Oracle® Communications

Cloud Native Core Security Guide

Release
F29934-01
April 2020

Oracle Communications Cloud Native Core Security Guide, Release

F29934-01

Contents

List of Tables

1 Introduction

This document contains recommendations (short statements on how to operate and manage the CNC software) and procedures (step by step instructions to assist the customer in tailoring or hardening the CNC system).

Where possible, the CNC system software install as "secure by default". In the few cases where this isn't possible, an install-time checklist procedureis created and listed on the Cloud Native Core Security Checklist, a short list of post-installation hardening activities that must be performed by the customer before placing the system into operation. The recommendations and other procedures found in this document are optional, and should be considered in the context of your organization's approved security policies.

This security guide also provides a simplified trust model for the system. Additional configuration changes not included in this document are not generally recommended as they could hinder the product's operation and/or Oracle's capability to provide appropriate support.

Audience

This guide is intended for installers, administrators and operators responsible for product and network security.

References

The following references provide additional background on product operations and support.

- OCCNE Installation Guide

Acronyms

Table 1-1 Acronyms

Term	Definition
OSSA	Oracle Software Security Assurance
OC-CNE	Oracle Communications CNE
NF	Network Function. A service providing some function in the 5G Core Network.
FAQ	Frequently Asked Questions
CNE	Cloud Native Environment
5GC	5G Core Network

Locate Product Documentation on the Oracle Help Center Site

Oracle Communications customer documentation is available on the web at the Oracle Help Center (OHC) site, http://docs.oracle.com. You do not have to register to access these documents. Viewing these files requires Adobe Acrobat Reader, which can be downloaded at http://www.adobe.com.

1. Access the Oracle Help Center site at http://docs.oracle.com.
2. Click the **Industries** link. The **Industries Documentation** page displays.
3. Click the `Oracle Communications` link.

   The **Communications** Documentation page appears. Most products covered by these documentation sets will appear under the headings **Signaling and Policy**.
4. Click on your Product and then the Release Number.

   A list of the entire documentation set for the selected product and release appears.

My Oracle Support

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

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

1. Select **2** for New Service Request.
2. Select **3** for Hardware, Networking and Solaris Operating System Support.
3. Select one of the following options:
   - 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**.

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.

2
Overview

Deployment Environment

The 4G / 5G Cloud Native Core offering provides a variety of possible configuration and deployment models:

Table 2-1 Deployment Environment

Type	Host	CNE	Description
Bare-Metal	HP Gen 10 Blades / Rack Mount Servers / Cisco Switches	OC-CNE [1]	In this environment, a kubernetes Cloud Native Environment is hosted directly on the bare metal hardware, while some other elements (DB or Bastion) are hosted using virtualized servers.
Cloud	Customer Cloud	OC-CNE	In this environment, all the system elements are hosted in virtualized servers deployed on a customer provided Openstack environment. The OC-CNE is deployed on the openstack infrastructure.
Cloud	Customer CNE	Customer CNE [2]	In this environment, the customer provides the CNE and deploys the 5G NFs directly into the environment. The Oracle provided common services, and DB Tier are not used; equivalent functionality is provided by the customer.

1. Oracle Communications CNE provides basic CNE environment for on-premise deployment
2. Customer CNE provides CNE environment for running 5G microservices

The cloud environment security recommendations and procedures focuses on the OC-CNE reference environment. Customers providing their own CNE must have equivalent security procedures already in place.

3
Cloud Native Core Network Functions

This document details the administrative steps required to secure 4G/5G network operations.

Table 3-1 5G Network Functions

Network	Abbreviation	Description
Network (function) Repository Function	NRF	NRF provides registration, discovery and authorization services to all the Network Functions (NF) in the 5G core network.
Service Communication Proxy Function	SCP	SCP provides a 5G-aware service mesh. The SCP is not a part of the current 3GPP 5G specification, but is expected to be added to a future iteration.
InterWorking Function	IWF	IWF provides 4G/ 5G inter-working support.
Network Slice Selection Function	NSSF	NSSF works with the Access and Mobility Function (AMF) to select the network slice to be used by the User Equipment (UE).
Security Edge Protection Proxy	SEPP	In the roaming architecture, the home and the visited network are connected through Security Protection Proxy (SEPP) for the control plane of the internetwork interconnect.
Unified Data Repository	UDR/UDSF	UDR is a repository of subscriber information, and is used by various NFs (including UDR, PCF, and NEF). The UDSF is a part of the Unified Data Management Function (UDF) and is used to store state information for Network Functions (NF).
Unified Data Management/ Authentication Server Function	UDM/AUSF	UDM uses the subscription data stored in UDR and implements the application logic to perform various functionalities such as authentication credential generation, user identification, service and session continuity etc. The Authentication Server Function (AUSF) uses data stored in the UDM to perform authentication.
Network Exposure Function	NEF	Securely exposes network capabilities and events to Application Functions (AF).
Policy Control Function	PCF	Implements a unified policy framework for implementing control plane rules.
Binding Support Function	BSF	Provides PCF binding (mapping and selection) for User Equipment (UE).

Table 3-2 4G Network Functions

Network	Abbreviation	Description
Cloud Native 4G Diameter Routing Agent Network Function	cnDRA	Provides core (subset) 4G DSR capabilities delivered in a CNE microservice.
Cloud Native 4G Policy Control Repository Function	cnPCRF	Provides core 4G PCRF capabilities delivered in a CNE microservice.

4
Secure Development Practices

Overview of Secure Development Practices

Oracle Software Security Assurance (OSSA) is Oracle's methodology for building security into the design, build, testing, and maintenance of its products in every phase of the product development life cycle. These products are used on-premises by customers, or delivered through Oracle Cloud. Oracle's goal is to ensure that the products help customers meet their security requirements and provide the most cost-effective ownership experience.

Secure Development - DevSecOps

Oracle secures the DevOps development process using a variety of techniques:

- Broad developer training to developers for understanding the principles of secure software development
- Early creation of Trust Models and Risk Assessments to avoid common security pitfalls in our designs. Identify and expose sensitive interfaces to targeted testing for reducing or eliminating software vulnerabilities
- Extensive use of automated security testing to identify vulnerabilities in third party software
- Check for common OWASP (Open Source Foundation for Application Security) top 10 items and perform fuzz testing on key exposed interfaces
- Evaluate deployed software configurations using industry best practices

Vulnerability Handling

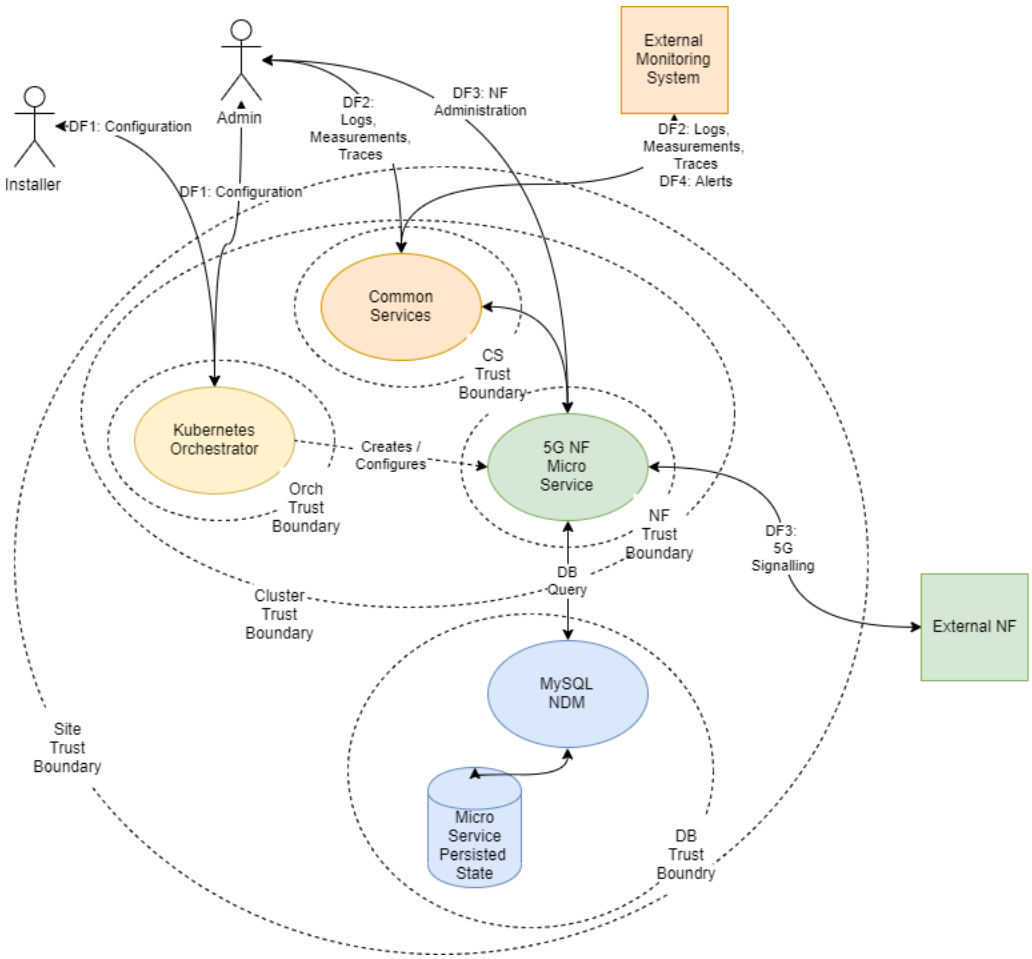
Review the Oracle Critical Patch Update Program

In general, the CNC Software is on a quarterly release cycle with each release providing feature updates and fixes, and updates to relevant third party software. These quarterly release provide cumulative patch updates.

5
Trust Model

The following Trust Model depicts the reference trust model (regardless of the target environment). The model describes the key access points and controls site deployment. While the model shows a single 5G NF microservice being deployed, typically many more would be deployed in an individual cluster.

Context diagram



Key Trust Boundaries

Table 5-1 Key Trust Boundaries

Trust Boundary	Contains	Access Control
Site Trust Boundary	All the NF and other supporting elements for a given site.	Cluster Access Policies are implemented using some kind of Access Control Group (or Security Group) mechanism.
Cluster Trust Boundary	All the Compute Elements for a given cluster	Network Policies controls traffic ingress and egress; Pod Security Policies controls the kinds of workloads allowed in the cluster (e.g., no pods requiring privilege escalation).
DB Trust Boundary	All the DB Tier Elements for a given Cluster	Firewall Policies control traffic ingress and egress; DB grants and other permission mechanisms provide authorization for authorized users.
Orchestrator Trust Boundary	The orchestration interface and keys	Firewall Policies control access to a Bastion server which provides orchestration services; access to the Bastion host uses SSH. The cluster orchestration keys are stored on the Bastion host.
CS Trust Boundary	The common services implementing logging, tracing, and measurements.	Each of the common services provides independent user interfaces (GUIs) that are currently open. The customer may want to introduce an api-gateway and implement authentication and authorization mechanisms to protect the OAM data. The common services may be configured to use Trasport Layer Security (TLS); when TLS is used, certificates will need to be generated and deployed via the orchestrator.
NF Trust Boundaries	A collection of one (or more) 5G Network Functions deployed as a service.	Some 5G NF microservices provide OAM access via a GUI. 5G NF microservices provide Signaling access via a TLS protected HTTP2 interface. The certificates for these interfaces are managed via the certificate manager.

External Data Flows

Table 5-2 External Data Flows

Data Flow	Protocol	Description
DF1: Configuration	SSH	The installer or administrator accesses the orchestration system, which is hosted on the Bastion Server. The install or administrator must use ssh keys to access the bastion to a special orchestration account (not root); no password access is allowed.
DF2: Logs, Measurements, Traces	HTTP/HTTPS	The administrator or operator interacts with the common services using web interfaces.

Table 5-2 (Cont.) External Data Flows

Data Flow	Protocol	Description
DF3: 5G Signaling	HTTP2 (w/TLS)	All signalling interaction between NFs at a site and NFs at an external site is sent via TLS protected HTTP/2.
DF4: Alerts	SNMP (Trap)	All alerting is performed using SNMP traps.

A full list of the network flows including service types and ports can be found in the Port Flow Appendix.

6

Common Security Recommendations and Procedures

4G/5G Application Authentication and Authorization

4G/5G NFs use mTLS to secure communication. All NFs require a trust relationship to be established with all peers (by exchanging and trusting peer root or intermediate certificates). All the peer certificates must be available in the trust store (K8s Secrets) in order to establish secured communication. Ideally, the trust store is populated from the customer PKI using ACME protocols; we also support manual importation and a semi-automatic import using the cert-manager external provider.

DB-Tier Authentication and Authorization

The DB-Tier provides a highly available multisite database used to store NF state and configuration. When installed, the MySQL DB is configured with a root account whose password is randomly generated. Each NF must have additional accounts for that particular NF. The procedures in this section explains how to change these account passwords. Additionally, communication between the NFs and the MySQL query nodes are protected using TLS.

Procedure: Modify MySQL NDB Root Password

This procedure is typically executed by the DB Administrator

For each of the MySQL Query nodes, perform the following steps :

Table 6-1 Modify MySQL NDB Root Password

Step	Description	Est Time
1.	Log into the next query node using `ssh: $ ssh admusr@<mysql query node>`	1m
2.	Become `root: $ sudo su`	1m
3.	Invoke mysql using existing DB Root credentials: `# mysql -h 127.0.0.1 -uroot -p` Enter password: `<enter existing root password>`	1m
4.	Change the DB Root credentials: mysql> ALTER USER'root'@'localhost'IDENTIFIED BY'<NEW_PASSWORD>'; mysql> FLUSH PRIVILEGES;	1m
5	Repeat steps 1 through 4 for each MySQL Query node.	

Note:

If you are accessing a DB instance for the first time, the DB Root password is stored in the `/var/occnedb/mysqld_expired.log` file. (The system generates a random password at installation time)

Note:

Recommendation: Separation of Roles

The roles of DB Administrator and Cluster Administration must be kept separate. The DB Administrator must be responsible for securing and maintaining the DB-Tier MySQL NDM cluster. The Cluster Administrator must be responsible for securing and operating the Bastion Host and K8s Cluster. When 5G NFs are installed, the DB Administrator will be required to create new NF database and NF DB accounts (using the DB Root credentials). Once this is completed, the Cluster Administrator installs the NF (using helm).

Note:

Recommendation: Use Strong Passwords

The DB Administrator must choose a complex DB Root password as per their organization's security guidelines.

7

Cloud Native Environment Security Recommendations and Procedures

After installation, the OC-CNE system security stance should be audited prior to placing the system into service. This primarily consists of changing credentials and sequestering SSH keys to trusted servers. The following table lists all the credentials that need to be checked, changed and retained:

Credential Name	Deployment	Type	Associated Resource	Initial Setting	Credential Rotation
TOR Switch	Bare Metal Only	username and password	Cisco Top or Rack Switch	username and password from PreFlight Checklist	Reset post-install
Enclosure Switch	Bare Metal Only	username and password	HP Enclosure Switch	username and password from PreFlight Checklist	Reset post-install
OA Admin	Bare Metal Only	username and password	On-board Administrator Console	username and password from PreFlight Checklist	Reset post-install
ILO Admin	Bare Metal Only	username and password	HP Integrated Lights Out Manger	username and password from PreFlight Checklist	Reset post-install
Server Super User (root)	All	username and password	Server Super User	Set to well-known Oracle default during server installation	Reset post-install
Server Admin User SSH	All	SSH Key Pair	Server Admin User	Key Pair generated at install time	Can rotate keys at any time; key distribution manual procedure

If factory or Oracle defaults were used for any of these credentials, it must be changed prior to placing the system into operation. The customer must store these credentials in a safe and secure way offsite. It is recommended that the customer must plan a regular schedule for updating (rotating) these credentials. Specific procedures and recommendations for OC-CNE credential management are provided below.

1.1.Network Security Recommendations and Procedures

Recommendation: Review and Follow TOR installation procedures

The OC-CNE on-premise installation guide provides detailed procedures on how to configure the TOR switches and configure them for remote monitoring. Deviations from the standard installation time configurations are not recommended.

Credential Management Procedures

Procedure 1: Setting Top Of Rack Switch Credentials

This procedure is used to set the credentials on the Cisco TOR switch as deployed with the bare metal deployment option. Steps for creating and deleting accounts and for setting account passwords is given below.

Table 7-1 Setting Top Of Rack Switch Credentials

Step No.	Description	Est time

1.	Login to the TOR switch (from the bastion host):
	`$ ssh <username>@<switchIP address>` User Access Verification
	`Password: <password>`
	Cisco Nexus Operating System (NX-OS) Software
	TAC support: www.cisco.com/tac
	`<switchname>#`
		1m

2.	Change the password for <username>: # configure Enter configuration commands, one per line. End with CNTL/Z. `(config)# username <username> password<newpassword>` (config)#exit	1m
3.	Create a new user (if desired): # configureEnter configuration commands, one per line. End with CNTL/Z. `(config)# username <newusername> password <newpassword> role` [network-operator|network-admin|vdc-admin|vdc-operator] (config)#exit	1m
4.	Verify the account changes by exiting the ssh session (type exit) and repeat step 1. # exit Connection to <switchIP address> closed. `$ $ ssh <newusername>@<switchIP address>` User Access Verification `Password: <newpassword>` Cisco Nexus Operating System (NX-OS) SoftwareTAC support: www.cisco.com/tac ..... <switchname>#	1m
5.	Delete an unneeded user account: # configureEnter configuration commands, one per line. End with CNTL/Z. `(config)# no username <username>` (config)#exit	1m

Table 7-1 (Cont.) Setting Top Of Rack Switch Credentials

Step No.	Description	Est time
6.	Change the enable secret: `(config)# enable secret <newenablepassword>` (config)# exit	1m
7.	Save the configuration changes: `# copy running-config startup-config` 100% Copy complete, now saving to disk (please wait)... Copy complete.	1m

> **Note:**
>
> **Recommendation: Change TOR passwords before placing site into service**. The TOR switch credentials show be changed prior to placing the site into service.

> **Note:**
>
> **Recommendation: Use Strong Passwords**.The Network Administrator must choose complex TOR Switch passwords as per their organization's security guidelines.

Procedure 2: Setting Enclosure Switch Credentials
This procedure is used to set the credentials on the HP enclosure switch as deployed with the bare metal deployment option. Steps for creating and deleting accounts and for setting account passwords is given below. For additional information, refer to: HP commands to configure enclosure switch username and password

Table 7-2 Setting Enclosure Switch Credentials

Step	Description	Est. Time
1.	Login to the HP enclosure switch (from the bastion host): `$ ssh <username>@< switchIP address>` Copyright (c)2010-2017Hewlett Packard Enterprise Development LP ** Without the owner's prior written consent, ** no decompiling or reverse-engineering shall be allowed. <switchname> <switchname> sysSystem View:returnto User View with Ctrl+Z.	1m

Table 7-2 (Cont.) Setting Enclosure Switch Credentials

Step	Description	Est. Time

2.	Change the password for the current username: `[switchname]local-user <username>class <currentclass>`
	`[switchname-luser-manage-<username>]password simple <newpassword>`
	`[switchname-luser-manage-<username>]quit`
		1m

3.	Create a new user account: `[switchname]local-user <newusername>class[manage|network]` New local user added `[switchname-luser-manage-<newusername>]password simple <newpassword>[switchname-luser-manage-<newusername>]quit`	1m
4.	Verify the account changes by exiting the ssh session (type exit) and repeat step 1. `<switchname> quitConnection to <switchIP address>closed. $` `$ ssh <newusername>@< switchIP address> <newusername>@<switchIP address>'s password: <newpassword>` Copyright (c)2010-2017Hewlett Packard Enterprise Development LP * * Without the owner's prior written consent, * * no decompiling or reverse-engineering shall be allowed. <switchname> <switchname> sys System View:returnto User View with Ctrl+Z.	1m
5.	Delete an unneeded user account: `[switchname]undo local-user <username>class <currentclass>`	1m
6.	Save the configuration changes: [switchname]save The current configuration will be written to the device. Are you sure? [Y/N]: y Please input the file name(*.cfg)[flash:/<filename>] (To leave the existing filename unchanged, press the enter key): flash:/<filename> exists, overwrite? [Y/N]: yValidating file. Please wait... Saved the current configuration to mainboard device successfully. Slot1: Save next configuration file successfully. [switchname]	1

> **Note:**
>
> **Recommendation: Set Enclosure Switch Credentials before Placing Into Service**
>
> The HP Enclosure switch credentials show be changed prior to placing the site into service.
>
> **Recommendation: Use Strong Passwords**
>
> The Network Administrator must choose complex Enclosure Switch passwords as per their organization's security guidelines.

1.2 Hosting Environment Security Recommendations and Procedures

The Oracle Linux 7 security guide is available at: https://docs.oracle.com/en/operating-systems/oracle-linux/7/security/E54670.pdf

This guide provides additional details for specific security procedures outlined below. However several of the procedures found in the general OL7 guide are not appropriate for the OC-CNE environment; contact Oracle Support before attempting any security related procedures which are not recommended below.

Repository Management Recommendations
System Update (YUM) Recommendations

Keep central repositories up-to date with latest yum packages; yum updates are performed on-site whenever a fresh install or upgrade is performed. An up-to date yum repository will help ensure that fixes for all publish vulnerabilities are applied.

Docker Repository Recommendations

Scan docker image repositories regularly: Scan your docker image repositories regularly using a tool such as clair or anchored-engine. All images are scanned and vulnerabilities assessed at product development time, but new exploits / vulnerabilities may be reported/fixed later. Scan tools typically use a database of known vulnerabilities - refer to tool vendor for instructions on creating off-line (internet isolated) vulnerability database

1.3 Credential Management Procedures

Procedure 1: Setting HP Onboard Administrator (OA) Credentials.

This procedure is used to set the credentials on the HP Onboard Administrator as deployed with the bare metal deployment option. Steps for creating and deleting accounts and for setting account passwords is shown. For additional information, please refer to: HP commands to configure OA username and password.

Table 7-3 Setting HP Onboard Administrator (OA) Credentials

Step	Description	Est Time

1	Login to the OA:
	`$ ssh <username>@<OA address>`
	---------- 	
	--- 	
	WARNING: This is aprivatesystem. Do not attempt to login unless you are anauthorized user. Any authorized or unauthorized access and use may be moni-tored and can result in criminal or civil prosecution under applicable law
	.----------
	--- 	
	Firmware Version:4.85
	Built:04/06/2018@06:14OA
	Bay Number:1
	OA Role: Active
	`<username>@<OA address>'s password: <password>`
	HPE BladeSystem Onboard Administrator
	(C) Copyright2006-2018Hewlett Packard Enterprise Development LP
	Type'HELP'to display a list of valid commands.
	Type'HELP <command>'to display detailed information about a specific command.
	Type'HELP HELP'to display more detailed information about the help system.
	OA-A45D36FD5FB1>
		1m

2	Change the current password: OA-A45D36FD5FB1> set password <newpassword> Changed passwordforthe"<username>"user account. OA-A45D36FD5FB1>	1m
3	Add new user: OA-A45D36FD5FB1> add user <newusername> New Password: <newpassword> Confirm : <newpassword> User"<newusername>"created. You may set user privileges with the 'SET USER ACCESS' and 'ASSIGN' commands. OA-A45D36FD5FB1> set user access <newusername> [ADMINISTRATOR|OPERATOR| USER]"<newusername>"has been given [administrator|operator|user] level privileges.	1m

Table 7-3 (Cont.) Setting HP Onboard Administrator (OA) Credentials

Step	Description	Est Time
4	Assign full access to the enclosure for the user: OA-A45D36FD5FB1> assign server all <newusername> <newusername> has been granted access to the valid requested bay(sOA-A45D36FD5FB1> assign interconnect all <newusername> <newusername> has been granted access to the valid requested bay(s)OA-A45D36FD5FB1> assign oa <newusername> <newusername> has been granted access to the OA.	1m
5	Verify the new account: OA-A45D36FD5FB1> exit Connection to <OA address> closed.[bastion host]# ssh <newusername>@<OA address> --- WARNING: This is a private system. Do not attempt to login unless you are unauthorized user. Any authorized or unauthorized access and use may be monitored and can result in criminal or civil prosecution under applicable law. --- Firmware Version:4.85 Built:04/06/2018@06:14 OA Bay Number:1 OA Role: Active `<newusername>@<OA address>'s password: <newpassword>` HPE BladeSystem Onboard Administrator (C) Copyright2006-2018Hewlett Packard Enterprise Development LP Type 'HELP' to display a list of valid commands. Type 'HELP <command>' to display detailed information about a specific command. Type 'HELP HELP' to display more detailed information about the help system. OA-A45D36FD5FB1>	1m
6	Delete an unneeded user account: OA-A45D36FD5FB1> remove user <username> Entering anything other than 'YES' will result in the command not executing. Are you sure you want to remove testuser1? yes User"<username>"removed.	1m

Procedure 2: Setting HP Integrated Lights Out Manger (ILO) Credentials

This procedure is used to set the credentials on the HP Integrated Lights Out Managers as deployed with the bare metal deployment option. Steps for creating and deleting accounts and for setting account passwords is shown.

Table 7-4 Setting HP Integrated Lights Out Manger (ILO) Credentials

Step	Description	Est Time
1	Login to the iLO: $ ssh <username>@<iLO address> <username>@<iLO address>'s password: <password>User:<username> logged-in to ...(<iLO address> / <ipv6 address>) iLO Advanced2.61at Jul272018 Server Name: <server name> Server Power: On </>hpiLO->	1m
2	Change the current password: `</>hpiLO-> set /map1/accounts1/ <username> password= <newpassword>` status=0 status_tag=COMMAND COMPLETED Tue Aug2013:27:082019 </>hpiLO->	1m
3	Create a new user account: `</>hpiLO-> create /map1/accounts1 username= <newusername> password= <newpassword>` group=admin,config,oemHP_rc,oemHP_power,oemHP_vm status=0 status_tag=COMMAND COMPLETED Tue Aug2013:47:562019 User added successfully.	1m
4	Verify the new user account: </>hpiLO-> exit status=0 status_tag=COMMAND COMPLETED Tue Aug2013:30:522019CLI session stoppedReceived disconnect from <iLO address> port22:11: Client Disconnect Disconnected from <iLO address> port22 [bastion host]# ssh <newusername>@<iLO address> <newusername>@<iLO address>'s password: <newpassword> User:<newusername> logged-in to ...(<iLO address> / <ipv6 address>) iLO Advanced2.61at Jul272018 Server Name: <server name>Server Power: On</>hpiLO->	1m
5	Delete an unneeded account: </>hpiLO-> delete /map1/accounts1/ <username> status=0 status_tag=COMMAND COMPLETED Tue Aug2013:59:042019 User deleted successfully.	

Procedure 3: Setting Root Passwords for All Cluster Nodes

The procedure to reset the root account required that the administrator login to each and every server.

To reset the root account ,for each and every server in the cluster perform the following steps:

Table 7-5 Setting Root Passwords for All Cluster Nodes

Step	Description	Est. time
1	Login to the next server: `$ ssh admusr @<cluster server IP>`	1m
2	Perform the root password change: `$ sudo passwd root` New password: <new password> Retype new password: <new password> Retype new password:<new password>	1m
3	Repeat steps 1 - 2 for each and every server in the cluster.	

Note:

The administrator (admusr) account is provided without a usable password hash. Thus requiring the use of SSH keys to access the account. The SUDO users access is configured without the requirement of a password. If you would like to enable the SUDO passwords for the administrator, you also need to assign a password to the administrator account using a procedure very similar to the one outlined above.

8

4G/5G Core Network Function Security Recommendations and Procedures

Network Repository Function(OCNRF) Security Recommendations and Procedures

This addendum provides Network Function Repository Function (OCNRF) specific security recommendations and procedures. Recommendations common to all 5G/4G are found in the Common Procedures Section

Access Token configuration

Table 8-1 Access Token configuration

Step	Description	Est time
1	Create following files:- ECDSA private key (For example:- `ecdsa_private_key_pkcs8.pem`) RSA private key (For example:- `rsa_private_key_pkcs1.pem`) TrustStore password file (For example:- `trustStorePassword.txt`) KeyStore password file (For example:- `keyStorePassword.txt`) CA signed ECDSA OCNRF certificate (For example:- `ecdsa_ocnrf_certificate.crt`) CA signed RSA OCNRF certificate (For example:- `rsa_ocnrf_certificate.crt`) **Note:** Creation of keys, certificates, password is on discretion of user/operator, how to create	5m
2	Login to Bastion Host or server from where kubectl can be executed	1m
3	Create namespace for the secret `$ kubectl create namespace ocnrf`	1m

Table 8-1 (Cont.) Access Token configuration

Step	Description	Est time
4	Create kubernetes secret for NF Access token: **Note:**The filenames in below command are same as in Step 1 `$ kubectl create secret generic ocnrfaccesstoken-secret --from-file= ecdsa_private_key_pkcs8.pem --from-file=rsa_private_key_pkcs1.pem --from-file= trustStorePassword.txt --from-file=keyStorePassword.txt --from-file= ecdsa_ocnrf_certificate.crt--from-file=rsa_ocnrf_certificate.crt -n ocnrf`	1m
5	Verify that secret is create successfully `$ kubectl describe secret ocnrfaccesstoken-secret -n ocnrf`	1m

How to update keys used to sign JSON Web Token (JWTs) for Access Token

Table 8-2 How to update keys used to sign JSON Web Token (JWTs) for Access Token

Step	Description	Est time
1	Update the following files as per need to update the keys: ECDSA private key (For example:- `ecdsa_private_key_pkcs8.pem`) RSA private key (For example:- `rsa_private_key_pkcs1.pem`) CA signed ECDSA OCNRF certificate (For example:- `ecdsa_ocnrf_certificate.crt`) CA signed RSA OCNRF certificate (For example:- `rsa_ocnrf_certificate.crt`) NOTE:- Updation of keys, certificates, password is on discretion of user/operator, how to create.	5m
2	Login to Bastion Host or server from where kubectl can be executed	1m

3	Update the secret with new/updated details
	`# Delete the secret and recreate it`
	`$ kubectl delete secret ocnrfaccesstoken-secret -n ocnrf`
	`# Recreate the secret with updated details`
	`$ kubectl create secret generic ocnrfaccesstoken-secret --from-file=ecdsa_private_key_pkcs8.pem --from-file=rsa_private_key_pkcs1.pem --from-file=trustStorePassword.txt --from-file=keyStorePassword.txt --from-file=ecdsa_ocnrf_certificate.crt--from-file=rsa_ocnrf_certificate.crt -n ocnrf`
		1m

OCNRF MYSQL Secret Configuration

Table 8-3 OCNRF MYSQL Secret configuration

Step	Description	Est time
1	Login to Bastion Host or server from where kubectl can be executed	1m
2	Create namespace for the mysql secret.Skip this step, if already created. `$ kubectl create namespace ocnrf`	1m
3	Create kubernetes secret for Mysql : `$ kubectl create secret generic database-secret --from-literal=dbUsername=<OCNRF Mysql database username> --from-literal=dbPassword=<OCNRF Mysql database passsword>--from-literal=dbName=<OCNRF Mysql database name> -n $<Namespace of MYSQL secret>`	1m
4	Verify that secret is create successfully: `$ kubectl describe secret database-secret -n ocnrf`	1m

OCNRF MYSQL Secret Updates for Password of DB User

Table 8-4 OCNRF MYSQL Secret updates for password of DB user

Step	Description	Est time
1	Login to Bastion Host or server from where kubectl can be executed	1m
2	Update the kubernetes secret for Mysql # Delete the secret kubectl create secret database-secret # Create the secret with updated details `$ kubectl create secret generic database-secret --from-literal=dbUsername=<OCNRF Mysql database username> --from-literal=dbPassword=<OCNRF Mysql database passsword>--from-literal=dbName=<OCNRF Mysql database name> -n $<Namespace of MYSQL secret>`	1m

Policy Control Function (PCF)Security Recommendations and Procedures

This addendum provides Policy Control Function (PCF) specific security recommendations and procedures. Recommendations common to all 5G/4G are found in the Common Procedures Section

Access Token configuration

Table 8-5 Access Token configuration

Step	Description	Est time
1	Create following files: ECDSA private key (For example: `ecdsa_private_key_pkcs8.pem`) RSA private key (For example: `rsa_private_key_pkcs1.pem`) TrustStore password file (For example: `trustStorePassword.txt`) KeyStore password file (For example: `keyStorePassword.txt`) CA signed ECDSA OCPCF certificate (For example: `ecdsa_ocpcf_certificate.crt`) CA signed RSA OCPCF certificate (For example: `rsa_ocpcf_certificate.crt`) **Note:** Creation of keys, certificates, password is on discretion of user/operator.	5m
2	Login to Bastion Host or server from where kubectl can be executed	1m
3	Create namespace for the secret: `$ kubectl create namespace ocpcf`	1m
4	Create kubernetes secret for NF Access token : **Note:** The filenames in below command are same as in Step 1 `$ kubectl create secret generic ocpcfaccesstoken-secret --from-file= ecdsa_private_key_pkcs8.pem --from-file=rsa_private_key_pkcs1.pem --from-file= trustStorePassword.txt --from-file=keyStorePassword.txt --from-file= ecdsa_ocpcf_certificate.crt--from-file=rsa_ocpcf_certificate.crt -n ocpcf`	1m
5	Verify that secret is create successfully: `$ kubectl describe secret ocpcfaccesstoken-secret -n ocpcf`	1m

How to update keys used to sign JSON Web Token (JWTs) for Access Token

Table 8-6 How to update keys used to sign JSON Web Token (JWTs) for Access Token

Step	Description	Est time
1	Update the following files as per need to update the keys: ECDSA private key (For example: `ecdsa_private_key_pkcs8.pem`) RSA private key (For example: `rsa_private_key_pkcs1.pem`) CA signed ECDSA OCPCF certificate (For example:- `ecdsa_ocpcf_certificate.crt`) CA signed RSA OCPCF certificate (For example:- `rsa_ocpcf_certificate.crt`) **Note:** How to create and update keys, certificates, password is on discretion of user or operator.	5m
2	Login to Bastion Host or server from where kubectl can be executed	1m

3	Update the secret with new/updated details
	`# Delete the secret and recreate it`
	`$ kubectl delete secret ocpcfaccesstoken-secret -n ocpcf`
	`# Recreate the secret with updated details`
	`$ kubectl create secret generic ocpcfaccesstoken-secret --from-file=ecdsa_private_key_pkcs8.pem --from-file=rsa_private_key_pkcs1.pem --from-file=trustStorePassword.txt --from-file=keyStorePassword.txt --from-file=ecdsa_ocpcf_certificate.crt--from-file=rsa_ocpcf_certificate.crt -n ocpcf`
		1m

OCPCF MYSQL kubernetes secret for storing database username and password

Table 8-7 OCPCF MYSQL kubernetes secret

Step	Description	Est time
1	Login to Bastion Host or server from where kubectl can be executed	1m
2	Create namespace for the mysql secret. Skip this step, if already created. `$ kubectl create namespace <namespace>`	

Table 8-7 (Cont.) OCPCF MYSQL kubernetes secret

Step	Description	Est time

3	Create a yaml file with the username and password in with the syntax shown below:
	`apiVersion: v1`
	`kind: Secret`
	`metadata:`
	`name: <secret-name>`
	`type: Opaque`
	`data:`
	`mysql-username: cGNmdXNy`
	`mysql-password: cGNmcGFzc3dk`
	Note: The values for "mysql-username" and "mysql-password" should be base64 encoded.
		1m

4	Execute "kubectl create -f <yaml_file_name> -n <namespace> to create the secret.	1m
5	Verify: `$ kubectl describe secret <secret-name> -n <namespace>`	1m

Native Core (cnDRA) Security Recommendations and Procedures

User (OAM) Authentication and Authorization

- cnDRA supports REST based MMI interface. There is no GUI provided in the current cnDRA release.
- The MMI interface is based on fixed user and password, using which the security token is requested by REST client from cnDRA.
- cnDRA does not allow or support configuration or modify these credentials (user and password).

Authentication and Authorization of Applications

cnDRA currently supports TCP based signaling traffic connection towards the Remote Peer Nodes. These connections are not currently secured via TLS etc mechanism. Currently there is no plan to enable securing of the application/Diameter traffic.

Cloud Native Core - Ingress/Egress Gateways - Security Recommendations / Procedures

Enabling TLS and Ciphers in Ingress/Egress Gateway

Step	Description
1	**Helm Configuration to enable TLS:** To open Https port in **Ingress gateway:** configure in helm **enableIncomingHttps: true** To have a Https client configured in **Egress gateway:** configure in helm **enableOutgoingHttps: true**
2	Create following files: 1. RSA or ECDSA Private key (For example: rsa_private_key_pkcs1.pem) 2. Trust store password (For example: trust.txt) 3. Key store password(For example: key.txt) 4. Certificate chain for trust store (For example: caroot.cer) 5. Signed server certificate (For example: ocingress.cer) or Signed client certificate (For example: ocegress.cer) **Note:** How to do the Creation of keys, certificates, password is on discretion of user or operator.
3	Create secret Command : `$ kubectl create secret generic ocingress-secret --from-file=rsa_private_key_pkcs1.pem --from-file=trust.txt --from-file=key.txt --from-file=ocingress.cer --from-file=caroot.cer -n ocingress`
4	Enable cipher suites: # Cipher Suites to be enabled on Server side (Ingress Gateway), # Cipher Suites to be enabled on Client side (Egress Gateway), **cipherSuites**: -TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 **Note:** The above list is the allowed cipher list as per Verizon requirement, this also coincides with the allowed list of ciphers as per Oracle standards. Helm deployment may fail due to invalid configuration or cipher suite mismatch. Manual restart of pod is required if there is update in cipher configuration during run time.

Certificate Management and Dynamic reload of certificates in Gateways

Whenever certificates gets compromised or a new certificate chain is required to be added to the truststore, we can update the key and truststore used by the application.

To update the key and the truststore, update or replace the secret:

Command:

```
$ kubectl create secret generic ocingress-secret --from-
file=rsa_private_key_pkcs1.pem --from-file=trust.txt --from-file=key.txt
--from-file=tmp.cer --from-file=caroot.cer --dry-run -o yaml -n
ocingress| kubectl replace -f - -n ocingress
```

Whenever there is an update in the certificate chain or signed certificate placed in secret, kubernetes watcher which is implemented in update container will check for change in file state and replace the key and truststore accordingly in the mounted shared volume.

Dynamic reload of certificates is not supported in **Ingress Gateway** as of now, so a manual restart of pod is required when any update in the configuration is made with respect to https.

In case of **Egress Gateway** update container will trigger the rest end point to dynamically reload key and truststore. Then egress gateway will pickup new store files from shared volume and reload trust and key managers. Egress gateway will use the replaced store to establish new connections and gracefully terminate existing connections by sending a GOAWAY frame.

A

Cloud Native Core Network Port Flows

Network Port Flows

- Cluster IP addresses are reachable outside of the cluster and are typically assigned via a Network Load Balancer
- Node IP addresses are reachable from the bastion host (and may be exposed outside of the cluster)

OC-CNE Port Flows

Table A-1 OC-CNE Port Flows

Name	Sever/ Container	Ingress Port ext[:int]/ Proto	TLS	Cluster IP (Service IP)	Node IP	Notes
SSH Access	ALL	22/TCP	Y		SSH Access	Administrative SSH Access; no root/key only.
RPC Bind	All	111/TCP, UDP	N		RPCBind	Used for installation; pxe booting of NFS mounted images
Repository	Bastion Host	80/TCP, 443/TCP, 5000/TCP	Y		Repository Access	Access repositories (YUM, Docker, Helm, etc.)
Prometheus Server	K8s Nodes	80:9090/TCP	N	GUI		Prometheus Server
Prometheus Push Gateway	K8s Nodes	9091/TCP	N		Push Gateway	Prometheus Push Gateway
Prometheus Exporters	K8s Nodes	9100-9551/TCP 24231/TCP (fluent) 9099/TCP (snmp)	N		Prometheus Exporters	Prometheus Exporters
MySQL Query	MySQL SQL Node	3306/TCP	N	Replication Traffic	Microservice SQL Access	The SQL Query interfaces are used for 5G NFs to access the database and for remote sites to replicate data
MySQL Management	MySQL Management Node	1186/TCP	N	Management Console Access		The SQL Management interface is used to access the management interfaces for the data cluster

Table A-1 (Cont.) OC-CNE Port Flows

Name	Sever/ Container	Ingress Port ext[:int]/ Proto	TLS	Cluster IP (Service IP)	Node IP	Notes
MySQL Data	MySQL Data Node	50501/TCP	N		SQL Query Backend	The SQL Data interface provide a backend DBMS interface for the SQL Query Nodes
Kubelet cAdvisor	K8s Nodes	4149/TCP	Y		Container Metrics	Default cAdvisor port used to query container metrics
Kubelet API	K8s Nodes	10250/TCP	Y		Control Plane Node Access	API which allows full node access
Kube-scheduler	K8s Nodes	10251/TCP	N		Scheduler Access	Serve HTTP insecurely
Kube-Scheduler	K8s Node	10259/TCP	Y		Scheduler Access	HTTPS Access
Kube-proxy	K8s Nodes	10256/TCP	N		Health Check	Health check server for Kube Proxy
Kube-controller	K8s Nodes	10252/TCP	N		Controller Access	Serve HTTP insecurely
Kube-controller	K8s Nodes	10257/TCP	Y		Controller Access	HTTPS Access
Kube API Server	K8s Master Nodes	6443/TCP	Y		K8s Orchestration	The Kube API Server provides an orchestration API for the creation of K8s resources.
Kibana	K8s Nodes	80:5601/TPC	N	GUI		Logging Visualization
Jaeger Query	K8s Nodes	80:16686/TCP	N	GUI		Service Frontend
Jaeger Collector	K8s Nodes	14268/TCP	N		Collector	Accept jaeger.thrift directly from clients
Jaeger Collector	K8s Nodes	9411/TCP	N		Collector	Zipkin compatable endpoint (optional)
Jaeger Agent	K8s Nodes	6831/UDP	N		Agent	Accept jaeger.thrift over compact thrift protocol
Jaeger Agent	K8s Nodes	6832/UDP	N		Agent	Accept jaeger.thrift over binary thrift protocol
Jaeger Agent	K8s Nodes	5778/TCP	N		Agent	Serve Configs
ILO	ILO Management Port	443/TCP	Y		Installation / Management	This interface is used to manage the frame; it provided low level management for all of the frame HW assets

Table A-1 (Cont.) OC-CNE Port Flows

Name	Sever/ Container	Ingress Port ext[:int]/ Proto	TLS	Cluster IP (Service IP)	Node IP	Notes
Grafana	K8s Nodes	80:3000/TCP	N	GUI		Grafana
ETCD Peer	K8s Master Nodes	2380/TCP	Y		Peer Access	ETCD Server Communication
ETCD Client	K8s Master Nodes	2379/TCP	Y		Client Access	Keystore DB used by K8s
ElasticSearch	K8s Nodes	9200/TCP	N	GUI		Search API access
ElasticSearch	K8s Nodes	9300/TCP	N		Logging	Internal Logging
BGP	K8s Nodes	179/TCP	N		BGP	Used on bare metal environments in load balancing
Alertmanager clustering	K8s Nodes	9094/TCP	N		Amertmanger Clustering	Alertmanager Clustering
Alertmanager	K8s Nodes	80:9093/TCP	N	GUI		Alertmanager

NF Port Flows

Table A-2 NF Port Flows

Name	Sever / Container	Ingress Port [external]:internal	TLS ?	Cluster IP (Service IP)	Node IP	Notes
5G NRF	K8s Nodes/NRF Service	80/TCP 443/TCP	Y	NfConfiguration IngressGateway	NfRegistration NfSubscription NfDiscovery NfAccessToken EgressGateway	5G NRF
5G SPF	K8s Nodes/SPF Worker	8000/TCP	N		5G Proxy	5G SCP (SPF) Proxy
5G SPF	K8s Nodes/ Soothsayer	8082/TCP	N	Proxy Configuration		5G SCP (SPF) Proxy Configuration

Table A-2 (Cont.) NF Port Flows

Name	Sever / Container	Ingress Port [external]:internal	TLS ?	Cluster IP (Service IP)	Node IP	Notes
5G SPF	K8s Nodes/ Istio	???/TCP	N		Mesh State Sharing	5G SCP (SPF) Mesh Management
5G NSSF	K8s Nodes/ NSSF Service	80/TCP	N	NSSF configuration	NSSF selection, NSSF policy, NSSF registration	5G NSSF
5G UDR/ UDSF	K8s Nodes/UDR Service	80/TCP	N		Nudr-dr/ Nudr-prov	5G UDR: Signaling network can be used for 1 management API exposed