	2	2	2	2	2
CHF Connector	6	6	4	4	2	2	2	2
LDAP Gateway Service	3	4	1	2	0	0	2	2
Binding Service	6	6	8	8	2	3	2.5	2
SOAP Connector	2	4	4	4	0	0	2	2
Alternate Route Service	2	2	2	4	2	2	2	2
Bulwark Service	8	8	6	6	2	19	2.5	2

Table 3-65 cnoDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
ndbappmysqld	12	12	18	18	18
ndbmgmd	3	3	8	8	2
ndbmtd	10	10	132	132	10
ndbmysqld	4	4	54	54	12

Note

Min Replica = Max Replica

3.2.4.2 CPU Utilization

This section lists the CPU utilization for Policy and cnoDBTier microservices. The average CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-66 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site1	CPU (X/Y) - Site2
ocpcf-alternate-route	0%/80%	0%/80%
ocpcf-appinfo	1%/80%	1%/80%
ocpcf-bulwark	22%/60%	23%/60%
ocpcf-occn-p-config-server	9%/80%	10%/80%
ocpcf-oc-diam-connector	8%/40%	8%/40%
ocpcf-occn-p-egress-gateway	11%/80%	10%/80%
ocpcf-occn-p-ingress-gateway	19%/80%	24%/80%
ocpcf-occn-p-nrf-client-nfdiscovery	5%/80%	5%/80%
ocpcf-occn-p-nrf-client-nfmanagement	0%/80%	0%/80%
ocpcf-oc-binding	17%/60%	17%/60%
ocpcf-occn-p-chf-connector	7%/50%	7%/50%
ocpcf-occn-p-udr-connector	15%/50%	14%/50%
ocpcf-ocpm-audit-service	0%/50%	0%/50%
ocpcf-ocpm-policyds	19%/60%	19%/60%
ocpcf-ocpm-pre	26%/80%	27%/80%
ocpcf-pcf-amservice	0%/30%	0%/30%
ocpcf-pcf-ueservice	0%/30%	0%/30%
ocpcf-pcf-smsservice	25%/50%	25%/50%
ocpcf-ocpm-queryservice	0%80%	0%80%

Table 3-67 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site1	CPU (X/Y) - Site2
ndbappmysqld	42%/80%	37%/80%
ndbmcmd	0%/80%	0%/80%
ndbmtd	32%/80%	31%/80%
ndbmysqld	4%/80%	4%/80%

3.2.4.3 Results

Table 3-68 Latency Observations

SM Call Flow	50th Percentile (mean)	95th Percentile (mean)
SM Create	37ms	57ms
SM Update	14ms	30ms
SM Delete	14ms	23ms

3.2.5 Test Scenario: PCF SM Call Model on Two-Site GeoRedundant setup, with 41K TPS Traffic on Site-1 and ASM Enabled

This test run benchmarks the performance and capacity of Policy SM data call model that is deployed in PCF mode on a two-site georedundant setup. The PCF application handles a total (Ingress + Egress+Diameter) traffic of 41K TPS on either site. For this setup Aspen Service Mesh (ASM) was enabled.

3.2.5.1 Test Case and Setup Details

Table 3-69 Test Case Parameters

Parameters	Values
Call Rate (Ingress + Egress + Diameter)	41K TPS on Site-1
ASM	Enable
Traffic Ratio	<ul style="list-style-type: none"> Internet – 1:15:1 (Create:Update:Delete) IMS – 1:8:1 (Create:Update:Delete) App - 1:0:1 (Create:Update:Delete) Admin - 1:0:1 (Create:Update:Delete)
Active Subscribers	10000000 subscribers and 20000000 sessions

Policy Project Details:

The Policy design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was High.

Policy Project's Complexity Level Definition:

- Low– No Usage of Loops in Blockly logic, No JSON operations, No complex Java Script code in Object Expression /Statement Expression.

- Medium - Usage of Loops in Blockly logic, Policy Table Wildcard match <= 3 fields, MatchList < 3, 3 < RegEx match < 6
- High - JSON Operations – Custom, complex Java Script code in Object Expression / Statement Expression, Policy Table Wildcard match > 3 fields, MatchLists >= 3, RegEx mat >= 6

Table 3-70 PCF Configuration

Name	Status
Bulwark Service	Enable
Binding Service	Enable
Subscriber State Variable (SSV)	Enable
Validate_user	Disable
Alternate Route	Enable
Audit Service	Enable
Enable Custom JSON	Enable

Table 3-71 PCF Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Enable
BSF (N7-Nbsf)	Enable
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA

Table 3-72 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Table 3-73 Configuring Policy Helm Parameters

Service Name	Policy Helm Configuration
Ingress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>
Egress Gateway	<pre> ingress-gateway: applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>

Note

The Policy customized parameters values remains same for both site1 and site2.

Table 3-74 Configuring cnDBTier Helm Parameters

Helm Parameter	Value
global.additionalNdbconfigurations.mysqlId.binlog_c ache_size	10485760
global.additionalNdbconfigurations.ndb.ConnectCh eckIntervalDelay	0
global.additionalNdbconfigurations.ndb.NoOfFragm entLogFiles	64
global.additionalNdbconfigurations.ndb.NoOfFragm entLogParts	4
global.additionalNdbconfigurations.ndb.MaxNoOfEx ecutionThreads	11
global.additionalNdbconfigurations.ndb.FragmentLo gFileSize	32M
db-monitor-svc.binlogthreadstore.capacity	5
global.additionalNdbconfigurations.mysqlId.ndb_allo w_copying_alter_table	ON
global.additionalNdbconfigurations.ndb.HeartbeatIn tervalDbDb	1250

Note

The cnDBTier customized parameters values remains same for both site1 and site2.

Table 3-75 Policy Microservices Resources

Service Name	CPU Limit Per Pod	CPU Request Per Pod	Memory Limit Per Pod (Gi)	Memory Request Per Pod (Gi)	Min Replicas	Max Replicas
Appinfo	2	2	1	0.5	2	2
Appinfo-Istio	2	2	2	2	2	2
Audit Service	2	2	4	4	2	2
Audit Service-Istio	2	2	2	2	2	2
CM Service	4	2	2	0.5	2	2
CM Service-Istio	2	2	2	2	2	2
Config Service	4	4	2	0.5	2	2
Config Service-Istio	2	2	2	2	2	2
Egress Gateway	8	8	6	6	9	9
Egress Gateway-Istio	4	4	2	2	9	9
Ingress Gateway	5	5	6	6	29	29
Ingress Gateway-Istio	2.5	2.5	2	2	29	29
Nrf Client Management	1	1	1	1	2	2
Nrf Client Management-Istio	2	2	2	2	2	2
Diameter Gateway	4	4	2	1	2	2
Diameter Gateway-Istio	2	2	2	2	2	2
Diameter Connector	4	4	2	1	2	2
Diameter Connector-Istio	2	2	2	2	2	2
Nrf Client Discovery	4	4	2	2	2	2
Nrf Client Discovery-Istio	2	2	2	2	2	2
Query Service	2	1	1	1	2	2
Query Service-Istio	2	2	2	2	2	2
Performance	1	1	1	0.5	2	2
PRE Service	4	4	4	4	39	39
PRE Service-Istio	1.5	1.5	2	2	39	39
SM Service	7	7	10	10	64	64

Table 3-75 (Cont.) Policy Microservices Resources

Service Name	CPU Limit Per Pod	CPU Request Per Pod	Memory Limit Per Pod (Gi)	Memory Request Per Pod (Gi)	Min Replicas	Max Replicas
SM Service-Istio	2.5	2.5	2	2	64	64
UDR Connector	6	6	4	4	8	8
UDR Connector-Istio	2	2	2	2	8	8
CHF Connector	6	6	4	4	4	4
CHF Connector-Istio	2	2	2	2	4	4
Binding Service	6	6	8	8	11	11
Binding Service-Istio	2.5	2.5	2	2	11	11
Alternate Route Service	2	2	4	2	2	2
Alternate Route Service-Istio	2	2	2	2	2	2
Bulwark Service	8	8	6	6	15	15
Bulwark Service-Istio	2.5	2.5	2	2	15	15

Table 3-76 cnDBTier Microservices Resources:

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
ndbappmysqld	12	12	18	18	18
ndbappmysqld-istio	3	3	2	2	18
ndbmgmd	3	3	8	8	2
ndbmgmd-istio	1	1	2	2	2
ndbmt	10	10	132	132	10
ndbmt-istio	4	4	2	2	10
ndbmysqld	4	4	24	24	0
ndbmysqld-istio	5	5	4	4	12

Note

Min Replica = Max Replica

3.2.5.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The average CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Table 3-77 Policy Microservices Resource Utilization

Service	CPU (X/Y) - Site 1	CPU (X/Y) - Site 2
ocpcf-occnp-alternate route	32.90%%/26.37%	None
ocpcf-appinfo	2.45%/26.51%	None
ocpcf-bulwark	25.43%/23.64%	None
ocpcf-occnp-config-server	6.39%/45.14%	None
ocpcf-occnp-egress-gateway	15.74%/34.63%	None
ocpcf-occnp-ingress-gateway	19.09%/44.05%	None
ocpcf-occnp-nrf-client-nfdiscovery	3.78%/67.32%	None
ocpcf-occnp-nrf-client-nfmanagement	0.35%/49.46%	None
ocpcf-oc-binding	14.30%/39.18%	None
ocpcf-occnp-chf-connector	12.21%/22.02%	None
ocpcf-occnp-udr-connector	13.64%/30.17%	None
ocpcf-ocpm-audit-service	1.20%/22.68%	None
ocpcf-ocpm-policyds	23.23%/51.93%	None
ocpcf-ocpm-pre	23.32%/45.91%	None
ocpcf-pcf-smsservice	27.51%/56.20%	None
ocpcf-ocpm-queryservice	0.05%/23.54%	None

Table 3-78 cnDBTier Services Resource Utilization

Name	CPU (X/Y) - Site1	CPU (X/Y) - Site2
ndbappmysqld	33.69%/39.51%	0.16%/24.91%
ndbmcmd	0.23%/25.37%	0.28%/25.40%
ndbmtl	52.08%/66.69%	8.63%/66.61%

3.2.5.3 Results

Table 3-79 Latency Observations

Services	Average Latency (ms)
Ingress	58.7
SM	53.2
PDS	24.9
UDR	3.28
NRFClient Discovery	0.15
CHF	2.95
Binding	27.0

Table 3-79 (Cont.) Latency Observations

Services	Average Latency (ms)
Diam-connector	1.09
Egress	0.53

3.2.6 Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Each Site Handling 30K TPS Traffic and ASM Enabled

This test run benchmarks the performance and capacity of Policy AM/UE data call model that is deployed in PCF mode. The PCF application handles a total (Ingress + Egress) traffic of 60K TPS, with each site handling a traffic of 30K TPS. For this setup, Aspen Service Mesh (ASM) was enabled between Policy services and it was disabled between Policy services and cnDBTier data services. Application data compression was enabled at AM, UE, and PDS services. The Multithreaded Applier (MTA) feature that helps in peak replication throughput was enabled at cnDBTier.

3.2.6.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Ingress + Egress)	60K TPS (30K on site-1 and 30K on SITE-2)
ASM	Enable
Traffic Ratio	AM 1-CREATE 0-update 1-delete UE 1-CREATE 0-update 1-delete
Active User Count	12000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match ≤ 3 fields, MatchList < 3 , and $3 < \text{RegEx match} < 6$
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists ≥ 3 , and RegEx mat ≥ 6

Call Model Data**Table 3-80 Traffic distribution**

	Ingress Gateway	Egress Gateway	Total Ingress/ Egress Traffic	Ingress Gateway	Egress Gateway	Total Ingress/ Egress Traffic
UE service	Site 1			Site 2		
	3157	10953	14109	3036	10579	13615
AM service	3158	10953	14111	3078	10579	13657
Total			28220			27271

Policy Configurations

Following Policy configurations were either enabled or disabled for running this call flow:

Table 3-81 Policy microservices configuration

Name	Status
Bulwark	Enabled
Binding	Disabled
Subscriber State Variable (SSV)	Enabled
Validate_user	Disabled
Alternate Route	Disabled
Audit	Enabled
Compression (Binding & SM Service)	Enabled
SYSTEM.COLLISION.DETECTION	Enabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-82 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enabled
N36 UDR subscription (N7/N15-Nudr)	Enabled
UDR on-demand nrf discovery	Disabled
CHF (Nchf)	Enabled
BSF (N7-Nbsf)	Enabled
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Subscriber HTTP Notifier (Gx)	NA

Table 3-83 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA

Table 3-83 (Cont.) PCRF interfaces

Feature Name	Status
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring Policy Helm Parameters

The following Policy optimization parameters were configured for this run:

Table 3-84 Optimization parameters for Policy services

Service	Policy Helm Configurations
policyds	<pre> - name: DATASOURCE_HIKARI_MIN_IDLE value: "90" - name: DATASOURCE_HIKARI_MAX_POOL_SIZE value: "90" - name: DEFAULT_BOUNDED_ELASTIC_QUEUE_SIZE value: "80" </pre>
UE	DB_MAX_POOL_SIZE=60
INGRESS	<pre> applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>
EGRESS	<pre> applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500 </pre>

Policy Microservices Resources

Table 3-85 Policy microservices Resource allocation for Site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
Appinfo	1	1	1 Gi	512Mi	2	2	2 Gi
Audit Service	2	2	4 Gi	4 Gi	2	2	2 Gi
CM Service	4	4	2 Gi	2 Gi	2	2	2 Gi
Config Service	4	2	2Gi	2Gi	2	2	2 Gi
Egress Gateway	2	2	6Gi	6Gi	27	4	2 Gi
Ingress Gateway	5	5	6Gi	6Gi	8	2500m	2Gi
NRF Client NF Discovery	6	6	10Gi	10Gi	9	2	2Gi
NRF Client Management	1	1	1Gi	1Gi	1	2	2Gi
AM Service	6	6		10Gi	12	3	2Gi
UE Service	8	8	2Gi	2Gi	20	3	1Gi
Query Service	2	1	1Gi	1Gi	2		
Performance	1	1	1Gi	512Mi	2		
PRE	4	4	4Gi	4Gi	7	1500m	2Gi
SM Service	1	1	1Gi	1Gi	1	3	2Gi
PDS	7	7	8Gi	8Gi	24	3	4 Gi
UDR Connector	4	4	4Gi	4Gi	20	2	2Gi
CHF Connector/ User Service	6	6	4Gi	4Gi	8	2	2Gi
Alternate Route Service	2	2	4Gi	2Gi	1	2	2Gi
Bulwark Service	8	8	4Gi	4Gi	7	3	4Gi

Table 3-86 Policy microservices Resource allocation for site2

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
Appinfo	1	1	1 Gi	512Mi	2	2	2 Gi
Audit Service	2	2	4 Gi	4 Gi	2	2	2 Gi
CM Service	4	4	2 Gi	2 Gi	2	2	2 Gi
Config Service	4	2	2Gi	500m	2	2	2 Gi
Egress Gateway	4	4	6Gi	6Gi	20	2	2 Gi
Ingress Gateway	5	5	6Gi	6Gi	8	2.5	2Gi
NRF Client NF Discovery	6	6	10Gi	10Gi	9	2	2Gi
NRF Client Management	1	1	1Gi	1Gi	1	2	2Gi
AM Service	6	6	10Gi	10Gi	9	3	2Gi
UE Service	8	8	4Gi	4Gi	18	2	2Gi
Query Service	2	1	1Gi	1Gi	2		
Performance	1	1	1Gi	512Mi	2		
PRE	4	4	4Gi	4Gi	7	1.5	2Gi
SM Service	1	1	1Gi	1Gi	1	0.5	2Gi
PDS	7	7	8Gi	8Gi	22	2.5	4Gi
UDR Connector	4	4	4Gi	4Gi	20	2	2Gi
CHF Connector/ User Service	6	6	4Gi	4Gi	3	2	2Gi
Alternate Route Service	0.5	0.5	4Gi	2Gi	1	0.5	2Gi
Bulwark Service	8	8	4Gi	4Gi	5	2	4Gi

Table 3-87 CnDBTier Resource allocation for site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
ndbappmysqld	12	12	20Gi	20Gi	12	5	5Gi
ndbmcmd	3	3	8Gi	8Gi	2	3	1Gi
ndbmtl	12	12	129Gi	129Gi	10	6	6Gi
ndbmssql	4	4	16Gi	16Gi	6	5	5Gi

Table 3-88 CnDBTier resource allocation for site2

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
ndbappmysqld	12	12	20Gi	20Gi	12	5	5Gi
ndbmcmd	3	3	8Gi	8Gi	2	3	1Gi
ndbmtl	12	12	129Gi	129Gi	10	6	6Gi
ndbmssql	4	4	16Gi	16Gi	6	5	5Gi

3.2.6.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-89 CPU/Memory Utilization by Policy Microservices

Service	CPU (Site 1)	Memory (Site 1)	CPU (Site 2)	Memory (Site 2)
ocpcf-occpn-alternate route/istio	0.10%	4.88%	0.60%	4.44%
ocpcf-occpn-alternate route	0.15%	9.38%	0.60%	6.76%
ocpcf-appinfo/istio	0.18%	5.35%	0.20%	5.18%
ocpcf-appinfo	2.65%	23.78%	4.40%	23.58%
ocpcf-bulwark/istio	25.27%	2.30%	59.09%	2.88%

Table 3-89 (Cont.) CPU/Memory Utilization by Policy Microservices

Service	CPU (Site 1)	Memory (Site 1)	CPU (Site 2)	Memory (Site 2)
ocpcf-bulwark	17.78% ¹	17.36%	29.15%	20.51%
ocpcf-occnp-config-server/istio	11.30%	5.42% ¹	14.03%	6.42%
ocpcf-occnp-config-server	7.51%	29.98%	9.46%	30.44%
ocpcf-occnp-egress-gateway/istio	5.90%	5.18%	13.11%	5.89%
ocpcf-occnp-egress-gateway	23.25%	19.32%	38.80%	20.48%
ocpcf-occnp-ingress-gateway/istio	21.98%	6.99%	18.80%	7.64%
ocpcf-occnp-ingress-gateway	19.87%	24.11%	23.62%	23.45%
ocpcf-occnp-nrf-client-nfdiscovery/istio	17.95%	5.21%	27.92%	5.83%
ocpcf-occnp-nrf-client-nfdiscovery	9.81%	9.91%	13.84%	9.48%
ocpcf-occnp-nrf-client-nfmanagement/istio	0.15%	4.79%	0.20%	5.22%
ocpcf-occnp-nrf-client-nfmanagement	0.40%	44.92%	0.40%	47.17%
ocpcf-performance/perf-info	1.90%	11.82%	1.00%	12.40%
ocpcf-occnp-chf-connector/istio	14.88%	5.22%	47.70%	6.23%
ocpcf-occnp-chf-connector	7.78%	14.96%	24.25%	14.87%
ocpcf-occnp-udr-connector/istio	20.30%	5.52%	29.43%	6.24%
ocpcf-occnp-udr-connector	18.32%	15.26%	23.51%	15.08%
ocpcf-ocpm-audit-service/istio	0.18%	4.61%	0.25%	5.10%
ocpcf-ocpm-audit-service	0.22%	13.00%	0.83%	12.59%
ocpcf-ocpm-cm-service/istio	0.80%	4.96%	0.92%	5.20%
ocpcf-ocpm-cm-service/cm-service	0.76%	28.34%	0.83%	30.76%
ocpcf-ocpm-policyds/istio	21.30%	2.84%	35.80%	3.03%
ocpcf-ocpm-policyds	24.84%	30.74%	33.41%	31.08%

Table 3-89 (Cont.) CPU/Memory Utilization by Policy Microservices

Service	CPU (Site 1)	Memory (Site 1)	CPU (Site 2)	Memory (Site 2)
ocpcf-occpn-amservice/istio	24.62%	5.72%	43.19%	6.43%
ocpcf-occpn-amservice	26.90%	9.40%	44.37%	10.71%
ocpcf-occpm-pre/istio	24.99%	5.81%	45.51%	5.82%
ocpcf-occpm-pre	18.59%	32.53%	30.70%	30.35%
ocpcf-pcf-smsservice/istio	0.17%	4.83%	.60%	6.01%
ocpcf-pcf-smsservice	0.40%	37.11%	0.40%	37.40%
ocpcf-pcf-ueservice/istio	15.49%	5.64%	35.09%	6.01%
ocpcf-pcf-ueservice	22.16%	34.16%	29.61%	38.23%
ocpcf-occpm-queryservice	0.05%	23.39%	0.50%	23.68%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-90 CPU/Memory Utilization by CnDBTier services

Service	CPU (Site 1)	Memory CPU (Site 1)	CPU (Site 2)	Memory (Site 2)
ndbappmysqld/istio	23.14%	2.48%	22.78%	2.50%
ndbappmysqld/mysqlndbcluster	21.31%	50.17%	26.48%	35.47%
ndbappmysqld/init-sidecar	2.25%	0.39%	3.00%	0.39%
ndbmcmd/istio-proxy	0.33%	10.74%	0.43%	11.38%
ndbmcmd/mysqlndbcluster	0.25%	25.21%	0.35%	25.16%
ndbmtl/istio-proxy	47.02%	2.06%	31.61%	1.96%
ndbmtl/mysqlndbcluster	44.95%	81.17%	42.45%	79.71%
ndbmysqld/istio-proxy	0.00%	0.00%	0.00%	0.00%
ndbmysqld/mysqlndbcluster	4.23%	30.30%	7.72%	28.85%
ndbmysqld/init-sidecar	2.00%	0.39%	2.83%	0.59%

3.2.6.3 Results

Average Latency Observations for the AM and UE call flow:

Call Flow	50th Percentile (ms)	95th Percentile (ms)
AM CREATE	40.8	110
AM DELETE	35.7	60.2
UE CREATE	56.8	102
UE DELETE	6.37	8.94

3.2.7 Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Single-Site Handling 60K TPS Traffic and ASM Enabled

This test run benchmarks the performance and capacity of Policy AM/UE data call model that is deployed in PCF mode.. The PCF application handles a total traffic (Ingress + Egress) of 60K TPS on one site and there is no traffic on the other site. APP Compression was enabled. The test was run for 1.0 hour duration. For this setup, Aspen Service Mesh (ASM) was enabled between Policy services and it was disabled between Policy service pods and DB data pods.

In this test setup, the Georedundant (GR) mode was enabled in cnDBTier. It was configured for 2 channel replication and the Application Data compression was enabled at AM, UE, and PDS services on Site 2.

3.2.7.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Ingress + Egress)	60k on site-1 and no traffic on site-2
ASM	Enable
Traffic Ratio	AM 1-Create 0-update 1-delete UE 1-Create 0-update 1-delete
Active User Count	12000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model Data**Table 3-91 Traffic distribution**

	Ingress Gateway	Egress Gateway	Total Ingress/ Egress Traffic	Ingress Gateway	Egress Gateway	Total Ingress/ Egress Traffic
UE service	Site 1			Site 2		
	6672	30024	36696	-	-	-
AM service	6672	16680	23352	-	-	-
Total			60048	-	-	-

Policy Configurations

Following Policy microservices were either enabled or disabled for running this call flow:

Table 3-92 Policy microservices configuration

Name	Status
Bulwark	Enabled
Binding	Disabled
Subscriber State Variable (SSV)	Enabled
Validate_user	Disabled
Alternate Route	Disabled
Audit	Enabled
Compression (Binding & SM Service)	Enabled
SYSTEM.COLLISION.DETECTION	Enabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-93 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enable
N36 UDR subscription (N7/N15-Nudr)	Enable
UDR on-demand nrf discovery	Disable
CHF (SM-Nchf)	Enable
BSF (N7-Nbsf)	Enable
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Subscriber HTTP Notifier (Gx)	NA

The following PCRF interfaces that were either enabled or disabled to run this call flow:

Table 3-94 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring Policy Helm Parameters

The following Policy optimization parameters were configured for this run:

Table 3-95 Optimization parameters for Policy services

Service	Policy Helm Configurations
policyds	<ul style="list-style-type: none"> - name: DATASOURCE_HIKARI_MIN_IDLE value: "90" - name: DATASOURCE_HIKARI_MAX_POOL_SIZE value: "90" - name: DEFAULT_BOUNDED_ELASTIC_QUEUE_SIZE value: "80"
UE	DB_MAX_POOL_SIZE=60
INGRESS	<pre>applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500</pre>
EGRESS	<pre>applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500</pre>

Policy Microservices Resources

Table 3-96 Policy microservices resource allocation for site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
Appinfo	1	1	1 Gi	512Mi	2	2	2 Gi
Audit Service	2	2	4 Gi	4 Gi	2	2	2 Gi
CM Service	4	4	2 Gi	2 Gi	2	2	2 Gi
Config Service	4	2	2Gi	2Gi	2	2	2 Gi
Egress Gateway	2	2	6Gi	6Gi	27	4	2 Gi
Ingress Gateway	5	5	6Gi	6Gi	8	2.5	2 Gi
NRF Client NF Discovery	6	6	10Gi	10Gi	9	2	2 Gi
NRF Client Management	1	1	1Gi	1Gi	1	2	2 Gi
AM Service	6	6	10Gi	10Gi	12	3	2 Gi
UE Service	8	8	2Gi	2Gi	20	2	1 Gi
Query Service	2	1	1Gi	1Gi	2		
Performance	1	1	1Gi	512Mi	2	2	1 Gi
PRE	4	4	4Gi	4Gi	7	1.5	2 Gi
SM Service	1	1	1Gi	1Gi	1	3	2 Gi
PDS	7	7	8Gi	8Gi	24	3	4 Gi
UDR Connector	4	4	4Gi	4Gi	20	2	2 Gi
CHF Connector/ User Service	6	6	4Gi	4Gi	8	2	2 Gi
Alternate Route Service	2	2	4Gi	2Gi	1	2	2 Gi
Bulwark Service	8	8	4Gi	4Gi	7	3	4 Gi

Table 3-97 Policy microservices resource allocation for site2

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory
Appinfo	1	1	1 Gi	512Mi	2	2	2 Gi
Audit Service	2	2	4 Gi	4 Gi	2	2	2 Gi
CM Service	4	4	2 Gi	2 Gi	2	2	2 Gi
Config Service	4	2	2Gi	500m	2	2	2 Gi
Egress Gateway	4	4	6Gi	6Gi	20	2	2 Gi
Ingress Gateway	5	5	6Gi	6Gi	8	2.5	2Gi
NRF Client NF Discovery	6	6	10Gi	10Gi	9	2	2 Gi
NRF Client Management	1	1	1Gi	1Gi	1	2	2 Gi
AM Service	6	6	10Gi	10Gi	9	3	2 Gi
UE Service	8	8	4Gi	4Gi	18	2	2 Gi
Query Service	2	1	1Gi	1Gi	2		
Performance	1	1	1Gi	512Mi	2		
PRE	4	4	4Gi	4Gi	7	1.5	2 Gi
SM Service	1	1	1Gi	1Gi	1	0.5	2 Gi
PDS	7	7	8Gi	8Gi	22	2.5	4 Gi
UDR Connector	4	4	4Gi	4Gi	20	2	2 Gi
CHF Connector/ User Service	6	6	4Gi	4Gi	3	2	2 Gi
Alternate Route Service	0.5	0.5	4Gi	2Gi	1	0.5	2 Gi
Bulwark Service	8	8	4Gi	4Gi	5	2	4 Gi

Table 3-98 CnDBTier resource allocation for site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory
ndbappmysqld/ mysqlndbcluster	12	12	20Gi	20Gi	12	5	5Gi
ndbappmysqld/init-sidecar	0.1	0.1	256Mi	256Mi	12		
ndbmgsmd/ mysqlndbcluster	3	3	8Gi	8Gi	2	3	1Gi
ndbmysqld/ mysqlndbcluster	12	12	129Gi	129Gi	10	6	6Gi
ndbmgsqld/ mysqlndbcluster	4	4	16Gi	16Gi	6	5	5Gi
ndbmgsqld/init-sidecar	0.1	0.1	256Mi	256Mi	6		

Table 3-99 CnDBTier resource allocation for site2

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory
ndbappmysqld/ mysqlndbcluster	12	12	20Gi	20Gi	12	5	5Gi
ndbappmysqld/init-sidecar	0.1	0.1	256Mi	256Mi	12		
ndbmgsmd/ mysqlndbcluster	3	3	8Gi	8Gi	2	3	1Gi
ndbmysqld/ mysqlndbcluster	12	12	129Gi	129Gi	10	6	6Gi
ndbmgsqld/ mysqlndbcluster	4	4	16Gi	16Gi	6	5	5Gi
ndbmgsqld/init-sidecar	0.1	0.1	256Mi	256Mi	6		

3.2.7.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-100 CPU/Memory Utilization by Policy Microservices

Service	CPU (Site 1)	Memory (Site 1)	CPU (Site 2)	Memory (Site 2)
ocpcf-appinfo/istio	0.25%	7.18%	0.22%	5.59%
ocpcf-appinfo	4.20	32.97%	2.50%	23.24%
ocpcf-bulwark/istio	0.10%	2.91%	0.15%	2.78%
ocpcf-bulwark	0.04%	37.21%	0.05%	12.23%
ocpcf-oc-binding/istio	0.20%	5.57%	0.30%	6.01%
ocpcf-oc-binding/binding	0.03%	7.73%	0.03%	7.46%
ocpcf-occnp-alternate route/istio	0.15%	5.27%	0.25%	5.42%
ocpcf-occnp-alternate route/istio	0.10%	9.59%	0.10%	9.35%
ocpcf-occnp-chf-connector/istio	11.60%	5.03%	0.50%	5.76%
ocpcf-occnp-chf-connector	12.10%	10.72%	0.08%	10.94%
ocpcf-occnp-config-server/istio	13.85%	6.13%	5.80%	6.23%
ocpcf-occnp-config-server	9.50%	43.14%	3.50%	36.67%
ocpcf-occnp-egress-gateway/istio	10.13%	5.40%	0.19%	5.92%
ocpcf-occnp-egress-gateway	49.76%	19.64%	0.07%	9.69%
ocpcf-occnp-ingress-gateway/istio	36.23%	10.00%	0.20%	5.85%
ocpcf-occnp-ingress-gateway	45.73%	32.97%	0.24%	19.07%
ocpcf-occnp-nrf-client-nfdiscovery/istio	59.12%	8.17%	0.26%	5.82%
ocpcf-occnp-nrf-client-nfdiscovery	51.44%	59.33%	0.08%	33.86%

Table 3-100 (Cont.) CPU/Memory Utilization by Policy Microservices

Service	CPU (Site 1)	Memory (Site 1)	CPU (Site 2)	Memory (Site 2)
ocpcf-occnp-nrf-client-nfmanagement/istio	0.70%	5.42%	0.20%	5.57%
ocpcf-occnp-nrf-client-nfmanagement	0.40%	44.82%	0.40%	46.39%
ocpcf-occnp-udr-connector/istio	69.88%	8.00%	0.47%	5.69%
ocpcf-occnp-udr-connector	35.60%	32.06%	0.08%	11.15%
ocpcf-ocpm-audit-service/istio	0.25%	5.59%	0.25%	5.47%
ocpcf-ocpm-audit-service	0.57%	23.69%	0.38%	13.01%
ocpcf-ocpm-cm-service/istio	0.85%	5.27%	0.55%	6.05%
ocpcf-ocpm-cm-service/cm-service	0.71%	37.21%	0.33%	33.81%
ocpcf-ocpm-policyds/istio	49.69%	3.91%	0.17%	2.86%
ocpcf-ocpm-policyds	40.46%	32.78%	0.03%	14.43%
ocpcf-ocpm-pre/istio	33.67%	7.14%	0.35%	6.24%
ocpcf-ocpm-pre	37.21%	49.02%	0.31%	8.65%
ocpcf-ocpm-queryservice	0.05%	28.22%	0.08%	24.41%
ocpcf-occnp-amservice/istio	32.87%	8.59%	0.39%	5.86%
ocpcf-occnp-amservice	29.83%	23.16%	0.04%	12.90%
ocpcf-pcf-ueservice/istio	56.27%	9.83%	0.35%	5.65%
ocpcf-pcf-ueservice	44.94%	45.22%	0.05%	14.07%
ocpcf-performance/perf-info	3.10%	10.84%	1.40%	11.04%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-101 CPU/Memory Utilization by CnDBTier services

App/Container	CPU (Site1)	Memory (Site1)	CPU (Site2)	Memory (Site2)
ndbappmysqld/istio-proxy	0.40%	2.00%	0.33%	2.22%

Table 3-101 (Cont.) CPU/Memory Utilization by CnDBTier services

App/Container	CPU (Site1)	Memory (Site1)	CPU (Site2)	Memory (Site2)
ndbappmysqld/ mysqlndbcluster	0.19%	20.91%	0.20%	20.88%
ndbappmysqld/init- sidecar	2.08%	0.39%	2.17%	0.39%
ndbmgmd/istio- proxy	0.55%	9.96%	0.68%	10.79%
ndbmgmd/ mysqlndbcluster	0.37%	25.12%	0.40%	25.12%
ndbmtl/istio-proxy	0.66%	1.75%	0.53%	1.39%
ndbmtl/ mysqlndbcluster	0.69%	81.13%	5110.41%	71.33%
ndbmysqld/istio- proxy	0.00%	0.00%	0.00%	0.00%
ndbmysqld/ mysqlndbcluster	0.52%	26.07%	0.57%	26.07%
ndbmysqld/init- sidecar	2.33%	0.39%	2.17%	0.39%

3.2.7.3 Results

Average Latency Observations for the AM and UE call flow:

Call Flow	50th Percentile (ms)	95th Percentile (ms)
AM CREATE	88.8	176
AM DELETE	84.2	152
UE CREATE	106	204
UE DELETE	5.74	10.4

3.2.8 Test Scenario: PCF AM/UE Call Model on Two-Site Georedundant Setup, with Single-Site Handling 75K TPS Traffic and ASM Enabled

This test run benchmarks the performance and capacity of Policy AM/UE data call model that is deployed in PCF mode. The PCF application handles a total traffic (Ingress + Egress) of 75K TPS on one site and there is no traffic on the other site. Application compression was enabled. For this setup, Aspen Service Mesh (ASM) was enabled between Policy services and it was disabled between Policy service pods and Database data pods.

In this test setup, the Georedundant (GR) mode was enabled in cnDBTier. It was configured for 3 channel replication and the Application Data compression was enabled at AM, UE, and PDS services on Site 2.

3.2.8.1 Test Case and Setup Details

Table 3-102 Testcase Parameters

Parameters	Values
Call Rate (Ingress + Egress)	75k on site-1 and no traffic on site-2
ASM	Enable
Traffic Ratio	AM 1-Create 0-update 1-delete UE 1-Create 0-update 1-delete
Active User Count	12000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match ≤ 3 fields, MatchList < 3 , and $3 < \text{RegEx match} < 6$
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists ≥ 3 , and RegEx mat ≥ 6

Call Model Data

Table 3-103 Traffic distribution on Site1

Services	Ingress Gateway	Egress Gateway	Total Ingress/Egress Traffic
UE service	8340	37530	45870
AM service	8340	20850	29190
Total			75060

Policy Configurations

Following Policy microservices or features were either enabled or disabled for running this call flow:

Table 3-104 Policy microservices or features configuration

Name	Status
Bulwark	Enabled
Binding	Disabled
Local Subscriber State Variable (SSV)	Enabled
Validate_user	Disabled

Table 3-104 (Cont.) Policy microservices or features configuration

Name	Status
Alternate Route	Disabled
Audit	Enabled
Compression (AM, SM, and PDS Service)	Enabled
SYSTEM.COLLISION.DETECTION	Enabled
CHF Async	Enabled
Session Limiting	Enabled
Collision Detection	Enabled
Pending Transaction for Bulwark	Enabled
Preferential Search	SUPI

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-105 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	Enabled
N36 UDR subscription (N7/N15-Nudr)	Enabled
UDR on-demand nrf discovery	Enabled
CHF (SM-Nchf)	Enabled
BSF (N7-Nbsf)	Disabled
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Subscriber HTTP Notifier (Gx)	NA

The following PCRF interfaces that were either enabled or disabled to run this call flow:

Table 3-106 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring Policy Helm Parameters

The following Policy optimization parameters were configured for this run:

Table 3-107 Optimization parameters for Policy services

Service	Policy Helm Configurations
policyds	<pre>- name: DATASOURCE_HIKARI_MIN_IDLE value: "90" - name: DATASOURCE_HIKARI_MAX_POOL_SIZE value: "90" - name: DEFAULT_BOUNDED_ELASTIC_QUEUE_SIZE value: "80"</pre>
UE	DB_MAX_POOL_SIZE=60
INGRESS	<pre>applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500</pre>
EGRESS	<pre>applicationThreadPoolConfig: corePoolSize: 20 maxPoolSize: 20 queueCapacity: 7500</pre>

Configuring cnDbTier Helm Parameters

The following cnDBTier optimization parameters were configured for this run:

Table 3-108 Optimization parameters for CnDBTier services

Helm Parameter	Value	CnDBTier Helm Configuration
ConnectCheckIntervalDelay	0	<pre>additionalIndbconfigurations: ndb: ConnectCheckIntervalDelay: 0</pre>

Table 3-108 (Cont.) Optimization paramters for CnDBTier services

Helm Parameter	Value	CnDBTier Helm Configuration
NoOfFragmentLogParts	6	<pre> ndbconfigurations: ndb: NoOfFragmentLogParts: 6 </pre>
MaxNoOfExecutionThreads	14	<pre> ndbconfigurations: ndb: MaxNoOfExecutionThreads: 14 </pre>
FragmentLogFileSize	128M	<pre> additionalndbconfigurations: ndb: FragmentLogFileSize: 128M </pre>
NoOfFragmentLogFiles	96	<pre> ndbconfigurations: ndb: NoOfFragmentLogFiles: 96 </pre>
binlogthreadstore.capacity	5	<pre> db-monitor-svc: binlogthreadstore: capacity: 5 </pre>
ndb_allow_copying_alter_table	1	<pre> additionalndbconfigurations: mysqld: # use replica_skip_errors as slave-skip-errors/ slave_skip_errors is deprecated ndb_allow_copying_alter_ table: 1 </pre>

Table 3-108 (Cont.) Optimization paramters for CnDBTier services

Helm Parameter	Value	CnDBTier Helm Configuration
binlog_cache_size	10485760	additionalndbconfigurations: mysqld: binlog_cache_size: '10485760'
maxnumberofconcurrentscans	495	maxnumberofconcurrentscans: 495
db_eventbuffer_max_alloc	12G	ndbmysqld db_eventbuffer_max_alloc: 12G
HeartbeatIntervalDbDb	1250	ndbmttd HeartbeatIntervalDbDb: 1250

Policy Microservices Resources

Note

Changes in the resource requirements are highlighted in bold.

Table 3-109 Policy microservices resource allocation for site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
Appinfo	1	1	1 Gi	512Mi	2	2	2 Gi
Audit Service	2	2	4 Gi	4 Gi	2	2	2 Gi
CM Service	4	4	2 Gi	2 Gi	2	2	2 Gi

Table 3-109 (Cont.) Policy microservices resource allocation for site1

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory (Gi)
Config Service	2	2	2Gi	2Gi	2	2	2 Gi
Egress Gateway	27	27	6Gi	6Gi	27	4	2 Gi
Ingress Gateway	8	8	6Gi	6Gi	8	2.5	2 Gi
NRF Client NF Discovery	9	9	10Gi	10Gi	9	2	2 Gi
NRF Client Management	2	2	1Gi	1Gi	2	2	2 Gi
AM Service	12	12	8Gi	8Gi	12	3	2 Gi
UE Service	20	20	6Gi	6Gi	20	2	1 Gi
Query Service	2	2	1Gi	1Gi	2		
Performance	1	1	1Gi	512Mi	2	2	1 Gi
PRE	7	7	4Gi	4Gi	7	1.5	2 Gi
SM Service	1	1	1Gi	1Gi	1	3	2 Gi
PDS	24	24	8Gi	8Gi	24	3	4 Gi
UDR Connector	20	20	4Gi	4Gi	20	2	2 Gi
CHF Connector/ User Service	8	8	4Gi	4Gi	8	2	2 Gi
Alternate Route Service	1	1	4Gi	2Gi	1	2	2 Gi
Bulwark Service	8	8	4Gi	4Gi	7	3	4 Gi

Table 3-110 CnDBTier resource allocation on Site-2

Service Name	CPU Resource per Container (Limit)	CPU Resource per Container (Request)	Memory Resource per Container (Limit)	Memory Resource per Container (Request)	Replica Count	Request/ Limit Istio CPU	Request/ Limit Istio Memory
ndbappmysqld	12	12	20Gi	20Gi	12	5	5Gi
ndbmgmd	2	2	8Gi	8Gi	2	3	1Gi
ndbmtid	10	10	129Gi	129Gi	10	6	6Gi
ndbmysqld	6	6	16Gi	16Gi	6	5	5Gi

3.2.8.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-111 Utilization by Policy Microservices

Service	CPU (X/Y) - Site1
ocpcf-occnp-alternate route	0%/80%
ocpcf-appinfo	1%/80%
ocpcf-bulwark	46%/60%
ocpcf-config-server	12%/80%
ocpcf-ingress-gateway	48%/80%
ocpcf-egress-gateway	45%/80%
ocpcf-nrf-client-nfdiscovery	31%/80%
ocpcf-nrf-client-nfmanagment	0%/80%
ocpcf-occnp-chf-connector	17%/50%
ocpcf-occnp-udr-connector	35%/50%
ocpcf-occpm-audit-service	0%/60%
ocpcf-occpm-policyds	43%/60%
ocpcf-amservice	26%/30%
ocpcf-pcf-pre	26%/80%
ocpcf-pcf-smsservice	0%/50%
ocpcf-pcf-ueservice	58%/30%
ocpcf-ocpm-queryservice	0%/80%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-112 Utilization by CnDBTier services

App/Container	CPU (X/Y) - Site2
ndbappmysql	31%/80%
ndbmcmd	0%/80%
ndbmtd	58%/80%
ndbmmysql	5%/80%

3.2.8.3 Results

Average Latency Observations for the AM and UE call flow:

Table 3-113 Latency Observations

Call Flow	50th Percentile (ms)	95th Percentile (ms)
AM CREATE	44.1	43.3
AM DELETE	41.3	38.8
UE CREATE	67.5	65.0
UE DELETE	11.7	10.3

3.3 Policy Call Model 3

3.3.1 Test Scenario: Policy Voice Call Model on Four-Site Georedundant Setup, with 7.5K TPS Traffic on Each Site and ASM Disabled

This test run benchmarks the performance and capacity of Policy voice call model that is deployed in converged mode on a four-site georedundant setup. Each of the sites handles a traffic of 7.5K TPS at Diameter Gateway. For this setup, Policy Event Record (PER) and Binding feature were enabled and Aspen Service Mesh (ASM) was disabled. This setup has single-channel replication.

3.3.1.1 Test Case and Setup Details

Test Case Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Diameter Gateway)	30K TPS (7.5KTPS on four site)
ASM	Disable
Traffic Ratio	CCRI-I, AARI -1, CCRU-2, AARU - 1, RAR-Gx-1, RAR-Rx-1, STR -1, CCRT-1.
Active Subscribers	10000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6

- High – JSON Operations – Custom, complex Java script code in object Expression/ statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model Data

Service Name	TPS
Ingress Service	NA
Egress Service	NA
Diameter Gateway	7.5K TPS
Diameter Connector	NA
SM service	NA
PDS Service	NA
PRE Service	NA
NRF Discovery	NA
UDR Connector	NA
CHF Connector	NA
Binding Service	NA
Bulwark Service	NA

Policy Configurations

Following PCF configurations were either enabled or disabled for running this call flow:

Table 3-114 Policy Configurations

Service Name	Status
Binding	Enabled
PRE	Enabled
SAL	Enabled
LDAP	Disabled
OCS	Disabled
Audit	Enabled
Replication	Enabled
Bulwark	Disabled
Alternate routing	Disabled

Following Policy Interfaces were either enabled or disabled for running this call flow:

Table 3-115 Policy Interfaces

Feature Name	Status
AMF on demand nrf discovery	NA
BSF (N7-Nbsf)	NA
CHF (SM-Nchf)	NA
LDAP (Gx-LDAP)	NA
N36 UDR query (N7/N15-Nudr)	NA
N36 UDR subscription (N7/N15-Nudr)	NA

Table 3-115 (Cont.) Policy Interfaces

Feature Name	Status
Sy (PCF N7-Sy)	NA
UDR on-demand nrf discovery	NA

Following PCRF interfaces were either enabled or disabled for running this call flow:

Table 3-116 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring Policy Helm Parameters

There are no optimization parameters configured for this run.

Configuring cnDbTier Helm Parameters

There are no optimization parameters configured for this run.

Policy Microservices Resources

Table 3-117 Policy microservices Resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
Appinfo	1	2	1	2	1
Audit Service	1	2	1	1	1
CM Service	2	4	0.5	2	1
Config Service	2	4	0.5	2	1
Egress Gateway	5	5	6	6	2
Ingress Gateway	3	4	4	6	2
Nrf Client Management	1	1	1	1	2
Diameter Gateway	3	4	1	2	9
Diameter Connector	3	4	1	2	5
Nrf Client Discovery	3	4	0.5	2	2
Query Service	1	2	1	1	1

Table 3-117 (Cont.) Policy microservices Resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
PCRF Core Service	7	8	8	8	24
Performance	1	1	0.5	1	2
PRE Service	4	4	0.5	4	15
SM Service	7	7	10	10	2
PDS	7	7	8	8	5
Binding Service	5	6	1	8	18

Table 3-118 cnDBTier services resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Storage
ndbappmysqld	8	8	19	20	5	32Gi
ndbmcmd	2	2	9	11	2	16Gi
ndbmtid	8	8	73	83	8	76Gi
ndbmysqld	4	4	25	25	6	131Gi

Note: Min Replica = Max Replica

3.3.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-119 CPU/Memory Utilization by Policy Microservices

Service Name	Site 1 CPU (X/Y)	Site 2 CPU (X/Y)	Site 3 CPU (X/Y)	Site 4 CPU (X/Y)
ocpcf-appinfo-hpa-v2	3%/80%	3%/80%	3%/80%	3%/80%
ocpcf-config-server-hpa-v2	8%/80%	9%/80%	7%/80%	7%/80%
ocpcf-diam-connector-hpa	0%/40%	0%/40%	0%/40%	0%/40%

Table 3-119 (Cont.) CPU/Memory Utilization by Policy Microservices

Service Name	Site 1 CPU (X/Y)	Site 2 CPU (X/Y)	Site 3 CPU (X/Y)	Site 4 CPU (X/Y)
ocpcf-egress-gateway-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-ingress-gateway-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfdiscovery-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfmanagement-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-oc-binding-hpa	6%/60%	6%/60%	6%/60%	6%/60%
ocpcf-ocpm-audit-service-hpa-v2	4%/60%	1%/60%	1%/60%	1%/60%
ocpcf-ocpm-policyds-hpa	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-pcf-pre-hpa	17%/80%	18%/80%	17%/80%	17%/80%
ocpcf-pcrf-core-hpa	12%/40%	12%/40%	12%/40%	12%/40%
ocpcf-query-service-hpa	0%/80%	0%/80%	0%/80%	0%/80%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-120 CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ndbappmysqld	88%/80%	87%/80%	89%/80%	88%/80%
ndbmcmd	0%/80%	0%/80%	0%/80%	0%/80%
ndbmtl	16%/80%	17%/80%	17%/80%	18%/80%
ndbmysqld	8%/80%	9%/80%	10%/80%	8%/80%

3.3.1.3 Results

Table 3-121 Average PCRF Core JDBC Latency Observations

Site 1	Site 2	Site 3	Site 4
2.19 ms	2.32 ms	2.66 ms	2.56 ms

3.3.2 Test Scenario: Policy Voice Call Model on Four-Site Georedundant Setup, with 15K TPS Traffic on Two Sites and No Traffic on Other Two Sites

This test run benchmarks the performance and capacity of Policy voice call model that is deployed in converged mode on a four-site georedundant setup. Two of the sites (site1 and site3) handle a traffic of 15K TPS at Diameter Gateway and there is no traffic on the other two sites (site2 and site4). For this setup, Binding and Policy Event Record (PER) features were

enabled and Aspen Service Mesh (ASM) was disabled. This setup has single-channel replication.

3.3.2.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Diameter Gateway)	30KTPS (15KTPS on two sites)
ASM	Disable
Traffic Ratio	CCRI-I, AARI –1, CCRU-2, AARU - 1, RAR-Gx-1, RAR-Rx-1, STR –1, CCRT-1.
Active Subscribers	10000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model Data

Services	TPS
Ingress Service	NA
Egress Service	NA
Diameter Gateway	15K TPS
Diameter Connector	NA
SM service	NA
PDS Service	NA
PRE Service	NA
NRF Discovery	NA
UDR Connector	NA
CHF Connector	NA
Binding Service	NA
Bulwark Service	NA

Policy Configurations

Following Policy configurations were either enabled or disabled for running this call flow:

Table 3-122 Policy Microservices Configuration

Service Name	Status
Binding	Enabled
PER	Enabled
SAL	Enabled
LDAP	Disabled
OCS	Disabled
Audit	Enabled
Replication	Enabled
Bulkwark	Disabled
Alternate routing	Disabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-123 Policy Interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	NA
N36 UDR subscription (N7/N15-Nudr)	NA
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	NA
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA

Following PCRF interfaces were either enabled or disabled for running this call flow:

Table 3-124 PCRF Interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring Policy Helm Parameters

There were no optimized parameters configured for this run.

Configuring cnDbTier Helm Parameters

There were no optimized parameters configured for this run.

Policy Microservices Resources

Table 3-125 Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
Appinfo	1	1	0.5	1	1
Audit Service	1	2	1	1	1
CM Service	2	4	0.5	2	1
Config Service	2	4	0.5	2	1
Egress Gateway	3	4	4	6	2
Ingress Gateway	3	4	4	6	2
Nrf Client Management	1	1	1	1	2
Diameter Gateway	3	4	1	2	9
Diameter Connector	3	4	1	2	5
Nrf Client Discovery	3	4	0.5	2	2
Query Service	1	2	1	1	1
PCRF Core Service	7	8	8	8	24
Performance	1	1	0.5	1	2
PRE Service	5	5	0.5	4	15
SM Service	7	8	1	4	2
PDS	5	6	1	4	5
Binding Service	5	6	1	8	18

Table 3-126 cnDBTier services resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Storage
ndbappmysqld	8	8	19	20	5	32Gi
ndbmcmd	2	2	9	11	2	16Gi
ndbmttd	8	8	73	83	8	76Gi
ndbmysqld	4	4	25	25	6	131Gi

Min Replica = Max Replica

3.3.2.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-127 CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ocpcf-appinfo-hpa-v2beta1	2%/80%	2%/80%	3%/80%	2%/80%
ocpcf-config-server-hpa-v2beta1	7%/80%	9%/80%	9%/80%	8%/80%
ocpcf-diam-connector-hpa-v2beta1	0%/40%	0%/40%	0%/40%	0%/40%
ocpcf-egress-gateway-v2beta1	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-ingress-gateway-v2beta1	1%/80%	0%/80%	1%/80%	0%/80%
ocpcf-nrf-client-nfdiscovery-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfmanagement-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-oc-binding-hpa	11%/60%	0%/60%	11%/60%	0%/60%
ocpcf-ocpm-audit-service-hpa-v2beta1	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-ocpm-policyds-hpa	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-pcf-pre-hpa	10%/80%	0%/80%	10%/80%	0%/80%
ocpcf-pcf-smsservice-hpa	0%/50%	0%/50%	0%/50%	0%/50%
ocpcf-pcrf-core-hpa	25%/40%	0%/80%	24%/40%	0%/40%
ocpcf-query-service-hpa	0%/80%	0%/40%	0%/80%	0%/80%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-128 CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ndbappmysqld	73%/80%	23%/80%	89%/80%	23%/80%
ndbmgmd	0%/80%	0%/80%	0%/80%	0%/80%
ndbmtl	22530%/80%	7280%/80%	16%/80%	7%/80%
ndbmysqld	8%/80%	4%/80%	8%/80%	4%/80%

3.3.2.3 Results

Table 3-129 Average PCRF Core JDBC Latency Observations

Site 1	Site 2	Site 3	Site 4
2.62 ms	-	4.28 ms	-

3.4 Policy Call Model 4

3.4.1 Test Scenario: Policy Call Model on Four-Site Georedundant Setup, with 7.5K TPS Traffic on Each Site and ASM Disabled

This test run benchmarks the performance and capacity of Policy data call model that is deployed in converged mode on a four-site georedundant setup. Each of the sites handles a traffic of 7.5K TPS at Diameter Gateway. For this setup, Binding feature was enabled and Aspen Service Mesh (ASM) was disabled. This setup has single-channel replication.

3.4.1.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Diameter Gateway)	30KTPS (7.5KTPS on each site)
ASM	Disable
Traffic Ratio	CCRI (Single APN), CCRU (Single APN), CCRT (Single APN), AARU, RAR -rx, RAR-gx, STR.
Active Subscribers	10000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match ≤ 3 fields, MatchList < 3 , and $3 < \text{RegEx match} < 6$
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists ≥ 3 , and RegEx mat ≥ 6

Call Model Data

Table 3-130 Traffic distribution

Service Name	TPS
Ingress Service	NA
Egress Service	NA
Diameter Gateway	7.5K TPS
Diameter Connector	NA
SM service	NA
PDS Service	NA
PRE Service	NA
NRF Discovery	NA
UDR Connector	NA
CHF Connector	NA
Binding Service	NA
Bulwark Service	NA

Policy Configurations

Following Policy services were either enabled or disabled for running this call flow:

Table 3-131 Policy services configuration

Service Name	Status
Binding	Enabled
PER	Disabled
SAL	Enabled
LDAP	Disabled
OCS	Disabled
Audit	Enabled
Replication	Enabled
Bulkwark	Disabled
Alternate routing	Disabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-132 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	NA
N36 UDR subscription (N7/N15-Nudr)	NA
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	NA
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA

Table 3-132 (Cont.) Policy interfaces

Feature Name	Status
Sy (PCF N7-Sy)	NA
Diameter GW (PGW to PCRF)	Active

Following PCRF interfaces were either enabled or disabled for running this call flow:

Table 3-133 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring PCF Helm Parameters

There were no optimization parameters configured for this run.

Configuring cnDbTier Helm Parameters

There were no optimization parameters configured for this run.

Policy Microservices Resources

Table 3-134 Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
Appinfo	1	1	0.5	1	1
Audit Service	1	2	1	1	1
CM Service	2	4	0.5	2	1
Config Service	2	4	0.5	2	1
Egress Gateway	3	4	4	6	2
Ingress Gateway	3	4	4	6	2
Nrf Client Management	1	1	1	1	2
Diameter Gateway	3	4	1	2	9
Diameter Connector	3	4	1	2	5
Nrf Client Discovery	3	4	0.5	2	2
Query Service	1	2	1	1	1

Table 3-134 (Cont.) Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
PCRF Core Service	7	8	8	8	24
Performance	1	1	0.5	1	2
PRE Service	5	5	0.5	4	15
SM Service	7	8	1	4	2
PDS	5	6	1	4	5
Binding Service	5	6	1	8	18

Table 3-135 cnDBTier services resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Storage
ndbappmysqld	8	8	19	20	5	32Gi
ndbmcmd	2	2	9	11	2	16Gi
ndbmtid	8	8	73	83	8	76Gi
ndbmysqld	4	4	25	25	6	131Gi

Min Replica = Max Replica

3.4.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-136 CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ocpcf-appinfo-hpa-v2beta1	1%/80%	2%/80%	2%/80%	1%/80%
ocpcf-config-server-hpa-v2beta1	8%/80%	9%/80%	8%/80%	7%/80%
ocpcf-diam-connector-hpa-v2beta1	0%/40%	0%/40%	0%/40%	0%/40%

Table 3-136 (Cont.) CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ocpcf-egress-gateway-v2beta1	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-ingress-gateway-v2beta1	1%/80%	1%/80%	1%/80%	1%/80%
ocpcf-nrf-client-nfdiscovery-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfmanagement-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-oc-binding-hpa	6%/60%	6%/60%	6%/60%	6%/60%
ocpcf-ocpm-audit-service-hpa-v2beta1	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-ocpm-policyds-hpa	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-pcf-pre-hpa	6%/80%	6%/80%	6%/80%	6%/80%
ocpcf-pcf-smsservice-hpa	0%/50%	0%/50%	0%/50%	0%/50%
ocpcf-pcrf-core-hpa	13%/40%	0%/80%	14%/40%	14%/40%
ocpcf-query-service-hpa	0%/80%	13%/40%	0%/80%	0%/80%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-137 CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ndbappmysqld	71%/80%	84%/80%	84%/80%	85%/80%
ndbmcmd	0%/80%	0%/80%	0%/80%	1%/80%
ndbmtl	14%/80%	11%/80%	16%/80%	15%/80%
ndbmysqld	12%/80%	12%/80%	13%/80%	12%/80%

3.4.1.3 Results

Table 3-138 Average PCRF Core JDBC Latency Observations

Site 1	Site 2	Site 3	Site 4
2.30 ms	2.20 ms	2.66 ms	2.85 ms

3.4.2 Test Scenario: Policy Call Model on Four-Site Georedundant Setup, with 15K TPS Traffic on Two Sites and No Traffic on Other Two Sites

This test run benchmarks the performance and capacity of Policy data call model that is deployed in converged mode on a four-site georedundant setup. Two of the sites (site1 and

site3) handle a traffic of 15K TPS at Diameter Gateway and there is no traffic on the other two sites (site2 and site4). For this setup, Binding feature was enabled and Aspen Service Mesh (ASM) was disabled. This setup has single-channel replication.

3.4.2.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Diameter Gateway)	30KTPS (15 KTPS on two site)
ASM	Disable
Traffic Ratio	CCRI (Single APN), CCRU (Single APN), CCRT (Single APN), AARU, RAR -rx, RAR-gx, STR.
Active Subscribers	10000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match ≤ 3 fields, MatchList < 3 , and $3 < \text{RegEx match} < 6$
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists ≥ 3 , and RegEx mat ≥ 6

Call Model Data

Table 3-139 Traffic distribution

Service Name	TPS
Ingress Service	NA
Egress Service	NA
Diameter Gateway	15K TPS
Diameter Connector	NA
SM service	NA
PDS Service	NA
PRE Service	NA
NRF Discovery	NA
UDR Connector	NA
CHF Connector	NA
Binding Service	NA
Bulwark Service	NA

Policy Configurations

Following Policy services were either enabled or disabled for running this call flow:

Table 3-140 Policy microservices configuration

Service Name	Status
Binding	Enabled
PER	Disabled
SAL	Enabled
LDAP	Disabled
OCS	Disabled
Audit	Enabled
Replication	Enabled
Bulkwark	Disabled
Alternate routing	Disabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-141 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	NA
N36 UDR subscription (N7/N15-Nudr)	NA
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	NA
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA
Diameter (PGW to PCRF)	Active

Following PCRF interfaces were either enabled or disabled for running this call flow:

Table 3-142 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA
Diameter (PGW to PCRF)	Active

Configuring cnDbTier Helm Parameters

There were no optimization parameters configured for this run.

Configuring cnDbTier Helm Parameters

There were no optimization parameters configured for this run.

Policy Microservices Resources

Table 3-143 Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
Appinfo	1	1	0.5	1	1
Audit Service	1	2	1	1	1
CM Service	2	4	0.5	2	1
Config Service	2	4	0.5	2	1
Egress Gateway	3	4	4	6	2
Ingress Gateway	3	4	4	6	2
Nrf Client Management	1	1	1	1	2
Diameter Gateway	3	4	1	2	9
Diameter Connector	3	4	1	2	5
Nrf Client Discovery	3	4	0.5	2	2
Query Service	1	2	1	1	1
PCRF Core Service	7	8	8	8	24
Performance	1	1	0.5	1	2
PRE Service	5	5	0.5	4	15
SM Service	7	8	1	4	2
PDS	5	6	1	4	5
Binding Service	5	6	1	8	18

Table 3-144 cnDBTier services resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Storage
ndbapppmysqld	8	8	19	20	5	32Gi
ndbmcmd	2	2	9	11	2	16Gi
ndbmtl	8	8	73	83	8	76Gi
ndbmysqld	4	4	25	25	6	131Gi

Min Replica = Max Replica

3.4.2.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-145 CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ocpcf-appinfo-hpa-v2	3%/80%	3%/80%	4%/80%	3%/80%
ocpcf-config-server-hpa-v2	8%/80%	8%/80%	7%/80%	7%/80%
ocpcf-diam-connector-hpa	0%/40%	0%/40%	0%/40%	0%/40%
ocpcf-egress-gateway-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-ingress-gateway-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfdiscovery-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-nrf-client-nfmanagement-v2	0%/80%	0%/80%	0%/80%	0%/80%
ocpcf-oc-binding-hpa	11%/60%	0%/60%	12%/60%	0%/60%
ocpcf-ocpm-audit-service-hpa-v2	4%/60%	4%/60%	3%/60%	4%/60%
ocpcf-ocpm-policyds-hpa	0%/60%	0%/60%	0%/60%	0%/60%
ocpcf-pcf-pre-hpa	37%/80%	0%/80%	37%/80%	0%/80%
ocpcf-pcrf-core-hpa	24%/40%	0%/40%	24%/40%	0%/40%
ocpcf-query-service-hpa	0%/80%	0%/80%	0%/80%	0%/80%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-146 CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ndbappmysqld	91%/80%	87%/80%	92%/80%	88%/80%
ndbmcmd	0%/80%	0%/80%	0%/80%	0%/80%

Table 3-146 (Cont.) CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)	Site3 - CPU (X/Y)	Site4 - CPU (X/Y)
ndbmtl	23%/80%	8%/80%	20%/80%	11%/80%
ndbmysqld	12%/80%	6%/80%	12%/80%	6%/80%

3.4.2.3 Results

Table 3-147 Average PCRF Core JDBC Latency Observations

Site 1	Site 2	Site 3	Site 4
2.66 ms	1.26 ms	4.14 ms	1.74 ms

3.4.3 Test Scenario: Policy Call Model on Two-Site Georedundant Setup, with 15K TPS Traffic on Two Sites

This test run benchmarks the performance and capacity of Policy data call model that is deployed in PCF mode on a two-site of a two-site non-ASM GR Setup. Replication is on single-channel and Binding and PRE Enabled. The Policy application handles a total Ingress and Egress traffic of 15K TPS on two sites.

3.4.3.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Diameter Gateway)	30KTPS (15K TPS on two site)
ASM	Disable
Traffic Ratio	CCRI-I, AARI -1, CCRU-2, AARU - 1, RAR-Gx-1, RAR-Rx-1, STR -1, CCRT-1
Active Subscribers	10000000

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model Data

Table 3-148 Traffic distribution

Service Name	TPS
Ingress Service	NA
Egress Service	NA
Diameter Gateway Ingress	8.33K TPS
Diameter Gateway Egress	6.31K TPS
Diameter Connector	NA
SM service	NA
PDS Service	NA
PRE Service	NA
NRF Discovery	NA
UDR Connector	NA
CHF Connector	NA
Binding Service	NA
Bulwark Service	NA

Policy Configurations

Following Policy services were either enabled or disabled for running this call flow:

Table 3-149 Policy microservices configuration

Service Name	Status
Binding	Enabled
PER	Enabled
SAL	Enabled
LDAP	Disabled
OCS	Disabled
Audit	Enabled
Replication	Enabled
Bulwark	Disabled
Alternate routing	Disabled

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-150 Policy interfaces

Feature Name	Status
N36 UDR query (N7/N15-Nudr)	NA
N36 UDR subscription (N7/N15-Nudr)	NA
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	NA
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA

Table 3-150 (Cont.) Policy interfaces

Feature Name	Status
LDAP (Gx-LDAP)	NA
Sy (PCF N7-Sy)	NA

Following PCRF interfaces were either enabled or disabled for running this call flow:

Table 3-151 PCRF interfaces

Feature Name	Status
Sy (PCRF Gx-Sy)	NA
Sd (Gx-Sd)	NA
Gx UDR query (Gx-Nudr)	NA
Gx UDR subscription (Gx-Nudr)	NA
CHF enabled (AM)	NA
Usage Monitoring (Gx)	NA
Subscriber HTTP Notifier (Gx)	NA

Configuring cnDbTier Helm Parameters

There were no optimization parameters configured for this run.

Configuring cnDbTier Helm Parameters

There were no optimization parameters configured for this run.

Policy Microservices Resources

Table 3-152 Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
Appinfo	1	1	0.5	1	1
Audit Service	1	2	1	1	1
CM Service	2	4	0.5	2	1
Config Service	2	4	0.5	2	1
Egress Gateway	3	4	4	6	2
Ingress Gateway	3	4	4	6	2
Nrf Client Management	1	1	1	1	2
Diameter Gateway	3	4	1	2	9
Diameter Connector	3	4	1	2	5
Nrf Client Discovery	3	4	0.5	2	2
Query Service	1	2	1	1	1

Table 3-152 (Cont.) Policy microservices resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas
PCRF Core Service	7	8	8	8	24
Performance	1	1	0.5	1	2
PRE Service	5	5	0.5	4	15
SM Service	7	8	1	4	2
PDS	5	6	1	4	5
Binding Service	5	6	1	8	18

Table 3-153 cnDBTier services resource allocation

Service Name	CPU Request Per Pod	CPU Limit Per Pod	Memory Request Per Pod (Gi)	Memory Limit Per Pod (Gi)	Replicas	Storage
ndbappmysqld	8	8	19	20	5	32Gi
ndbmcmd	2	2	9	11	2	16Gi
ndbmtl	8	8	73	83	8	76Gi
ndbmcsqld	4	4	25	25	6	131Gi

Min Replica = Max Replica

3.4.3.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-154 CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)
ocpcf-appinfo-hpa-v2	4%/80%	5%/80%
ocpcf-config-server-hpa-v2	8%/80%	8%/80%
ocpcf-diam-connector-hpa	0%/40%	0%/40%
ocpcf-egress-gateway-v2	0%/80%	0%/80%
ocpcf-ingress-gateway-v2	0%/80%	0%/80%
ocpcf-nrf-client-nfdiscovery-v2	0%/80%	0%/80%

Table 3-154 (Cont.) CPU/Memory Utilization by Policy Microservices

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)
ocpcf-nrf-client-nfmanagement-v2	0%/80%	0%/80%
ocpcf-oc-binding-hpa	8%/60%	8%/60%
Diam-Gw (from dashboard)	2.5%/80%	2.5%/80%
ocpcf-ocpm-audit-service-hpa-v2	4%/60%	4%/60%
ocpcf-ocpm-policyds-hpa	0%/60%	0%/60%
ocpcf-pcf-pre-hpa	40%/80%	42%/80%
ocpcf-pcrf-core-hpa	25%/40%	24%/40%
ocpcf-query-service-hpa	0%/80%	0%/40%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-155 CPU/Memory Utilization by CnDBTier services

Service Name	Site1 - CPU (X/Y)	Site2 - CPU (X/Y)
ndbappmysqld	85%/80%	92%/80%
ndbmcmd	0%/80%	0%/80%
ndbmtl	15%/80%	15%/80%
ndbmysqld	6%/80%	6%/80%

3.4.3.3 Results

Table 3-156 Average PCRF Core JDBC Latency Observations

Site 1	Site 2
2.07 ms	2.03 ms

3.5 PCF Call Model 5

3.5.1 Test Scenario: PCF Call Model on Single-Site Setup, Handling 30K TPS Traffic with Binding Feature Enabled

This test was run to benchmark the performance and capacity of PCF call model with 30K traffic on a single site. For this setup, Aspen Service Mesh (ASM) was disabled, Binding feature was enabled. User Connector microservice restart with a duration of 4.0 hours.

3.5.1.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Ingress + Egress)	30K TPS on a single site Non ASM PCF Setup
ASM	Disable
Traffic Ratio	IGW-11,EGW-26,Diam-in 9,Diam-Out 3IGW-11 ,EGW-26,Diam-in=9,Diam-out - 3
Deployment Model	PCF 1 at Site1

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model

Table 3-157 Traffic distribution

Traffic	TPS
Ingress Gateway	6637
Egress Gateway	15988
Diam In	5279
Diam out	1844
Total	29747

Table 3-158 Traffic distribution to Policy databases

Number of Entries	TPS
occnp_pcf_sm.AppSession	132704
occnp_pcf_sm.SmPolicyAssociation	434302
occnp_pcf_sm.SmPolicyAssociation\$EX	0
occnp_policyds.pdssubscriber	434475
occnp_policyds.pdssubscriber\$EX	0
occnp_policyds.pdsprofile	324110
occnp_policyds.pdsprofile\$EX	0
occnp_binding.contextbinding	434668
occnp_binding.contextbinding\$EX	0
occnp_binding.dependentcontextbinding	77294
occnp_binding.dependentcontextbinding\$EX	0

Table 3-159 Traffic distribution at Policy services

Policy Service	Avg TPS/MPS
Ingress Gateway(MPS)	12075.40103
Egress Gateway(MPS)	28537.36981
SM Service(MPS)	44669.88753
AM Service(MPS)	0.00000
UE Service(MPS)	0.00000
PDS(MPS)	12643.96131
Pre Service(MPS)	0.00000
Nrf Discovery(MPS)	0.00000
CHF Connector(MPS)	6591.08083
UDR Connector(MPS)	0.00000
Binding(MPS)	12064.61603

Policy Configurations

Following PCF configurations were either enabled or disabled for running this call flow:

Table 3-160 Policy configurations

Name	Status
Bulwark	Disabled
Binding	Enabled
Subscriber State Variable (SSV)	Disabled
Validate_user	Enabled
Alternate Route	Enabled
Audit	Enabled
Compression (Binding & SM Service)	Disabled
SYSTEM.COLLISION.DETECTION	Disabled

Policy Interfaces

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-161 Policy interfaces

Feature Name	Status
Subscriber Tracing[For 100 subscriber]	Enabled
N36 UDR subscription (N7/N15-Nudr)	Enabled
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	Enabled
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Binding Feature	Enabled

Policy Microservices Resources

Table 3-162 Policy microservices Resource allocation

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
Appinfo	2	1	1	0.5	1
Binding Service	2	6	6	1	8
Diameter Connector	4	4	4	1	2
Diameter Gateway	2	4	4	1	2
Audit Service	1	1	2	1	1
CM Service	1	4	4	0.5	2
Config Service	1	4	4	0.5	2
Egress Gateway	8	4	4	4	6
Ingress Gateway	8	4	4	4	6
NRF Client NF Discovery	1	4	4	0.5	2
NRF Client Management	1	1	1	1	1
Query Service	1	1	2	1	1
PRE	13	4	4	0.5	2
SM Service	9	8	8	1	4
PDS	8	6	6	1	4
UDR Connector	2	6	6	1	4
CHF Connector/ User Service	2	1	4	6	6

cnDBTier Microservices Resources

Table 3-163 CnDBTier Resource allocation

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
ndbappmysqld	4	12	12	24	24
ndbmcmd	2	4	4	10	10
ndbmtd	8	8	8	42	42
db-infra-monitor-svc	1	200	200	500	500
db-backup-manager-svc	1	100	100	128	128

3.5.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-164 CPU/Memory Utilization by Policy Microservices

App/ Container	CPU	Memory
AppInfo	3.80%	24.71%
Binding Service	24.36%	23.96%
Diameter Connector	29.76%	49.39%
CHF Connector	33.37%	39.40%
Config Service	3.14%	42.07%
Egress Gateway	46.77%	28.76%
Ingress Gateway	53.61%	55.54%
NRF Client NF Discovery	0.07%	31.45%
NRF Client NF Management	0.30%	46.00%
UDR Connector	19.05%	22.53%
Audit Service	0.15%	46.29%
CM Service	0.47%	34.08%
PDS	39.39%	45.96%
PRE Service	19.81%	85.36%
Query Service	0.05%	25.83%
AM Service	0.05%	13.18%
SM Service	57.00%	89.29%
UE Service	0.40%	34.96%
Performance	1.00%	13.18%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-165 CPU/Memory Utilization by CnDBTier services

Service	CPU	Memory
ndbappmysqld/mysqlIndbcluster	60.41%	38.09%
ndbappmysqld/init-sidecar	2.00%	0.39%
ndbmgmd/mysqlIndbcluster	0.18%	20.12%
ndbmgmd/db-infra-monitor-svc	2.00%	9.38%
ndbmtl/mysqlIndbcluster	36.65%	82.12%

Table 3-165 (Cont.) CPU/Memory Utilization by CnDBTier services

Service	CPU	Memory
ndbmtl/db-backup-executor-svc	0.10%	2.31%
ndbmtl/db-infra-monitor-svc	2.37%	9.08%
ocpcf-oc-diam-gateway/diam-gateway	18.56%	35.06%

3.5.1.3 Results

Table 3-166 Average latency observations

Scenario	Average Latency (ms)	Peak Latency (ms)
create-dnn_ims	28.631	28.733
N7-dnn_internet_1st	1527.421	2239.414
N7-dnn_internet_2nd	1518.459	1990.823
N7-dnn_internet_3rd	1567.876	1967.632
delete-dnn_ims	14.595	14.666
Overall	931.397	2239.414

Table 3-167 Average NF service latency

NF Service Latency (In Seconds)	Avg
PCF_IGW_Latency	0.01588
PCF_POLICYPDS_Latency	0.01112
PCF_UDRCONNECTOR_Latency	0.00237
PCF_NRFCLIENT_Latency	0.00000
PCF_EGRESS_Latency	0.00060

3.5.2 Test Scenario: PCF Call Model on Single-Site Setup, Handling 30K TPS Traffic with Binding Feature Disabled

This test was run to benchmark the performance and capacity of PCF call model with 30K traffic on a single site. For this setup, Aspen Service Mesh (ASM) was disabled, Binding feature was disabled.

3.5.2.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Ingress + Egress)	30K TPS on a single site Non ASM PCF Setup
ASM	Disable

Parameters	Values
Traffic Ratio	IGW-11,EGW-26,Diam-in 9,Diam-Out 3 IGW-11 ,EGW-26,Diam-in=9,Diam-out - 3
Deployment Model	PCF 1 at Site1

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match <= 3 fields, MatchList < 3, and 3 < RegEx match < 6
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists >= 3, and RegEx mat >= 6

Call Model

Table 3-168 Traffic distribution

Traffic	TPS
Ingress Gateway	6637
Egress Gateway	15988
Diam In	5279
Diam out	1844
Total	29747

Table 3-169 Traffic distribution to Policy databases

Number of Entries	TPS
occnp_pcf_sm.AppSession	132704
occnp_pcf_sm.SmPolicyAssociation	434302
occnp_pcf_sm.SmPolicyAssociation\$EX	0
occnp_policyds.pdssubscriber	434475
occnp_policyds.pdssubscriber\$EX	0
occnp_policyds.pdsprofile	324110
occnp_policyds.pdsprofile\$EX	0
occnp_binding.contextbinding	434668
occnp_binding.contextbinding\$EX	0
occnp_binding.dependentcontextbinding	77294
occnp_binding.dependentcontextbinding\$EX	0

Table 3-170 Traffic distribution at Policy services

Policy Service	Avg TPS/MPS
Ingress Gateway(MPS)	13294.09
Egress Gateway(MPS)	30644.41
SM Service(MPS)	46777.97
AM Service(MPS)	0.00
UE Service(MPS)	0.00
PDS(MPS)	13115.32
CHF Connector(MPS)	6452.53
UDR Connector(MPS)	3638.04
Binding(MPS)	0.00

Policy Configurations

Following PCF configurations were either enabled or disabled for running this call flow:

Table 3-171 Policy configurations

Name	Status
Bulwark	Disabled
Binding	Disabled
Subscriber State Variable (SSV)	Enabled
Validate_user	Enabled
Alternate Route	Enabled
Audit	Enabled
Compression (Binding & SM Service)	Disabled
SYSTEM.COLLISION.DETECTION	Disabled

Policy Interfaces

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-172 Policy interfaces

Feature Name	Status
Subscriber Tracing[For 100 subscriber]	Enabled
N36 UDR subscription (N7/N15-Nudr)	Enabled
UDR on-demand nrf discovery	NA
CHF (SM-Nchf)	Enabled
BSF (N7-Nbsf)	NA
AMF on demand nrf discovery	NA
LDAP (Gx-LDAP)	NA
Binding Feature	Disabled

Policy Microservices Resources

Table 3-173 Policy microservices Resource allocation

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
Appinfo	2	1	1	0.5	1
Binding Service	2	6	6	8	8
Diameter Connector	4	4	4	1	2
Diameter Gateway	4	4	4	1	2
Audit Service	1	2	2	4	4
CM Service	1	4	4	0.5	2
Config Service	1	4	4	0.5	2
Egress Gateway	8	4	4	6	6
Ingress Gateway	8	4	4	6	6
NRF Client NF Discovery	1	4	4	0.5	2
NRF Client Management	1	1	1	1	1
Query Service	1	2	2	1	1
PRE	13	4	4	4	4
SM Service	9	8	8	6	6
PDS	8	6	6	6	6
UDR Connector	2	6	6	4	4
CHF Connector/ User Service	2	6	6	4	4

cnDBTier Microservices Resources

Table 3-174 CnDBTier Resource allocation

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
ndbappmysqld	4	12	12	28	28
ndbmcmd	2	4	4	9	12
ndbmtl	8	8	8	42	42
db-infra-monitor-svc	1	200	200	500	500
db-backup-manager-svc	1	100	100	128	128

3.5.2.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-175 CPU/Memory Utilization by Policy Microservices

App/ Container	CPU	Memory
AppInfo	4.00%	25.40%
Diameter Connector	39.80%	75.70%
CHF Connector	57.30%	58.90%
Config Service	2.78%	3.60%
Egress Gateway	47.50%	26.90%
Ingress Gateway	53.60%	42.42%
NRF Client NF Discovery	0.102%	33.59%
NRF Client NF Management	0.214%	41.6%
UDR Connector	25.50%	71.90%
Audit Service	0.669%	46.3%
CM Service	0.38%	34.16%
PDS	48.67%	64.20%
PRE Service	15.9%	49.6%
Query Service	0.0357%	25.12%
AM Service	0.02%	14.96%
SM Service	64.60%	76.23%
UE Service	0.387%	34.57%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-176 CPU/Memory Utilization by CnDBTier services

Service	CPU	Memory
ndbappmysqld/mysqlIndbcluster	51.50%	44.70%
ndbmgmd/db-infra-monitor-svc	10.30%	16.90%
ndbmttd/mysqlIndbcluster	35.1%	72.60%
ndbmttd/db-backup-executor-svc	35.1%	2.32%
ndbmttd/db-infra-monitor-svc	35.1%	13.60%

3.5.2.3 Results

Table 3-177 Average latency observations

Scenario	Average Latency (ms)	Peak Latency (ms)
create-dnn_ims	54.142	66.775
N7-dnn_internet_1st	20.316	22.226
N7-dnn_internet_2nd	23.517	26.133
N7-dnn_internet_3rd	20.071	21.323
delete-dnn_ims	29.722	47.689
Overall	29.554	66.775

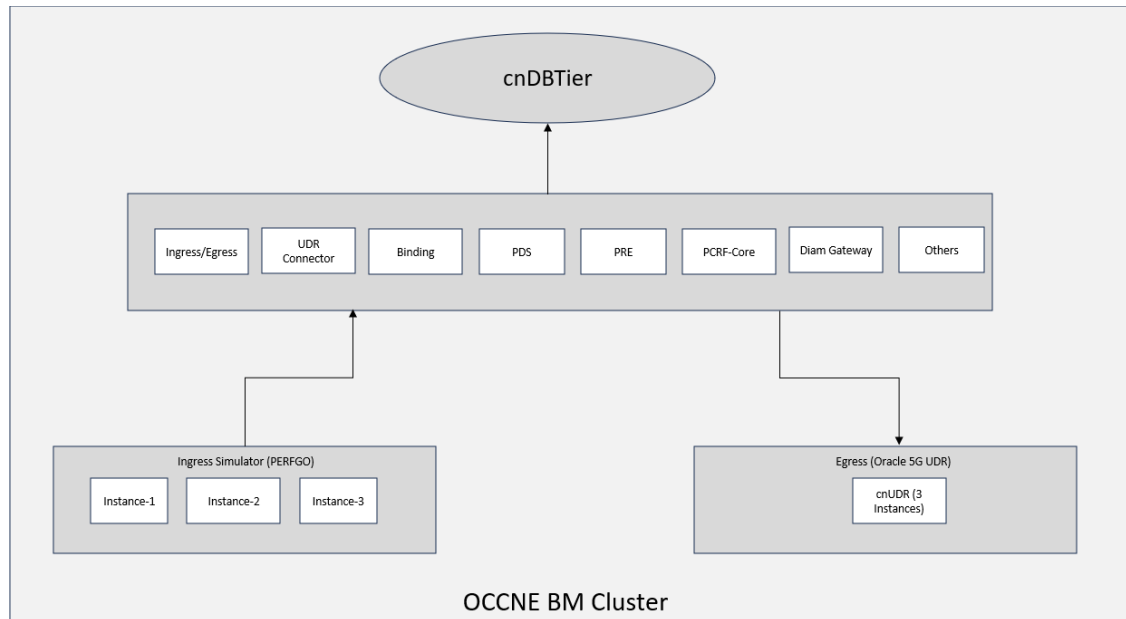
Table 3-178 Average NF service latency

NF Service Latency (In Seconds)	Avg
PCF_IGW_Latency	17.45
PCF_POLICYPDS_Latency	16.85
PCF_UDRCONNECTOR_Latency	2.19
PCF_NRFCLIENT_Latency	0.00
PCF_EGRESS_Latency	0.51

3.6 PCF Call Model 6

3.6.1 Test Scenario: 10K TPS Diameter Ingress Gateway and 17K TPS Egress Gateway TPS Traffic with Usage Monitoring Enabled

This test was run to benchmark the performance and capacity of PCF call model with 10K TPS Diameter Ingress Gateway and 17K TPS Diameter Egress Gateway TPS Traffic with Usage Monitoring Enabled.

Figure 3-2 Policy Deployment in a single site Setup:

3.6.1.1 Test Case and Setup Details

Testcase Parameters

The following table describes the testcase parameters and their values:

Parameters	Values
Call Rate (Ingress + Egress)	27K TPS on a single site Non ASM PCF Setup
ASM	Disable
Traffic Ratio	PCF 10K Diameter Ingress Gateway TPS and 17K Egress Gateway TPS
Deployment Model	PCF as a standalone

Project Details

The Policy Design editor based on the Blockly interface was used to set the Policy project for each of the Policy services. The complexity level of Policy Project configured for this run was **High**.

Complexity Level Definition:

- Low – No usage of loops in Blockly logic, no JSON operations, and no complex Java Script code in object expression/statement expression.
- Medium – Usage of loops in Blockly logic, Policy table wildcard match ≤ 3 fields, MatchList < 3 , and $3 < \text{RegEx match} < 6$
- High – JSON Operations – Custom, complex Java script code in object Expression/statement expression, Policy table wildcard match > 3 fields, MatchLists ≥ 3 , and RegEx mat ≥ 6

Call Model**Table 3-179 Traffic distribution**

Traffic	TPS
Ingress Gateway	1000
Egress Gateway	17000
Diam In	10000
Diam out	0
Total	29747

Table 3-180 Traffic distribution to Policy databases

Number of Entries	TPS
occnp_policyds.pdssubscriber	3084338
occnp_policyds.pdssubscriber\$EX	0
occnp_policyds.pdsprofile	2278801
occnp_policyds.pdsprofile\$EX	0
occnp_binding.contextbinding	82382
occnp_binding.contextbinding\$EX	0
occnp_binding.dependentcontextbinding	0
occnp_binding.dependentcontextbinding\$EX	0
occnp_pcrf_core.gxsession	82351
occnp_pcrf_core.gxsession\$EX	0
occnp_usagemon.UmContext	737281
occnp_usagemon.UmContext\$EX	0

Policy Configurations

Following PCF configurations were either enabled or disabled for running this call flow:

Table 3-181 Policy configurations

Name	Status
Binding	Disabled
Validate_user	Enabled
Usage Monitoring	Enabled
PRE	Enabled

Policy Interfaces

Following Policy interfaces were either enabled or disabled for running this call flow:

Table 3-182 Policy interfaces

Feature Name	Status
N36 UDR subscription (N7/N15-Nudr)	Enabled
UDR on-demand nrf discovery	Disabled

Table 3-182 (Cont.) Policy interfaces

Feature Name	Status
LDAP (Gx-LDAP)	NA
Binding Feature	Disabled

Policy Microservices Resources**Table 3-183 Policy microservices Resource allocation**

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
Appinfo	2	1	1	1	1
Binding Service	10	1	1	1	1
Diameter Connector	4	4	4	2	2
Diameter Gateway	2	4	4	2	2
Config Service	1	4	4	2	2
Egress Gateway	8	4	4	6	6
LDAP Gateway	0	3	4	1	2
Ingress Gateway	8	1	1	1	1
NRF Client NF Discovery	1	1	1	1	1
NRF Client Management	1	1	1	1	1
Audit Service	1	2	2	4	4
CM Service	1	4	4	0.5	2
PDS	8	6	6	6	6
PRE	13	4	4	4	4
Query Service	1	2	2	1	1
SM Service	9	8	8	6	6
PCRF-Core	10	8	8	8	8
Usage Monitoring	16	8	8	4	4
Performance	2	1	1	0.5	1
UDR Connector	10	6	6	4	4

cnDBTier Microservices Resources**Table 3-184 CnDBTier Resource allocation**

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
ndbappmysqld	6	12	12	20	20

Table 3-184 (Cont.) CnDBTier Resource allocation

Service Name	Replicas	CPU Request per Pod (#)	CPU Limit per Pod (#)	Memory Request per Pod (Gi)	Memory Limit per Pod (Gi)
ndbmcmd	2	4	4	8	10
ndbmtl	6	12	12	75	75
ndbmysqld	2	4	4	16	16
db-infra-monitor-svc	1	4	4	4	4
db-backup-manager-svc	1	0.1	0.1	0.128	0.128

3.6.1.2 CPU Utilization

This section lists the CPU utilization for Policy and cnDBTier microservices. The CPU utilization is the ratio between the (total CPU utilization against total CPU request (X)) versus (target CPU Utilization (Y) configured for the pod).

Policy Microservices Resource Utilization

The following table describes the bench mark number as per the system maximum capacity utilization for Policy microservices.

The average CPU utilization is the ratio between the current usage of resource to the requested resources of the pod i.e., total sum of CPU utilized for service pods / total CPU requested for service pods.

Table 3-185 CPU/Memory Utilization by Policy Microservices

App/ Container	CPU	Memory
AppInfo	3.00%	25.00%
Diameter Connector	1.00%	12.00%
Diameter Gateway	18.60%	18.00%
Config Service	5.00%	19.00%
Egress Gateway	7.00%	18.00%
Ingress Gateway	0.00%	10.00%
NRF Client NF Discovery	0.00%	33.59%
NRF Client NF Management	0.00%	45.00%
UDR Connector	5.00%	24.00%
Audit Service	0.00%	28.70%
CM Service	3.50%	38.00%
PDS	6.00%	28.00%
PRE Service	8.00%	48.00%
Query Service	0.00%	23.00%
SM Service	0.00%	14.00%
Usage Monitoring	5.00%	67.00%

Observed CPU utilization Values of cnDBTier Services

The following table provides information about observed values of cnDBTier services.

Table 3-186 CPU/Memory Utilization by CnDBTier services

Service	CPU	Memory
ndbappmysqld/mysqlInDbcluster	51.50%	44.70%
ndbmcmd/db-infra-monitor-svc	10.30%	16.90%
ndbmttd/mysqlInDbcluster	35.1%	72.60%
ndbmttd/db-backup-executor-svc	35.1%	2.32%
ndbmttd/db-infra-monitor-svc	35.1%	13.60%

3.6.1.3 Results

Table 3-187 Average latency observations

Scenario	Average Latency (ms)	Peak Latency (ms)
Gx-init	130	260
Gx-Update_1st	103	207
Gx-Update_2nd	104	209
Gx-Update_3rd	104	208
Gx-Terminate	86	172
Overall	105	211

Table 3-188 Average NF service latency

NF Service Latency(In Seconds)	Avg (ms)
Ingress Gateway	31.8
PDS	83.8
UDR	22.4
Binding	51.8
Egress Gateway	20.4
Usage-Mon	94.4
PCRF-Core	3.84
Diameter Gateway	124
PRE	123