NOKIA

Centralized License Manager

Release 20.3

Installation and Upgrade Guide

3HE-16174-AAAA-TQZZA

Issue 1

March 2020

Legal notice

Nokia is a registered trademark of Nokia Corporation. Other products and company names mentioned herein may be trademarks or tradenames of their respective owners.

The information presented is subject to change without notice. No responsibility is assumed for inaccuracies contained herein.

© 2020 Nokia.

Contents

About this document

Purpose

The *CLM Installation and Upgrade Guide* provides detailed information regarding the installation of the CLM, including pre- and post-installation activities.

Safety information

For your safety, this document contains safety statements. Safety statements are given at points where risks of damage to personnel, equipment, and operation may exist. Failure to follow the directions in a safety statement may result in serious consequences.

Document support

Customer documentation and product support URLs:

- Documentation Center
- Technical support

How to comment

Documentation feedback

1 Pre-installation

1.1 Introduction

1.1.1 Overview

CAUTION

Service Disruption

A CLM instance is node-locked to the server where it is installed.

Network Pool Keys issued by Nokia are tied to a specific CLM instance by its UUID.

Resource and network parameters associated with the CLM server shall not be altered after Network Pool Keys are received for a specific CLM instance.

An operator may migrate a CLM instance within a VM environment only if the server resource and network characteristics remain constant.

This chapter provides information and procedures that may need to be understood/performed prior to installing or upgrading the CLM.

1.2 Operating system specifications

1.2.1 Red Hat Enterprise Linux (RHEL) description and recommendations

The CLM is supported on the following base RHEL versions:

- RHEL server 7 x86-64 - Update 3 (7.3)
- RHEL server 7 x86-64 - Update 4 (7.4)
- RHEL server 7 x86-64 - Update 5 (7.5)
- RHEL server 7 x86-64 - Update 6 (7.6)
- RHEL server 7 x86-64 - Update 7 (7.7)

Previous releases, or other variants of Red Hat, and other Linux variants are not supported.

The CLM does not necessarily support all functionality provided in RHEL. SELinux, iptables, and Network Manager are not supported in CLM configurations. The CLM should use a time synchronization mechanism, such as NTP, to ensure accurate time. The CLM also requires that the server hostname is configured in the */etc/hosts* file. RHEL must be installed in 64 bit mode where the CLM will be installed.

Customers are expected to purchase RHEL software and support for all platforms running RHEL Server with the CLM. It is strongly recommended to purchase a support package from Red Hat that provides 24x7 support.

Nokia recommends the installation of any OS, driver, or firmware updates that the hardware vendor advises for RHEL.

With the exception of documented Operating System parameter changes for CLM, all other settings must be left at the RHEL default configuration.

1.2.2 Third-party applications

Applications that are not sanctioned by Nokia must not be running on any virtual instance running the CLM. Nokia reserves the right to remove any applications that are suspected of causing issues from workstations running CLM.

1.3 RHEL OS installation requirements

1.3.1 Introduction

This section describes the RHEL OS installation requirements for a CLM system.

Each CLM server requires the following:

- a specific RHEL Software Selection as the base environment
- the installation and removal of specific OS packages

Note: The RHEL rpm utility requires hardware driver files in binary format. If the RHEL driver files provided by your server hardware vendor are in source rpm format, you may need to install additional packages in order to compile the files into binary format. See the station hardware documentation for information.

1.3.2 Using the yum utility

To simplify package management, Nokia recommends that you use the RHEL yum utility to install and remove OS packages.

The package installation syntax is the following:

```
yum -y install package_1 package_2 ... package_n ↵
```

The package removal syntax is the following:

```
yum -y remove package_1 package_2 ... package_n ↵
```

Note: Package installation using yum requires a yum repository. The following repository types are available:

- local repository, which you can create during the RHEL OS installation
- Internet-based repository, which you can access after you register with the Red Hat Network

See the RHEL documentation for information about setting up a yum repository.

Note: If a package has dependencies on one or more additional packages that are not listed in the package documentation, the yum utility installs the additional packages.

1.3.3 Required RHEL environment and OS packages

During the RHEL OS installation for a CLM server, you must do the following.

1. Specify “Minimal Install” as the Software Selection in the RHEL installer.

2. Install specific OS packages, as described in 1.3.4 “RHEL OS packages to install” (p. 8).
3. Remove specific OS packages, as described in 1.3.5 “RHEL OS packages to remove” (p. 14).
4. Upgrade or install specific OS packages, as required, depending on the RHEL version; see 1.3.6 “Special RHEL OS package requirements” (p. 15).
5. Optionally, install one or more packages listed in 1.3.7 “Optional RHEL OS packages” (p. 21).
6. Reboot the station, as described in 1.3.8 “Reboot station” (p. 21).

1.3.4 RHEL OS packages to install

CAUTION

Risk of excessive resource consumption

The RHEL gnome desktop may consume excessive memory and result in system performance degradation.

The CLM does not require the gnome desktop, which is provided for customer and support convenience. It is recommended that you disable the gnome desktop on CLM if you do not require the gnome desktop.

You can stop the gnome desktop using the following command as the root user:

```
systemctl gdm stop ↵
```

To disable the gnome desktop so that it does not start after a station reboot, enter the following as the root user:

```
systemctl disable gdm ↵
```

You must install a set of RHEL OS packages that are common to each CLM server. Most of the common packages are available from the RHEL ISO disk image and the default RHEL package repository. Such packages are listed in “Required packages, RHEL ISO image or default RHEL repository” (p. 9).

You must also install additional packages that are available only from the RHEL optional package repository. Such packages are listed in “Required packages, RHEL optional package repository” (p. 13).

Required packages, RHEL ISO image or default RHEL repository

The RHEL ISO image and default package repository each contain the following OS packages that you must install. To facilitate the installation, copy the following command block and paste it in a CLI:

```
yum -y install @base @gnome-desktop @legacy-x @x11

yum -y install autofs bc.x86_64 binutils.x86_64 compat-libcap1.x86_64

yum -y install cups-client.x86_64

yum -y install cups-libs.x86_64
```

```
yum -y install dialog elfutils-libelf-devel.x86_64 elfutils.x86_64

yum -y install firefox.x86_64 ftp gcc.x86_64 gcc-c++.x86_64 glibc.i686

yum -y install glibc.x86_64 glibc-devel.i686 glibc-devel.x86_64

yum -y install gtk2.i686 haproxy.x86_64 hdparm.x86_64 irqbalance.x86_64

yum -y install keepalived.x86_64 keyutils-libs-devel.x86_64

yum -y install krb5-devel.x86_64 ksh.x86_64 libaio.i686 libaio.x86_64

yum -y install libaio-devel.i686 libaio-devel.x86_64

yum -y install libcom_err-devel.x86_64 libffi-devel.x86_64 libgcc.i686

yum -y install libgcc.x86_64 libgcrypt-devel.x86_64

yum -y install libgpg-error-devel.x86_64 libibverbs.x86_64

yum -y install libkadm5.x86_64 libselinux-devel.x86_64

yum -y install libsepol-devel.x86_64 libstdc++.i686 libstdc++.x86_64

yum -y install libstdc++-devel.i686 libstdc++-devel.x86_64

yum -y install libverto-devel.x86_64 libXi.i686 libXi.x86_64

yum -y install libxml2-devel.x86_64 libxslt-devel.x86_64

yum -y install libXrender.i686 libXtst.i686 libXtst.x86_64 lshw.x86_64

yum -y install lsof.x86_64 make.x86_64 man mcelog net-snmp

yum -y install net-snmp-utils ntp numactl-devel.i686

yum -y install numactl-devel.x86_64 openssh.x86_64

yum -y install openssh-askpass.x86_64 openssh-clients.x86_64

yum -y install openssh-server.x86_64 openssl-devel.x86_64

yum -y install pcre-devel.x86_64 procps python-devel.x86_64

yum -y install redhat-lsb-core.x86_64

yum -y install redhat-lsb-submod-security.x86_64

yum -y install rsync.x86_64 tcpdump.x86_64 unzip.x86_64 which
```

```
yum -y install xinetd.x86_64 xz-devel.x86_64 zip.x86_64
```

Table 1-1 Required OS packages from default RHEL repository or ISO image

Package	Description
@base	Base package group
@gnome-desktop	Gnome package group
@legacy-x	Legacy X package group
@x11	X11 package group
autofs	A tool for automatically mounting and unmounting filesystems
bc.x86_64	GNU's bc (a numeric processing language) and dc (a calculator)
binutils.x86_64	A GNU collection of binary utilities
compat-libcap1.x86_64	Library for getting and setting POSIX.1e capabilities
cups-client.x86_64	CUPS printing system - client programs
cups-libs.x86_64	CUPS printing system - libraries
dialog	A utility for creating TTY dialog boxes
elfutils.x86_64	A collection of utilities and DSOs to handle compiled objects
elfutils-libelf-devel.x86_64	Development support for libelf
firefox.x86_64	Mozilla Firefox web browser
ftp	The standard UNIX FTP client
gcc.x86_64	Various compilers, for example, C, C++, Objective-C, and Java
gcc-c++.x86_64	C++ support for GCC
glibc.i686	The GNU libc libraries
glibc.x86_64	The GNU libc libraries
glibc-devel.i686	Object files for development using standard C libraries
glibc-devel.x86_64	Object files for development using standard C libraries
gtk2.i686	The GIMP ToolKit (GTK+), a library for creating GUIs for X
haproxy.x86_64	TCP/HTTP proxy and load balancer for high availability environments
hdparm.x86_64	Utility for displaying and/or setting hard disk parameters
irqbalance.x86_64	Daemon that evenly distributes IRQ load across multiple CPUs
keepalived.x86_64	Load balancer and high availability service
keyutils-libs-devel.x86_64	Development package for building Linux key management utilities
krb5-devel.x86_64	Development files needed to compile Kerberos 5 programs
ksh.x86_64	The Original ATT Korn Shell

Table 1-1 Required OS packages from default RHEL repository or ISO image (continued)

Package	Description
libaio.i686	Linux-native asynchronous I/O access library
libaio.x86_64	Linux-native asynchronous I/O access library
libaio-devel.i686	Development files for Linux-native asynchronous I/O access
libaio-devel.x86_64	Development files for Linux-native asynchronous I/O access
libcom_err-devel.x86_64	Common error description library
libffi-devel.x86_64	GCC development for FFI
libgcc.i686	GCC version 4.8 shared support library
libgcc.x86_64	GCC version 4.4 shared support library
libgcrypt-devel.x86_64	Development files for the libgcrypt package
libgpg-error-devel.x86_64	Development files for the libgpg-error package
libibverbs.x86_64	Core user space library that implements hardware abstracted verbs protocol
libkadm5.x86_64	Kerberos 5 Administrative libraries
libselinux-devel.x86_64	Header files and libraries used to build SELinux
libsepol-devel.x86_64	Header files and libraries used to build policy manipulation tools
libstdc++.i686	GNU Standard C++ Library
libstdc++.x86_64	GNU Standard C++ Library
libstdc++-devel.i686	Header files and libraries for C++ development
libstdc++-devel.x86_64	Header files and libraries for C++ development
libverto-devel.x86_64	Development files for libverto
libXi.i686	X.Org X11 libXi runtime library
libXi.x86_64	X.Org X11 libXi runtime library
libxml2-devel.x86_64	Libraries, includes, etc. to develop XML and HTML applications
libXrender.i686	X.Org X11 libXrender runtime library
libxslt-devel.x86_64	Development libraries and header files for libxslt
libXtst.i686	X.Org X11 libXtst runtime library
libXtst.x86_64	X.Org X11 libXtst runtime library
lshw.x86_64	Hardware lister
lsof.x86_64	Provides a utility to list information about open files
make.x86_64	GNU tool which simplifies the build process for users
man	A set of documentation tools: man, apropos and whatis
mcelog	Tool to translate x86-64 CPU Machine Check Exception data

Table 1-1 Required OS packages from default RHEL repository or ISO image (continued)

Package	Description
net-snmp	SNMP Agent Daemon and documentation
net-snmp-utils	SNMP clients such as snmpget and snmpwalk
ntp	The NTP daemon and utilities
numactl-devel.i686	Development package for building Applications that use numa
numactl-devel.x86_64	Development package for building Applications that use numa
openssh.x86_64	Open source implementation of SSH protocol versions 1 and 2
openssh-askpass.x86_64	Passphrase dialog for OpenSSH and X
openssh-clients.x86_64	Open-source SSH client application
openssh-server.x86_64	Open source SSH server daemon
openssl-devel.x86_64	Files for development of applications which will use OpenSSL
pcre-devel.x86_64	Development files for PCRE
procps	OS utilities for /proc
python-devel.x86_64	The libraries and header files needed for Python development
redhat-lsb-core.x86_64	LSB Core module support
redhat-lsb-submod-security.x86_64	LSB Security submodule support
rsync.x86_64	A program for synchronizing files over a network
tcpdump.x86_64	Command-line packet analyzer and network traffic capture; used by technical support for debugging
unzip.x86_64	A utility for unpacking zip files
which	Displays where a particular program in your path is located
xinetd.x86_64	A secure replacement for inetd
xz-devel.x86_64	Development libraries and headers for liblzma
zip.x86_64	A file compression utility

Required packages, RHEL optional package repository

The RHEL optional package repository contains the OS packages listed in Table 1-2, “Required OS packages from RHEL optional package repository” (p. 13) that you must install. To facilitate the installation, copy the following command and paste it in a CLI:

```
yum -y install compat-libstdc++-33.i686 compat-libstdc++-33.x86_64
```

Table 1-2 Required OS packages from RHEL optional package repository

Package name	Description
compat-libstdc++-33.i686	Compatibility standard C++ libraries

Table 1-2 Required OS packages from RHEL optional package repository (continued)

Package name	Description
compat-libstdc++-33.x86_64	Compatibility standard C++ libraries

1.3.5 RHEL OS packages to remove

After you install the required OS packages on a CLM server station, you must remove packages that are installed by default but not required by the CLM.

For all RHEL 7 versions, you must remove the packages listed in Table 1-3, “RHEL OS packages to remove, all RHEL versions” (p. 15).

To facilitate the package removal, copy the following command block and paste it in a CLI:

```
yum -y remove anaconda-core.x86_64 anaconda-gui.x86_64

yum -y remove anaconda-tui.x86_64 avahi.x86_64 biosdevname

yum -y remove dnsmasq.x86_64 gnome-boxes.x86_64

yum -y remove initial-setup.x86_64 initial-setup-gui.x86_64

yum -y remove libstoragemgmt.x86_64 libstoragemgmt-python.noarch

yum -y remove libvirt-daemon-config-network.x86_64

yum -y remove libvirt-daemon-driver-network.x86_64

yum -y remove libvirt-daemon-driver-qemu.x86_64

yum -y remove libvirt-daemon-kvm.x86_64 libvirt-gconfig.x86_64

yum -y remove libvirt-gobject.x86_64

yum -y remove NetworkManager-libreswan.x86_64

yum -y remove NetworkManager-libreswan-gnome.x86_64

yum -y remove NetworkManager-team.x86_64 NetworkManager-tui.x86_64

yum -y remove qemu-kvm.x86_64 qemu-kvm-common.x86_64

yum -y remove setroubleshoot.x86_64 setroubleshoot-plugins.noarch

yum -y remove setroubleshoot-server.x86_64

yum -y remove subscription-manager-initial-setup-addon.x86_64
```

Table 1-3 RHEL OS packages to remove, all RHEL versions

Package	Description
anaconda-core.x86_64	Core of the Anaconda installer
anaconda-gui.x86_64	Graphical user interface for the Anaconda installer
anaconda-tui.x86_64	Textual user interface for the Anaconda installer
avahi.x86_64	Local network service discovery
biosdevname	Utility that provides an optional convention for naming network interfaces
dnsmasq.x86_64	A lightweight DHCP/caching DNS server
gnome-boxes.x86_64	A simple GNOME 3 application to access remote or virtual systems
initial-setup.x86_64	Initial system configuration utility
initial-setup-gui.x86_64	Graphical user interface for the initial-setup utility
libstoragemgmt.x86_64	Storage array management library
libstoragemgmt-python.noarch	Python2 client libraries and plug-in support for libstoragemgmt
libvirt-daemon-config-network.x86_64	Default configuration files for the libvirtd daemon
libvirt-daemon-driver-network.x86_64	Network driver plugin for the libvirtd daemon
libvirt-daemon-driver-qemu.x86_64	Qemu driver plugin for the libvirtd daemon
libvirt-daemon-kvm.x86_64	Server side daemon & driver required to run KVM guests
libvirt-gconfig.x86_64	libvirt object APIs for processing object configuration
libvirt-gobject.x86_64	libvirt object APIs for managing virtualization hosts
NetworkManager-libreswan.x86_64	NetworkManager VPN plugin for libreswan
NetworkManager-libreswan-gnome.x86_64	NetworkManager VPN plugin for libreswan - GNOME files
NetworkManager-team.x86_64	Team device plugin for NetworkManager
NetworkManager-tui.x86_64	NetworkManager curses-based UI
qemu-kvm.x86_64	QEMU metapackage for KVM support
qemu-kvm-common.x86_64	QEMU common files needed by all QEMU targets
setroubleshoot.x86_64	Helps troubleshoot SELinux problem
setroubleshoot-plugins.noarch	Analysis plugins for use with setroubleshoot
setroubleshoot-server.x86_64	SELinux troubleshoot server
subscription-manager-initial-setup-addon.x86_64	Initial setup screens for subscription manager

1.3.6 Special RHEL OS package requirements

The CLM requires:

- specific versions of some packages; see “Special package version requirements, all RHEL versions” (p. 15)
- for RHEL 7.3 or 7.4, the removal of packages not listed in Table 1-3, “RHEL OS packages to remove, all RHEL versions” (p. 15); see “Special RHEL 7.3 or 7.4 package requirements” (p. 16)
- for RHEL 7.6, the installation or removal of specific packages, as described in “Special RHEL 7.6 or 7.7 package requirements” (p. 17)

Special package version requirements, all RHEL versions

The CLM requires the version of each RHEL 7 package quoted in Table 1-4, “Required RHEL OS package versions” (p. 15), or a later version. After the initial OS installation, if a listed package version is lower than the minimum required, you must upgrade the package.

To facilitate the package upgrade, copy the following command block and paste it in a CLI:

```
yum -y install nspr.x86_64 nss-softokn-freebl.i686

yum -y install nss-softokn-freebl.x86_64 nss-softokn.x86_64

yum -y install nss-util.x86_64
```

Table 1-4 Required RHEL OS package versions

Package	Minimum version required
nspr.x86_64	4.19.0-1.el7
nss-softokn-freebl.i686	3.36.0-5.el7
nss-softokn-freebl.x86_64	3.36.0-5.el7
nss-softokn.x86_64	3.36.0-5.el7
nss-util.x86_64	3.36.0-1.el7

Special RHEL 7.3 or 7.4 package requirements

NOTICE

RHEL 7.3 requirement

A CLM system on RHEL 7.3 requires openssl-devel.x86_64 version 1.0.2k or later.

Before you can upgrade a CLM on RHEL 7.3, you must ensure that the openssl-devel.x86_64 package is at a supported version.

For RHEL 7.3 or 7.4, in addition to the packages listed in 1.3.5 “RHEL OS packages to remove” (p. 14), you must also remove the packages listed in Table 1-5, “OS packages to remove, RHEL 7.3 or 7.4” (p. 17).

To facilitate the package removal, copy the following command and paste it in a CLI:

```
yum -y remove NetworkManager.x86_64 NetworkManager-wifi.x86_64
```

Note: The packages are required by RHEL 7.5 and later because of a package dependency introduced in RHEL 7.5.

Table 1-5 OS packages to remove, RHEL 7.3 or 7.4

Package	Description
NetworkManager.x86_64	Network connection manager and user applications
NetworkManager-wifi.x86_64	Wifi plugin for NetworkManager

Special RHEL 7.6 or 7.7 package requirements

For RHEL 7.6 or 7.7, you must do the following:

- Install the packages listed in Table 1-6, “Additional OS packages, RHEL 7.6 or 7.7” (p. 17).
- Remove the packages listed in Table 1-7, “OS packages to remove, RHEL 7.6 or 7.7” (p. 19).

Table 1-6, “Additional OS packages, RHEL 7.6 or 7.7” (p. 17) lists the additional packages that you must install for RHEL 7.6 or 7.7.

To facilitate the installation, copy the following command block and paste it in a CLI:

```
yum -y install bpftool c-ares gcc-gfortran hyphen-en
yum -y install javapackages-tools.noarch libcurl-devel.x86_64
yum -y install libgfortran libquadmath libquadmath-devel
yum -y install libverto-libevent libyaml-devel.x86_64 nfs-utils orca
yum -y install python-babel python-javapackages python-jinja2
yum -y install python-jsonpatch python-jsonpointer python-markupsafe
yum -y install python-paramiko python-pillow python-prettytable
yum -y install python-pygments python-requests python-urllib3
yum -y install python2-oauthlib socat
yum -y install tigervnc-server.x86_64
```

Table 1-6 Additional OS packages, RHEL 7.6 or 7.7

Package	Description
bpftool	Inspection and simple manipulation of eBPF programs and maps
c-ares	A library that performs asynchronous DNS operations
gcc-gfortran	Fortran 95 support for gcc
hyphen-en	English hyphenation rules
javapackages-tools.noarch	Macros and scripts for Java packaging support
libcurl-devel.x86_64	A Tool for Transferring Data from URLs
libgfortran	Fortran runtime
libquadmath	GCC __float128 shared support library
libquadmath-devel	GCC __float128 support
libverto-libevent	libevent module for libvert

Table 1-6 Additional OS packages, RHEL 7.6 or 7.7 (continued)

Package	Description
libyaml-devel.x86_64	Development files for LibYAML applications
nfs-utils	NFS utilities and supporting clients and daemons for the kernel NFS server
orca	GNOME screen reader for people with visual impairments
python-babel	Internationalization utilities
python-javapackages	Module for handling various files for Java packaging
python-jinja2	Python template engine
python-jsonpatch	Apply JSON-Patches (RFC 6902)
python-jsonpointer	Identify specific nodes in a JSON document (RFC 6901)
python-markupsafe	XML/HTML/XHTML markup safe string package for Python
python-paramiko	SSH2 protocol library
python-pillow	Python image processing library
python-prettytable	Python library for displaying data in ASCII table format
python-pygments	Syntax highlighting package written in Python
python-requests	Python HTTP for Humans
python-urllib3	HTTP library with thread-safe connection pooling, file post, and more
python2-oauthlib	A Generic Implementation of the OAuth Request-Signing Logic
socat	Multipurpose relay for bidirectional data transfer
tigervnc-server.x86_64	Server for the VNC remote display system

Table 1-7, “OS packages to remove, RHEL 7.6 or 7.7” (p. 19) lists the packages to remove from RHEL 7.6 or 7.7.

To facilitate the package removal, copy the following command block and paste it in a CLI:

```
yum -y remove anaconda-user-help anaconda-widgets
yum -y remove cryptsetup-python cyrus-sasl cyrus-sasl-gssapi
yum -y remove daxctl-libs fcoe-utils glade-libs glusterfs-cli
yum -y remove gnome-initial-setup ipxe-roms-qemu
yum -y remove iscsi-initiator-utils iscsi-initiator-utils-iscsiuio
yum -y remove isomd5sum kernel-devel keybinder3 ldns
yum -y remove libblockdev-nvdimm libconfig libgovirt libnm-gtk
yum -y remove librdmacm libreport-anaconda libreport-plugin-bugzilla
yum -y remove libreport-rhel-anaconda-bugzilla libreswan
yum -y remove libtimezonemap libuser-python
yum -y remove libverto-tevent lldpad mtools ndctl ndctl-libs
yum -y remove netcf-libs nmap-ncat numad oddjob oddjob-mkhomedir
yum -y remove pykickstart pyparted pyserial
yum -y remove python-blivet python-coverage python-di
yum -y remove python-meh-gui python-nss python-ntplib
```

```
yum -y remove python-pwquality python-pyblock python2-blockdev
yum -y remove python2-subprocess32 pytz radvd realmd seabios-bin
yum -y remove seavgabios-bin sgabios-bin spice-server
yum -y remove unbound-libs yajl
```

Table 1-7 OS packages to remove, RHEL 7.6 or 7.7

Package	Description
anaconda-user-help	Anaconda built-in help system
anaconda-widgets	A set of custom GTK+ widgets for use with anaconda
cryptsetup-python	Python bindings for libcryptsetup
cyrus-sasl	Implementation of Cyrus SASL API
cyrus-sasl-gssapi	Plugin for the GSSAPI SASL mechanism
daxctl-libs	Management library for "Device DAX" devices
fcoe-utils	FCoE userspace management tools
glade-libs	Widget library for Glade UI designer
glusterfs-cli	GlusterFS CLI
gnome-initial-setup	GNOME Initial Setup Assistant
ipxe-roms-qemu	Network boot loader roms supported by QEMU, .rom format
iscsi-initiator-utils	iSCSI daemon and utility programs
iscsi-initiator-utils-iscsiuio	Userspace configuration daemon required for some iSCSI hardware
isomd5sum	Utilities for working with md5sum implanted in ISO images
kernel-devel	Development files needed for building kernel modules
keybinder3	A library for registering global keyboard shortcuts
ldns	A library for developing the Domain Name System
libblockdev-nvdimm	The NVDIMM plugin for the libblockdev library
libconfig	C++ bindings development files for libconfig
libgovirt	A GObject library for interacting with oVirt REST API
libnm-gtk	Private libraries for NetworkManager GUI support
librdmacm	Userspace RDMA Connection Manager
libreport-anaconda	Default configuration for reporting anaconda bugs
libreport-plugin-bugzilla	libreport's bugzilla plugin
libreport-rhel-anaconda-bugzilla	Default configuration for reporting anaconda bugs to Red Hat Bugzilla
libreswan	IPsec implementation with IKEv1 and IKEv2 keying protocols
libtimezonemap	Time zone map widget for Gtk+
libuser-python	Python bindings for the libuser library

Table 1-7 OS packages to remove, RHEL 7.6 or 7.7 (continued)

Package	Description
libverto-tevent	Python bindings for the libuser library
lldpad	Intel LLDP Agent
mtools	Tools to access MS-DOS filesystems without kernel drivers
ndctl	Manage "libnvdimm" subsystem devices (Non-volatile Memory)
ndctl-libs	Management library for "libnvdimm" subsystem devices (Non-volatile Memory)
netcf-libs	Libraries for netcf
nmap-ncat	Nmap's Netcat replacement
numad	Userspace daemon that automatically binds workloads to NUMA node
oddjob	A D-Bus service which runs odd jobs on behalf of client applications
oddjob-mkhomedir	An oddjob helper which creates and populates home directories
pykickstart	A python library for manipulating kickstart files
pyparted	Python module for GNU parted
pyserial	Python serial port access library
python-blivet	A python module for system storage configuration
python-coverage	Code coverage measurement for Python
python-di	Python library for dependency injection support
python-meh-gui	Graphical user interface for the python-meh library
python-nss	Python bindings for Network Security Services (NSS)
python-ntplib	Python module that offers a simple interface to query NTP servers
python-pwquality	Library for password quality checking -- Python bindings
python-pyblock	Python modules for dealing with block devices
python2-blockdev	Python2 gobject-introspection bindings for libblockdev
python2-subprocess32	Backport of subprocess module from Python 3.2 to Python 2
pytz	World Timezone Definitions for Python
radvd	A Router Advertisement daemon
realmd	Kerberos realm enrollment service
seabios-bin	Seabios for x86
seavgabios-bin	Seavgabios for x86
sgabios-bin	Sgabios for x86
spice-server	Implements the server side of the SPICE protocol
unbound-libs	Libraries used by the unbound server and client applications

Table 1-7 OS packages to remove, RHEL 7.6 or 7.7 (continued)

Package	Description
yajl	Yet Another JSON Library Tools

1.3.7 Optional RHEL OS packages

Table 1-8, “Optional RHEL OS packages” (p. 21) lists the optional packages that you can install. To facilitate the package installation, copy the following command and paste it in a CLI:

```
yum -y install nfs-utils
```

Table 1-8 Optional RHEL OS packages

Package	Description
nfs-utils	NFS utilities and supporting clients and daemons for the kernel

1.3.8 Reboot station

After you complete the OS installation, reboot the station before installing the CLM software. Enter the following command as the root user:

systemctl reboot ↵

1.4 Virtual machine requirements

1.4.1 Overview

Nokia recommends that the CLM be installed on virtual machines using VMWare ESXi or RHEL KVM, including OpenStack. The Guest Operating System for a CLM deployment must be a supported version of RHEL 7.3, 7.4, 7.5, 7.6 or 7.7 Server x86-64.

Installations of CLM are server- and vendor-agnostic, but must meet any defined hardware criteria and performance targets to be used with the CLM. Server class hardware must be used, not desktops. Processors must be x86-64 based with a minimum core speed of 2.4GHz.

Defined CPU and Memory resources for a virtual machine must be reserved and dedicated to that guest OS, and cannot be shared or oversubscribed. Disk and network resources should be managed appropriately to ensure that other guest OSs on the same physical server do not negatively impact the operation of the CLM.

Provisioned CPU resources are based upon the CLM hardware platform requirements. Virtual machines should be configured with all vCPUs on one virtual socket

A guest virtual machine must use only one time synchronization protocol such as NTP. Additional time synchronization applications must be disabled to ensure the proper operation of CLM.

Nokia support personnel must be provided with the details of the provisioned Virtual Machine. These details can either be provided through read-only access to the hypervisor or must be available to Nokia support when requested. Failure to provide these details could impact support of the CLM.

1.5 VMware Virtualization

1.5.1 Overview

The CLM supports using VMware vSphere ESXi 6.0, 6.1, or 6.5 only, on x86 based servers natively supported by ESXi. VMware's Hardware Compatibility List (HCL) should be consulted to determine specific hardware support.

Not all features offered by ESXi are supported when using the CLM. For example, Fault Tolerant, High Availability (HA), Memory Compression, and Distributed Resource Scheduler (DRS) features are not supported. Contact Nokia to determine if a specific ESXi feature is supported with an CLM installation.

If using NTP or a similar time synchronization protocol on the guest virtual machine, then you must disable VMwareTools time synchronization.

Virtual Machine Version 11 or above must be used. The disk must be "Thick Provisioned" with "Eager Zero" set. The SCSI controller must be set to "VMware Paravirtual" and the Disk Provisioning must be "Thick Provision Eager Zero". The Network Adapter must be "VMXNET 3". See the following table for additional Virtual Machine setting requirements:

Table 1-9 Additional Virtual Machine setting requirements

Resource type	Parameter	Setting
CPU	Shares	Set to High
	Reservation	Must be set to half the number of vCPUs * the CPU frequency. For example, on a 2.4 GHz 8 vCPU configuration, the reservation must be set to (1/2*8*2400) = 9600 MHz.
	Limit	Check box checked for unlimited
Memory	Shares	Set to High
	Reservation	Slider set to the size of the memory allocated to the VM
	Limit	Check box checked for unlimited
Disk	Shares	Set to High
	Limit — IOPs	Set to Unlimited

1.6 KVM virtualization

1.6.1 Overview

The CLM supports using RHEL 6.3 through 6.7 KVM using QEMU version 0.12.1.2 and RHEL 7.2 through 7.5 KVM using QEMU version 1.5.3, 2.0.0, 2.3.0, or 2.10.0 only, on x86 based servers natively supported by KVM. Consult the RHEL's Hardware Compatibility List (HCL) to determine specific hardware support.

Not all features offered by KVM are supported when using the CLM. For example, Live Migration, Snapshots, or High Availability are not supported. Contact Nokia to determine if a specific KVM feature is supported with a CLM installation.

1.6.2 Configuration

When you configure the KVM, set the parameters listed in the following table to the required values.

Table 1-10 KVM configuration parameters

Parameter	Value
Disk Controller type	virtio
Storage format	raw
Cache mode	none
I/O mode	native
I/O scheduler	deadline
NIC device model	virtio
Hypervisor type	kvm

1.7 OpenStack requirements

1.7.1 OpenStack support

The CLM supports deployment in an OpenStack environment using Red Hat OpenStack Platform Release 8, 10, and 11. While a CLM installation may function in other OpenStack environments, the CLM Product Group does not commit to make the CLM compatible with a customer's alternate OpenStack environment.

To ensure the stability of the CLM and compatibility with OpenStack, you must follow the recommendations provided in this section.

1.7.2 Hypervisor

The only hypervisor supported within an OpenStack environment is KVM. For details about the KVM hypervisor supported versions, see 1.6 "KVM virtualization" (p. 23).

1.7.3 CPU and memory resources

Defined CPU and memory resources must be reserved and dedicated to the individual Guest OSs, and cannot be shared or oversubscribed. You must set both the cpu_allocation_ratio and ram_allocation_ratio parameters to 1.0 in the OpenStack Nova configuration either on the control NE or on each individual compute node where a VM hosting the CLM could reside.

1.7.4 HyperThreading

The usage of CPUs with enabled HyperThreading must be consistent across all compute nodes. If there are CPUs that do not support HyperThreading, then you must disable HyperThreading at the hardware level on all compute nodes where the CLM could be deployed.

1.7.5 CPU pinning

Nokia recommends enabling CPU pinning because it restricts the use of OpenStack migration. The CLM is node locked.

1.7.6 Availability zones/affinity/placement

Nokia does not provide recommendations on configuring OpenStack for VM placement.

1.7.7 Networking

Basic Neutron functionality using Open vSwitch with the ML2 plugin can be used in a deployment of CLM. The use of OpenStack floating IP addresses is supported for CLM.

1.7.8 VM storage

The VM storage must be persistent block (Cinder) storage and not ephemeral. For each VM to be deployed, a bootable Cinder volume must be created.

1.7.9 Firewalls

Firewalls can be enabled using OpenStack Security Groups, or on the VMs using the firewalld service. If firewalld is enabled, then an OpenStack Security Group that allows all incoming and outgoing traffic must be used.

1.8 Platform requirements

1.8.1 Minimum hardware platform requirements

CLM supports up to 1000 network functions. The hardware requirements are independent of the number of network functions. The following table lists the minimum hardware platform requirements for the deployment of CLM for RHEL x86-64 operating system.

Table 1-11 CLM hardware platform requirements

Hardware	Requirement
CPU cores	2 (minimum 2.4 GHz)

Table 1-11 CLM hardware platform requirements (continued)

Hardware	Requirement
Memory	minimum 16 GB
Disk	1 SAS 10K RPM drive, 200 GB or more

1.9 Partitioning

1.9.1 Partitioning requirements

CAUTION

Service Disruption

Each disk partition described in this section must be a mounted partition and not a symbolic link.

The CLM does not support the use of symbolic links to represent partitions.

Table 1-12, “CLM servers partitioning scheme” (p. 25) lists the partitioning requirements for CLM components in both live and lab deployments.

Table 1-12 CLM servers partitioning scheme

Partition	Content	Size (Gbytes)
swap	Swap space	16
/	Root	26
/home	User home directories	0.5
/tmp	Temporary files	6
/var	System data	14
/var/log	System logs	6
var/log/audit	System audit logs	6
/opt/nsp	CLM software, operating data, and backups	90
/opt/nsp/os	nspOS software, operating data, and backups	90
/extra	Application software, etc	15

1.10 Securing the CLM

1.10.1 Overview

Nokia recommends that you to perform the following steps to achieve workstation security for the CLM:

- Install the latest recommended patch cluster from Red Hat
- Implement firewall rules to control access to ports on CLM systems, as detailed below
- Use a CA signed certificate rather than a self-signed certificate.
- Use SSL certificates with strong hashing algorithms.
- Enforce minimum password requirements and password renewal policies on user accounts
- Configure Launchpad Security Statement.
- Configure login throttling to prevent denial of service attacks.
- Configure maximum session limits for administrators and users.
- Configure user lockout after a threshold of consecutive failed login attempts.

Communications is secured using TLS. By default, the CLM only supports TLSv1.2; TLS 2.0 is not supported. If you need to use other TLS versions (for example TLSv1.1, and TLSv1.0), then follow the steps in the new procedure 5.12 “To update the supported CLM TLS versions and ciphers” (p. 67).

1.11 Operating system security for CLM workstations

1.11.1 RHEL patches

Nokia supports customers applying RHEL patches provided by Red Hat which will include security fixes as well as functional fixes. If a patch is found to be incompatible with the CLM, the patch may need to be removed until a solution to the incompatibility is provided by Red Hat or Nokia.

1.11.2 Platform hardening

Additional efforts to secure the system could impact CLM operation or future upgrades of the product. Customers must perform some level of basic testing to validate additional platform hardening does not impact the operation of the CLM. The CLM Product Group makes no commitment to make the CLM compatible with a customer's hardening requirements.

1.12 Port information

1.12.1 Overview

The tables provided in this section identify the listening ports in the CLM.

The CLM deployment types are:

- standalone
- redundant

Table 1-13 Port information for CLM

Default port(s)	Type	Encryption	Description	CLM deployment
All applications				
22	TCP	Dynamic Encryption	SSH/SCP/SFTP Used for remote access and secure file transfer	All
nspOS				
80	TCP	None	HTTP port for nspOS common applications, redirect to 443	All
443	TCP	Dynamic Encryption provided by TLS	Secure HTTPS port for nspOS common applications	All
2181	TCP	None	Zookeeper (unsecure) Enabled by default	All
2281	TCP	Dynamic Encryption provided by TLS	Zookeeper (secure) Disabled by default	All
2390	TCP	Dynamic Encryption provided by TLS	nspdctl	All
6432	TCP	Dynamic Encryption provided by TLS	PostgreSQL database	All
7983	TCP	None	Solr (Help) Local port to the host	All
8195	TCP	None	tomcat shutdown port Local port to the host	All
8196	TCP	None	CLM shutdown port Local port to the host	All
8544	TCP	Dynamic Encryption provided by TLS	HTTPS port for CLM	All
8575	TCP	Dynamic Encryption provided by TLS	System Token Server	All
8983	TCP	None	Solr (Help) Local port to the host.	All

Table 1-13 Port information for CLM (continued)

Default port(s)	Type	Encryption	Description	CLM deployment
9092	TCP	None	Kafka server (unsecure) Enabled by default	All
9192	TCP	Dynamic Encryption provided by TLS	Kafka server (secure) Disabled by default	All
47100–47199	TCP	None	CAS ignite cache	All
47500–47599	TCP	None	CAS ignite cache	All
48500–48599	TCP	Dynamic Encryption provided by TLS	session-manager ignite cache	All
48600–48699	TCP	Dynamic Encryption provided by TLS	session-manager ignite cache	All
PKI server				
2391	TCP	None	PKI server	Only where PKI server is installed and running

1.13 CLM firewall rules and NAT

1.13.1 Overview

A firewall can be deployed to protect the CLM from different networks and applications. Firewall rules are applied to the incoming network interface traffic of the CLM workstations. As a rule, firewall rules are not applied to the outgoing network traffic.

The firewall rules to be applied to a CLM deployment will depend on the deployment configuration and network topology.

Some CLM operations require idle TCP ports to remain open for long periods of time. Therefore, a customer firewall that closes idle TCP connections should adjust OS TCP keep-alives to ensure that the firewall will not close sockets that are in use by the CLM.

The CLM supports the use of Network Address Translation (NAT) between themselves and client applications (API and GUI).

Communications with Zookeeper and Kafka are secure by default and the firewall tables below will reflect this configuration. Where unsecure Zookeeper and Kafka is configured, the corresponding unsecure port numbers for those components should be substituted.

1.13.2 CLM and nspOS firewall rules

Firewall rules need to be applied bidirectionally.

Table 1-14 Firewall rules for traffic between the active and standby CLM in a redundant deployment

Protocol	From port	To port
TCP	>32768	22
TCP	>32768	2390
TCP	>32768	6432

Table 1-15 Firewall rules for traffic between the CLM and PKI server

Protocol	From port	From component	To port	To component
TCP	>32768	CLM	2391	PKI server

Table 1-16 Firewall rules for traffic between CLM and client (GUI/REST) applications

Protocol	To port	To component	Purpose
TCP	80	CLM / nspOS	re-directs to port 443 (Launchpad)
TCP	443	CLM / nspOS	for Launchpad
TCP	8544	CLM	CLM GUI and REST API
TCP	9092	CLM / nspOS	External notifications (messaging)

Table 1-17 Firewall rules for CLM to NE communications

NE type	Protocol	Port
NEs that use manual pool capacity reservation (VSR, 7x50 SR/XRS, CMG)	Telnet	23
NEs that use manual pool capacity reservation (VSR, 7x50 SR/XRS, CMG)	SSH	22
NEs that use network function driven pool capacity reservation (1830 PSS, Release 12 and later)	REST API over HTTPS	8544

Table 1-18 Firewall rules for remote user authentication

Protocol	From port	On	To port	On	Notes
TCP/UDP	Any	CLM	49	TACAS server	For TACAS authentication
TCP/UDP	Any	CLM	389	LDAP server	For LDAP authentication
TCP/UDP	Any	CLM	636	LDAP server	For LDAP authentication (TLS)
UDP	Any	CLM	1812	RADIUS server	For RADIUS authentication

2 Standalone installation and upgrade

2.1 Introduction

2.1.1 Overview

This chapter describes the standalone CLM installation and upgrade processes, as well as related operations.

2.1.2 Hosts file

A hosts file identifies the CLM server(s) that host the components of your deployment. This file must be created during CLM server installation. Depending on the configuration of your deployment, the host file is populated with one of more of the entries in the following table.

Table 2-1 Hosts file sections and parameters

Deployed component	Required hosts file entry
nspOS + CLM (standalone)	**[nspos]** ***IPaddress*** **[clm]** ***IPaddress*** Where *IPaddress* is the IP address of the server that is to host the nspOS software will be installed.
nspOS + CLM (1+1 redundancy)	**[nspos]** **<primary server address> dc=<locationA>** **<standby server address> dc=<locationB>** **[clm]** **<primary server address> dc=<locationA>** **<standby server address> dc=<locationB>** where *primary server address* is the IP address of the primary server *standby server address* is the IP address of the standby server *location* is the datacenter in which the given server resides. This string must be unique to each server in the redundant deployment

2.1.3 Configuration file

A configuration file is used to configure a CLM server to perform specific functions. This file must be created during CLM server installation. Of the following configuration blocks, add only those that apply to your CLM server, based on the components that it will host. See 2.1.4 “SSO configuration file parameters” (p. 33) for a list of parameters available in the SSO block of the configuration file.

Based on your requirements, you must edit the sections of the configuration file that apply to your deployment.

Based on your requirements, you must edit the sections of the configuration file that apply to your deployment; an example configuration file is in the following directory:

NSP_installer_directory/NSD_NRC_*R_r*/examples

If you use the example configuration file, you must comment out the portions that are not used, nfmp, nfmt, nrct, sros.

A line in the file that begins with ## is a comment line, and is not to be modified. A line that begins with # is configurable.

To enable a section and the required parameters in the section, you must do the following:

1. Remove the leading # character from the section label.
2. Remove the leading # character from each parameter that you need to configure.
3. Enter the required value for each parameter, as described in the comment lines above the section label.

Note: It is recommended that you remove parameter lines that you do not require from the configuration file. Use the default values, where listed. If there is no default value listed, you must specify a value. Also, failing to provide a parameter value may have undesired consequences.

Table 2-2 Configuration file parameters

Section and parameters	Description	
auto_start	Whether CLM starts automatically after installation Default: true	
nspos — inter-component communication parameters		
rest	System-wide REST parameters	
session	ttlInMins	REST token time to live, in minutes Default: 60
	maxNumber	Maximum number of concurrent REST session tokens Default: 50
secure	Whether internal service communication between CLM components is secured using TLS You must set this to true for CLM. Default: false	
tls — TLS parameters		
pki_server	PKI server IP address or hostname Default: none	
pki_server_port	PKI server port Default: 2391	

Table 2-2 Configuration file parameters (continued)

Section and parameters	Description
pki_org	Organization name for TLS certificate Default: Nokia
pki_cn	Common name for TLS certificate Default: NSP
custom_keystore_path	If you are providing custom TLS keystore, keystore path and filename Default: none
custom_truststore_path	If you are providing custom TLS truststore, truststore path and filename Default: none
custom_keystore_password	If you are providing custom TLS keystore, keystore password Default: none
custom_truststore_password	If you are providing custom TLS truststore, truststore password Default: none
custom_key_alias	If you are providing custom TLS keystore, alias of required key in keystore Default: alias
regenerate_certs	Whether to force TLS certificate regeneration Default: true
ean — external application notification parameters	
max_subscribers	Maximum number of subscribers that receive external application notifications Default: 10

Note: Parameters not being configured should be removed from the configuration file entirely. Failing to provide a value for a parameter may have undesired consequences.

2.1.4 SSO configuration file parameters

The SSO section of the configuration file is used to define CLM remote user authentication sources and login features.

Note: TLS certificates for secure LDAP communication must be copied to the /tls/ldap directory in the installation directory. If an LDAP certificate contains an IP address or hostname in the SAN field, the same IP address or hostname must be used in the config.yml file.

Table 2-3 SSO configuration file parameters

Section and parameters	Description
session — general user session parameters	

Table 2-3 SSO configuration file parameters (continued)

Section and parameters			Description
concurrent_limits_enabled			Whether a maximum concurrent session limit is enabled Default: true
max_sessions_per_user			Maximum number of concurrent sessions per user - does not apply to admin group Default: 10
max_sessions_for_admin			Maximum number of concurrent sessions for users in admin group Default: 10
ldap — LDAP parameters			
enabled			Whether LDAP is to be used for authentication Default: true
servers			List of LDAP servers; specify a server using the parameters below
	type		LDAP server type Default: AUTHENTICATION/AD/ANONYMOUS
	url		LDAP server URL with IP address or hostname and port Default: none
	security		Type of LDAP server security Default: SSL/STARTTLS/NONE
	timeout		Timeout period, in seconds, for receiving an authentication response Default: 10
	user_base_dn		User base dn value Default: *example_only*
	user_filter		Filter criteria for username Default: *example_only*
	group_base_dn		Group base dn value Default: *example_only*
	group_search		Custom group search options
		filter	Group search filter criteria Default: *example_only*
		attribute_id	Group search attribute on which to filter Default: *example_only*
	bind		LDAP bind credentials for authenticated access only
		dn	User with authority to bind to LDAP server Default: *example_only*

Table 2-3 SSO configuration file parameters (continued)

Section and parameters	Description	
	credential	Password for bind user **Note:** The password must be enclosed in double quotation marks. Default: *example_only*
min_pool_size	Minimum pool size Default: 0	
max_pool_size	Maximum pool size Default: 10	
use_entry_resolver	Whether an entry resolver is to be used for extracting additional user information Default: *example_only*	
radius — RADIUS parameters		
enabled	Whether RADIUS is to be used for authentication Default: none	
address	Comma-separated list of RADIUS servers Default: none	
secret	Shared server secret **Note:** The shared secret value must be enclosed in double quotation marks. Default: none	
protocol	Protocol to use—PAP or CHAP Default: none	
retries	Maximum number of attempts to reach server Default: 3	
timeout	Timeout, in seconds, for attempts to reach RADIUS server Default: 60	
failover_on_exception	Whether second server is tried if first server fails with exception Default: none	
failover_on_rejection	Whether second server is tried if first server fails with rejection Default: none	
authentication_port	RADIUS port Default: 1812	
vendor_id	Vendor ID for VSA search Default: 123	
role_VSA_id	VSA ID used to identify group Default: 3	
tacacs — TACACS+ parameters		

Table 2-3 SSO configuration file parameters (continued)

Section and parameters	Description
enabled	Whether TACACS+ authentication is to be used Default: none
address	Comma-separated list of TACACS+ servers Default: none
secret	Shared server secret **Note:** The shared secret must be enclosed in double quotation marks. Default: none
protocol	Protocol to use Default: PAP
timeout	Timeout, in seconds, for attempts to reach TACACS+ server Default: 7
failover_on_exception	Whether second server is tried if first server fails with exception Default: none
failover_on_rejection	Whether second server is tried if first server fails with rejection Default: none
authentication_port	TACACS+ port Default: 49
default_group	Default group to assign if no group defined on server Default: none
VSA_enabled	Whether VSA search is enabled Default: true
role_VSA_id	Role used for VSA search Default: sam-security-group
VSA_service_id	VSA search service identifier Default: sam-app
throttling — user login throttling parameters	
enabled	Whether to enable login throttling Default: none
rate_threshold	Login failure threshold used for calculating login failure rate; see rate_seconds parameter Default: 3
rate_seconds	Number of seconds used for calculating login failure rate; exceeded if login attempt comes within rate_seconds/rate_ threshold seconds of a previous failed login attempt Default: 9

Table 2-3 SSO configuration file parameters (continued)

Section and parameters	Description
lockout_period	Number of seconds after throttling threshold exceeded to wait before attempting to authenticate the same user and source address combination Default: 5
login_failure — user login failure parameters	
enabled	Whether to lock out users who have more consecutive login failures than specified by the threshold parameter Default: none
threshold	Maximum number of consecutive login failures before user lockout Default: 3
lockout_minutes	Number of minutes to lock the user out after the threshold parameter value is exceeded Default: 1

Note: Any certificates required for secure LDAP communications should be copied to <*nsp installer directory*>/ssl/ldap/. If an LDAP certificate contains its IP address or hostname in SAN field, that same IP address or hostname must be used in the config.yml file.

Note: If hostnames are used instead of IP addresses within the config.yml file, those hostnames need to be used in the hosts file as well.

2.2 To install a standalone CLM system

2.2.1 Purpose

Use this procedure to install a standalone CLM system.

2.2.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 1, “Pre-installation”.

2.2.3 Steps

CAUTION

Deployment failure

The RHEL OS requires specific versions of some RHEL packages. If the required package versions are not installed, the CLM installation fails.

See 1.3.6 “Special RHEL OS package requirements” (p. 15) for the required package versions.

1

Download the NSP_CLM_installer_<*release_load*>.tar.gz from the Nokia Support portal (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM, where <*release_load*> is the numbered CLM software release, such as 20_3 for release 20.3.

Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

An NSD_NRC_*R_r* directory is created in the current directory, where *R_r* is the CLM release identifier in the form *MAJOR_minor*.

Note: In subsequent steps, the directory is called the NSP installer directory or *NSP_installer_directory*.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from the Nokia Support portal. You can compare the checksum value on the download page with, for example, the output of the RHEL md5sum or sha256sum command. See the appropriate RHEL man page for information.

Note: The install.sh utility communicates with remote target stations using an SSH session. To perform an operation on a remote station using the utility, you must do one of the following.

- Configure the required SSH keys on the stations.
- Use the --ask-pass argument when you run install.sh, in which case each remote station must have the same root user password; for example:

  `./install.sh --ask-pass --target remote_station`

2

Enter the following to navigate to the NSP installer directory:

`cd NSD_NRC_R_r` ↵

3

Create a hosts file in the directory where the CLM installer bundle was extracted and add the required entries based on the components that the server will host. See 2.1.2 “Hosts file” (p. 31) for more information.

4

Create a YAML or JSON configuration file in the directory where the CLM installer bundle was extracted and add only the configuration blocks that apply to your deployment.

Note: You must set the secure parameter to true.

See 2.1.3 “Configuration file” (p. 31) for more information.

5

If the TLS block of the configuration file was populated in Step 4, copy the TLS certificates into the installer directory.

6

If LDAP authentication settings were configured in Step 4, copy the LDAP server certificate into the *tls/ldap* directory.

7

Perform 5.7 “To configure and enable a PKI server” (p. 59) to enable the configuration of TLS in the system.

8

Install the CLM. Execute the following commands as root user to install the components specified in the hosts file:

:

```
cd bin ↲

./install.sh ↲
```

Enter the password for the CLM when prompted.

9

If the auto_start parameter was set to false in Step 4, execute the following commands to start the system:

```
systemctl start nspos-nspd ↲

nspdctl --host <IP_address> start ↲
```

where *IP_address* is the IP address of the desired CLM server

End of steps

2.3 To upgrade a standalone CLM server

2.3.1 Purpose

Use this procedure to upgrade a standalone CLM server.

2.3.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 1, “Pre-installation”.

Note: Before performing an upgrade, all processes must be stopped on both the primary and standby servers and a database backup should be performed.

2.3.3 Steps

CAUTION

Deployment failure

The RHEL OS requires specific versions of some RHEL packages. If the required package versions are not installed, the CLM upgrade fails.

See 1.3.6 "Special RHEL OS package requirements" (p. 15) for the required package versions.

CAUTION

Deployment failure

Upgrades should not be performed on an CLM server that has never been operational.

Confirm that the CLM server to be upgraded has been started successfully before performing this procedure.

1

Stop all processes. Execute:

```
nspdctl --host <IP_address> stop ↵
```

```
systemctl stop nspos-nspd ↵
```

where *IP_address* is the IP address of the desired CLM server

2

Ensure that the supported version of RHEL 7 is running. As root user, execute the following command on the CLM server:

```
cat /etc/redhat-release ↵
```

Note: Any server running an unsupported version of RHEL 7 must be upgraded to a supported version.

3

Download the NSP_CLM_installer_<*release_load*>.tar.gz from the Nokia Support portal (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM, where <*release_load*> is the numbered CLM software release, such as 20_3 for release 20.3.

Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

An NSD_NRC_*R_r* directory is created in the current directory, where *R_r* is the CLM release identifier in the form *MAJOR_minor*.

Note: In subsequent steps, the directory is called the NSP installer directory or *NSP_installer_directory*.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from the Nokia Support portal. You can compare the checksum value on the download page with, for example, the output of the RHEL md5sum or sha256sum command. See the appropriate RHEL man page for information.

Note: The install.sh utility communicates with remote target stations using an SSH session. To perform an operation on a remote station using the utility, you must do one of the following.

- Configure the required SSH keys on the stations.
- Use the --ask-pass argument when you run install.sh, in which case each remote station must have the same root user password; for example:
  `./install.sh --ask-pass --target remote_station`

4

Enter the following to navigate to the NSP installer directory:

`cd NSD_NRC_R_r` ↵

5

Create a hosts file in the directory where the CLM installer bundle was extracted and add the required entries based on the components that the server will host. See 2.1.2 “Hosts file” (p. 31) for more information.

6

Create a YAML or JSON configuration file in the directory where the CLM installer bundle was extracted and add only the configuration blocks that apply to your deployment.

Note: You must set the secure parameter to true.

See 2.1.3 “Configuration file” (p. 31) for more information.

7

If the TLS block of the configuration file was populated in Step 6, copy the TLS certificates into the installer directory.

8

If LDAP authentication settings were configured in Step 6, copy the LDAP server certificate into the *tls/ldap* directory.

9

Perform 5.7 “To configure and enable a PKI server” (p. 59) to enable the configuration of TLS in the system.

10

Install the CLM. As root user, execute the following commands:

cd bin ↵

./install.sh ↵

Enter the password for the CLM when prompted.

11

If the auto_start parameter was set to false in Step 6, execute the following commands to start the system:

systemctl start nspos-nspd ↵

nspdctl --host <IP_address> start ↵

where *IP_address* is the IP address of the desired CLM server

END OF STEPS

3 Redundant installation and upgrade

3.1 Introduction

3.1.1 Overview

CAUTION

Service Disruption

In a redundant system, a GUI client that uses a main server IP address to open a browser connection to the CLM system may need to use the IP address of the peer main server after a main server communication failure.

To ensure GUI client access to the CLM in a redundant system, it is highly recommended that you do the following:

- Configure DNS for GUI clients to map each main server IP address to the same DNS name
- Configure each GUI client to use the DNS name for browser connections to the CLM system
- Use a client browser that caches multiple IP addresses associated with one hostname

This chapter describes redundant CLM installation and upgrade processes, as well as related operations.

3.2 To install a redundant CLM system

3.2.1 Purpose

Use this procedure to install a CLM system with 1+1 redundancy, which requires the installation of both a master CLM instance, and a standby CLM instance.

3.2.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 1, “Pre-installation”.

3.2.3 Steps

CAUTION

Deployment failure

The RHEL OS requires specific versions of some RHEL packages. If the required package versions are not installed, the CLM installation fails.

See 1.3.6 “Special RHEL OS package requirements” (p. 15) for the required package versions.

1

Download the NSP_CLM_installer_<*release_load*>.tar.gz from the Nokia Support portal (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM, where <*release_load*> is the numbered CLM software release, such as 20_3 for release 20.3.

Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

An NSD_NRC_*R_r* directory is created in the current directory, where *R_r* is the CLM release identifier in the form *MAJOR_minor*.

Note: In subsequent steps, the directory is called the NSP installer directory or *NSP_installer_directory*.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from the Nokia Support portal. You can compare the checksum value on the download page with, for example, the output of the RHEL md5sum or sha256sum command. See the appropriate RHEL man page for information.

Note: The install.sh utility communicates with remote target stations using an SSH session. To perform an operation on a remote station using the utility, you must do one of the following.

- Configure the required SSH keys on the stations.
- Use the --ask-pass argument when you run install.sh, in which case each remote station must have the same root user password; for example:

  `./install.sh --ask-pass --target remote_station`

2

Enter the following to navigate to the NSP installer directory:

`cd NSD_NRC_R_r` ↵

3

Create a hosts file in the directory where the CLM installer bundle was extracted and add the required entries based on the components that the CLM server will host. See 2.1.2 “Hosts file” (p. 31) for more information.

4

Create a YAML or JSON configuration file in the directory where the CLM installer bundle was extracted and add only the configuration blocks that apply to your deployment.

Note: You must set the secure parameter to true.

See 2.1.3 “Configuration file” (p. 31) for more information.

5

Perform 5.7 “To configure and enable a PKI server” (p. 59) to enable the configuration of TLS in the system.

6

Install the CLM. As root user, execute the following commands:

```
cd bin ↵
```

```
./install.sh ↵
```

Enter the password for the CLM when prompted.

7

If the auto_start parameter was set to false in Step 4, enter the following sequence of commands on each CLM server:

```
systemctl start nspos-nspd ↵
```

```
nspdctl --host <IP_address> start ↵
```

where *IP_address* is the IP address of the desired CLM server

The CLM server starts.

8

Close the open console windows.

End of steps

3.3 To upgrade redundant CLM servers

3.3.1 Purpose

Use this procedure to upgrade a CLM server deployed with 1+1 redundancy.

3.3.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 1, “Pre-installation”.

Note: Before performing an upgrade, all processes should be stopped on both the primary and standby servers and a database backup should be performed.

3.3.3 Steps

CAUTION

Deployment failure

The RHEL OS requires specific versions of some RHEL packages. If the required package versions are not installed, the CLM upgrade fails.

See 1.3.6 “Special RHEL OS package requirements” (p. 15) for the required package versions.

CAUTION

Deployment failure

Upgrades should not be performed on an CLM server that has never been operational.

Confirm that the CLM server to be upgraded has been started successfully before performing this procedure.

1

Stop all processes. Execute the following command on both the primary and standby CLM servers:

```
nspdctl --host <IP_address> stop ↵
```

```
systemctl stop nspos-nspd ↵
```

where *IP_address* is the IP address of the desired CLM server

2

Ensure that the supported version of RHEL 7 is running. As root user, execute the following command on both the primary and standby CLM servers:

```
cat /etc/redhat-release ↵
```

Note: Any server running an unsupported version of RHEL 7 must be upgraded to a supported version.

3

Download the NSP_CLM_installer_<*release_load*>.tar.gz from the Nokia Support portal (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM, where <*release_load*> is the numbered CLM software release, such as 20_3 for release 20.3.

Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

An NSD_NRC_*R_r* directory is created in the current directory, where *R_r* is the CLM release identifier in the form *MAJOR_minor*.

Note: In subsequent steps, the directory is called the NSP installer directory or *NSP_installer_directory*.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from the Nokia Support portal. You can compare the checksum value on the download page with, for example, the output of the RHEL md5sum or sha256sum command. See the appropriate RHEL man page for information.

Note: The install.sh utility communicates with remote target stations using an SSH session. To perform an operation on a remote station using the utility, you must do one of the following.

- Configure the required SSH keys on the stations.
- Use the --ask-pass argument when you run install.sh, in which case each remote station must have the same root user password; for example:
  ./install.sh --ask-pass --target *remote_station*

4

Enter the following to navigate to the NSP installer directory:

cd NSD_NRC_R_r ↵

5

Create a hosts file in the directory where the CLM installer bundle was extracted and add the required entries based on the components that the CLM server will host. See 2.1.2 “Hosts file” (p. 31) for more information.

6

Create a YAML or JSON configuration file in the directory where the CLM installer bundle was extracted and add only the configuration blocks that apply to your deployment. See 2.1.3 “Configuration file” (p. 31) for more information.

7

Perform 5.7 “To configure and enable a PKI server” (p. 59) to enable the configuration of TLS in the system.

8

If LDAP authentication settings were configured in Step 6, copy the LDAP server certificate into the *tls/ldap* directory.

9

Install the CLM servers. Execute the following commands:

cd bin ↵

./install.sh ↵

The CLM servers are automatically deployed on both servers.

10

If the auto_start parameter was set to false in Step 6, enter the following sequence of commands on each CLM server:

systemctl start nspos-nspd ↵

nspdctl --host <IP_address> start ↵

where *IP_address* is the IP address of the desired CLM server

The CLM servers start.

11

Close the open console windows.

END OF STEPS

3.4 To convert a standalone CLM system to a redundant CLM system

3.4.1 Purpose

Use this procedure to convert a previously-installed standalone CLM system to a redundant CLM system.

Note: Upon converting to a redundant CLM system, TLS communication configurations must be updated so that the IP addresses of both the active and standby CLM servers are included in the SAN entries.

3.4.2 Steps

1

Open the existing hosts file, that is located in the directory where the CLM installer bundle was extracted, with a plain-text editor such as vi.

2

Modify the entries for each component that the CLM servers will host so as to use their 1+1 redundancy versions. See 2.1.2 “Hosts file” (p. 31) for more information.

3

In the config.yml file, configure the *auto_start* parameter with a value of *false*.

4

Shutdown all the active processes on the active, standalone CLM system. Execute:

```
nspdctl --host <IP_address> stop ↵
```

```
systemctl stop nspos-nspd ↵
```

where *IP_address* is the IP address of the desired CLM server

5

Install the CLM. Execute the following commands on one of the servers:

```
cd bin ↵
```

```
./install.sh ↵
```

6

On what was previously the active, standalone CLM system, execute:

```
systemctl start nspos-nspd ↵
```

```
nspdctl --host <IP_address> start ↵
```

where *IP_address* is the IP address of the desired CLM server

7

On the standby CLM system, execute:

```
systemctl start nspos-nspd ↵
```

```
nspdctl --host <IP_address> start ↵
```

where *IP_address* is the IP address of the desired CLM server

END OF STEPS

4 Post-installation activities

4.1 Introduction

4.1.1 Overview

This chapter contains procedures that may need to be performed after installing or upgrading a CLM server.

4.2 To uninstall an CLM system

4.2.1 Purpose

Use this procedure to uninstall either a standalone CLM system, or a redundant CLM system.

Note: The uninstall.sh utility communicates with remote target stations using an SSH session. To perform an operation on a remote station using the utility, you must do one of the following.

- Configure the required SSH keys on the stations.
- Use the --ask-pass argument when you run uninstall.sh, in which case each remote station must have the same root user password; for example:
  `./uninstall.sh --ask-pass --target remote_station`

4.2.2 Steps

1

Perform one of the following:

a. Modify the hosts file in the installer directory so as to contain the IP addresses of the systems from which the CLM software will be uninstalled.

b. Create a new hosts file, as described in 2.2 “To install a standalone CLM system” (p. 37), that contains the IP addresses of the systems from which the CLM software will be uninstalled.

2

Execute the following commands:

`cd bin/` ↵

`./uninstall.sh` ↵

The CLM software is removed from all hosts declared in the hosts file.

END OF STEPS

4.3 To enable CLM notification forwarding to NSP Fault Management application

4.3.1 Purpose

You can configure the CLM to forward notifications to the NSP Fault Management application. If forwarding is enabled, the CLM notifications are raised and cleared as Fault Management alarms. The synchronization of information between the CLM and Fault Management occurs at 30-second intervals. For example, if a CLM notification clears, the associated Fault Management alarm clears within 30 seconds.

4.3.2 Before you begin

- The CLM must be in the same subnet as the NSP system that hosts the Fault Management application.
- The same PKI server used to install NSP must be used for the CLM installation.
- The CLM and NSP must be at Release 19.6 or later

4.3.3 Steps

1

Stop the CLM by executing the following command:

```
nspdctl --host <IP_address> stop ↵
```

where *IP_address* is the IP address of the desired CLM server

2

Create a configuration file in: /opt/nsp/os/app1-tomcat/conf/applications/license-manager/application.conf that contains the following:

```
app {
    alarmForwarder {
        fm {
            enabled: true
            locations: "IP_address_1:port;IP_address_2:port;[f30::
aede:48ff::44]:2281"
         }
    }
}
```

where

IP_address_1 and *IP_address_2* are the NSP server addresses; for an HA deployment, specify only the virtual IP address of the cluster

port is 2181, if the NSP nspOS security is disabled, and 2281 if enabled

Note: An IPv6 address must be enclosed in square bracket.

Note: The configuration file must be named application.conf.

3

Restart the CLM by executing the following command:

`nspdctl --host <IP_address> start` ↵

where *IP_address* is the IP address of the desired CLM server

END OF STEPS

5 CLM user account and security

5.1 Introduction

5.1.1 Overview

This chapter describes various tasks related to user accounts, authentication, and security-related tasks that may need to be performed during or after CLM deployment.

5.2 CLM user accounts

5.2.1 Default and RHEL user accounts

The default CLM user account is “admin”'.

The CLM also requires a RHEL user account called “nsp” in the nsp user group. The group and account is created at installation. Only the nsp user can start or stop a CLM server. Server uninstallation does not remove the nsp user account, user group, or home directory.

The nsp home directory is */opt/nsp/*. The initial nsp password is randomly generated, and must be changed by the root user. Root user privileges are required only for component installation or upgrade, and for low-level support functions.

5.3 CLM authentication

5.3.1 Description

For increased security, local user authentication is not supported. The CLM requires an external authentication source that is specified during system installation. For example, RADIUS, LDAP, or TACACS+.

See 2.1.4 “SSO configuration file parameters” (p. 33) for configuration information.

5.4 CLM login security

5.4.1 User login throttling

User login throttling limits failed login attempts based on a user and client source IP address combination in order to suppress password guessing and other abuse scenarios. Login throttling is enabled by default. A login failure rate can be configured, as well as a lockout period, if login attempts exceed the defined failure rate. The throttling parameters are configured at installation in the config.yml file.

After a failed login attempt, subsequent login attempts by the same user from the same source IP address during the login threshold period are blocked for the duration of the specified lockout period.

The login threshold period is defined by two parameters: The rate_seconds parameter defines a time interval, in seconds, and the rate_threshold parameter defines the number of allowed login attempts during the time interval.

The lockout_period parameter defines the interval, in seconds, during which further login attempts by the user from the same source address are blocked, if the login threshold is exceeded.

See 2.1.4 “SSO configuration file parameters” (p. 33) for information about the login_throttling parameters in the CLM configuration file.

5.4.2 User login failures

During CLM deployment, you can specify whether, and for how long, to lock out users that exceed a specified number of consecutive login failures.

See 2.1.4 “SSO configuration file parameters” (p. 33) for information about the login_failure parameters in the CLM configuration file.

5.5 To suppress security warnings in CLM browser sessions

5.5.1 Description

The following steps describe how to prevent the repeated display of security warnings in a browser that connects to the CLM using a private-CA-signed or self-signed TLS certificate.

Note: You do not need to perform the procedure if the certificate is signed by a public root CA, which is trusted by default.

5.5.2 Steps

1

Perform one of the following.

a. If you deployed TLS using a PKI server, transfer the ca.pem certificate file from the PKI server to each client station on which you want to suppress the browser warnings.

b. If you deployed TLS using the manual method, transfer your certificate file to each client station on which you want to suppress the browser warnings.

2

Perform one of the following.

a. Import the certificate to the certificate store of a client station OS.

Perform the appropriate procedure in the OS documentation to import the certificate; specify the certificate file as the certificate source.

Note: Such a procedure varies by OS type and version.

b. Import the certificate to the certificate store of a client browser.

Perform the appropriate procedure in the browser documentation to import the certificate; specify the certificate file as the certificate source.

Note: Such a procedure varies by browser type and version.

3

Open a browser session and verify that CLM opens without the display of security warnings.

END OF STEPS

5.6 CLM TLS configuration and management

5.6.1 Automated TLS deployment using a PKI server

To reduce the complexity of configuring TLS in a new CLM system, or adding components to an existing system, you can use a utility called a Public Key Infrastructure (PKI) server. Based on user input, a PKI server creates, signs, and distributes certificates to each entity that is configured to use the PKI server.

Note: A system upgrade preserves the TLS keystore and truststore files, which are used if no PKI server is specified during the upgrade.

Benefits of automated TLS deployment

In addition to simplifying the implementation of TLS, using a PKI server has the following benefits:

- No system downtime when adding components or during operations such as system conversion to redundancy
- No complex CLI operations or manual file transfers
- No operator requirement for knowledge of interface IP address or hostname assignments
- Compatible with current and future product releases
- Can generate a certificate, use an existing certificate, or use a new certificate that you provide

See 5.7 “To configure and enable a PKI server” (p. 59) for information about using a PKI server to deploy TLS.

Functional description

The PKI server is a standalone utility that implements TLS certificate signing requests (CSRs) from requesting entities in a CLM system. A PKI server is available on a station to which you extract a CLM software bundle.

Note: Only one PKI server instance is required for automated TLS deployment; the instance serves an entire CLM system.

Note: Nokia recommends that you run the utility from the installation location on a CLM server; optionally, however, you can run a copy of the utility on any station that is reachable by each requestor.

Note: The CLM messaging subsystems require a separate TLS certificate that is used internally. The certificate is generated and distributed automatically during an installation or during an upgrade and requires that the PKI server is running during the deployment. The

separate internal certificate is required regardless of the TLS configuration method you choose from system-wide CLM communication.

Initially, a PKI server attempts to import an existing TLS certificate; if no certificate is available, the server prompts the operator for certificate parameters and creates a local private root CA service. Subsequently, the PKI server polls for CSRs.

Upon receiving a CSR, for example, from a CLM server, the PKI server directs the private root CA to sign the requestor certificate, and then returns the signed certificate to the requestor. The requestor uses the signed certificate to create the required keystore and truststore files, and then enables TLS on the required local interfaces.

For a PKI server to implement TLS on a CLM component, the component configuration must include the PKI server information.

If a PKI server is specified:

- but no keystore and truststore files are specified, the PKI server generates a TLS certificate using the specified alias, which is mandatory
- but no keystore and truststore passwords are specified, the default password, which is available from technical support, is used

5.6.2 Managing TLS certificate expiry

A CLM server checks the expiry date of each TLS certificate in the local keystore during initialization, and every 24 hours after initialization. If a certificate is expired or approaching expiry, a CLM notification displays on the GUI. If the forwarding function is enabled to forward CLM notifications to the Fault Management application, the following alarms are viewable in the Fault Management application:

- a Warning alarm, if the certificate is to expire within 30 days of the current time
- a Critical alarm, if the certificate is to expire within 7 days of the current time
- a Critical alarm, if the certificate is expired

When a TLS certificate is expired, the CLM server continues to operate normally, but some functions that depend on secure communication may be inoperable.

Note: The Days Remaining value in an expiry alarm is based on the number of complete 24-hour periods until the certificate expiry time. If fewer than 24 hours remain until the expiry, the Days Remaining value is zero, but the CLM does not raise a notification about the expired certificate until the actual expiry time.

Note: If a keystore contains hierarchical certificates, the CLM checks the expiry date of each certificate in the hierarchy, starting with the lowest, and uses the earliest expiry date found as the reference point for raising a notification.

TLS certificate renewal

The TLS certificate renewal process is the same as the initial configuration process.

5.7 To configure and enable a PKI server

5.7.1 Purpose

The following procedure describes:

- how to configure the parameters for TLS certificate generation on a PKI server
- how to import an existing TLS certificate to the PKI server for distribution to requestors

After you perform the procedure, the PKI server:

- creates a local private root CA service
- generates a TLS certificate and uses the CA service to sign it, or imports a certificate
- polls for certificate requests
- distributes the certificate to each requestor

Note: You require root user privileges on a station.

Note: If you are configuring the CLM to forward notifications to the NSP Fault Management application, the same PKI server used to install NSP must be used for the CLM installation.

5.7.2 Steps

1

A PKI server is installed by default on a CLM server station. You can run the utility from the default installation location, or can copy the utility to another station that is reachable by all requestors. The PKI server file path is:

NSP_installer_directory/tools/pki

where *NSP_installer_directory* is the directory where the CLM software package was extracted

If you want to run the utility from another location, copy the pki-server file to the location.

2

Log in as the root user on the station on which you want to run the PKI server.

3

Open a console window.

4

Navigate to the directory that contains the pki-server file. The default installation location is:

NSP_installer_directory/tools/pki

where *NSP_installer_directory* is the directory where the CLM software package was extracted

5

If you have a set of signed certificate files that you want the PKI server to import and distribute to requestors, copy the files to the directory that contains the pki-server file. The files must be named:

- ca.key — private RSA key of the CA
- ca.pem — X.509 public key certificate signed using ca.key

Note: The files must be located in the same directory as the pki-server file, and the user that invokes the PKI server requires read access to the files.

6

Perform one of the following.

a. Enter the following to use the default PKI server port:

```
# ./pki-server ↵
```

b. Enter the following to specify a port other than the default:

```
# ./pki-server -port port ↵
```

where *port* is the port to use for receiving and responding to requests

Note: If you specify a port other than the default, you must specify the non-default port number when you configure each requestor to use the PKI server.

7

If you are importing a certificate, as described in Step 5, or have previously configured the root CA parameters for the PKI server, go to Step 21.

8

If this is the first time that the PKI server is run on the station, the following message and prompt are displayed:

```
********************************************************************************
No External Root CA detected on the filesystem.
********************************************************************************
Create new External Root CA Identity [y/n]?
```

9

Enter y ↵. The following prompt is displayed:

```
Organization Name (eg, company) []:
```

10

Enter your company name.

The following prompt is displayed:

```
Country Name (2 letter code) []:
```

11

Enter the two-letter ISO alpha-2 code for your country.

The following prompt is displayed:

```
State or Province Name (full name) []:
```

12

Enter your state or province name.

The following prompt is displayed:

```
Validity (days) [3650]:
```

13

Enter the length of time, in days, for which the TLS certificate is valid, or press ↵ to accept the default.

The following messages are displayed as the PKI server creates a local TLS root CA and begins to poll for TLS certificate requests:

```
date time Root CA generated successfully.
```

14

If this is the first time that the PKI server is run on the station, the following message and prompt are displayed:

```
**********************************************************************
No Internal Root CA detected on the filesystem.
**********************************************************************
Creating new Internal Root CA Identity.
```

15

The following prompt is displayed:

```
Organization Name (eg, company) []:
```

16

Enter your company name.

The following prompt is displayed:

```
Country Name (2 letter code) []:
```

17

Enter the two-letter ISO alpha-2 code for your country.

The following prompt is displayed:

```
State or Province Name (full name) []:
```

18

Enter your state or province name.

The following prompt is displayed:

```
Validity (days) [3650]:
```

19

Enter the length of time, in days, for which the TLS certificate is valid, or press ↵ to accept the default.

The following messages are displayed as the PKI server creates a local TLS root CA and begins to poll for TLS certificate requests:

```
date time Root CA generated successfully.
date time Using Root CA from disk, and serving requests on port port
```

20

Make a backup copy of the following private root CA files, which are in the current directory; store the files in a secure and remote location, such as a separate physical facility:

- ca.key
- ca.pem

21

When the PKI server receives a certificate request, the following is displayed:

```
date time Received request for CA cert from IP_address:port
```

If the PKI server successfully responds to the request, the following is displayed:

```
date time Successfully returned a signed certificate valid for IPs:
[IP_address_1...IP_address_n] and hostnames: [hostname_1...hostname_n]
```

22

The PKI server log is the pki-server.log file in the current directory. View the log to determine when the PKI server has distributed a certificate to each requestor.

23

When the PKI server has distributed a certificate to each requestor, enter CTRL+C to stop the PKI server.

Note: The PKI server must continue to run until the installation of all products that use the PKI server is complete.

24

Close the console window.

END OF STEPS

5.8 To migrate to the PKI server

5.8.1 Purpose

Use this procedure to migrate to the PKI server if the deprecated ROOT CA method, which involves generating ca.jks and ca-cert.pem files, has been used previously.

Note: This procedure should only be used if deployment was configured using the deprecated ROOT CA method.

5.8.2 Steps

1

Copy over the ca.jks file, which is the ROOT CA keystore, and the ca-cert.pem file, which is the ROOT CA certificate.

2

Use the existing ca.jks file to create a new ca.key file. Execute the following commands:

Note: You must enclose a password that contains a special character in single quotation marks; for example:
`-srcstorepass 'MyStorepa$$word' -deststorepass 'MyStorepa$$word'`

***path*/keytool -importkeystore -srckeystore ca.jks -destkeystore keystore.p12 -srcstorepass *storePassword* -deststorepass *storePassword* -deststoretype PKCS12**

openssl pkcs12 -in keystore.p12 -passin pass:*keyPassword* -nocerts -nodes -out ca.key

where

path is the path to the keytool utility

storePassword is the password to access the contents of the keystore

keyPassword is the password that is used to access the private key stored within the keystore

3

Move the new ca.key file to the PKI server location. By default, this is the *NSP_installer_directory*/tools/pki directory, where *NSP_installer_directory* is the directory where the CLM software package was extracted.

4

Copy the existing ca-cert.pem file to the PKI server location.

5

Rename the ca-cert.pem file to ca.pem.

6

Start the PKI server. Execute:

```
./pki-server
```

Note: The PKI server now uses the existing certificates within the file system. If ca.key and ca.pem files are not added as directed, the PKI server creates new files.

END OF STEPS

5.9 To manually generate a keystore

5.9.1 Purpose

A TLS keystore provides identity verification and encryption on all northbound and internal interfaces. You can manually generate a keystore file for distribution and use in a CLM system.

You can use the Java keytool utility to generate a TLS keystore file that contains a self-signed security certificate. The keytool utility is included in each Java Development Kit, or JDK, and Java Runtime Environment, or JRE.

Note: The keytool utility that you use must be from the Java version that the CLM uses. After a CLM server installation, you can find the keytool utility in /opt/nsp/os/jre/bin on the server. If the CLM is not yet installed, ensure that you use the keytool utility from the Java version that the CLM uses.

Note: A CLM keystore must be in Java Key Store, or JKS, keystore format.

5.9.2 Steps

1

Enter the following:

Note: You must enclose a password that contains a special character in single quotation marks; for example:
`-keypass 'Mypa$$word' -storepass 'Mypa$$word'`

path*/keytool –genkeypair –keystore *filename* –keypass *keyPassword* -storepass *storePassword* –keyalg rsa –alias *aliasName* –dname "CN=*commonName*, OU=*organizationalUnit*, O=*organization*, L=*location*, ST=*state*, C=*country*" –validity 7300 -ext bc=ca:true -ext san=*sanString
↵

where

path is the path to the keytool utility

filename is the absolute path to the Java KeyStore file that will hold the public/private key pair that is generated

keyPassword is the password that is used to access the private key stored within the keystore

storePassword is the password to access the contents of the keystore

aliasName is the human-readable identifier for the key pair that is used to differentiate between different keys in a keystore

commonName is the name of the keystore owner

organizationalUnit is the name of the organizational unit to which the keystore owner belongs

organization is the name of the organization to which the keystore owner belongs

location is the name of the city in which the keystore owner resides

state is the name of the state or province in which the keystore owner resides

country is the name of the country in which the keystore owner resides

sanString is a list of all interfaces on the CLM server(s), prefixed with the “IP:” string. This list must contain the loopback (127.0.0.1) interface. For example, a redundant CLM deployment with two servers having the IP addresses 10.0.0.1 and 10.0.0.2 would use: -ext san=IP:127.0.0.1,IP:10.0.0.1,IP:10.0.0.2. If hostnames were used during installation, they must be included, prefixed with the “DNS:” string. For example, -ext san=IP:127.0.0.1,DNS:*hostname*.nokia.com.

2

Use the custom_keystore_path parameter, under the TLS section, to point to the generated keystore file. You should also set the other TLS values to match the parameters specified in the preceding command.

END OF STEPS

5.10 Data privacy

5.10.1 Securing private data in the system

The following table indicates how private data is handled within the CLM. The servers in a CLM deployment reside within the secure domain of the customer network.

Table 5-1 CLM data privacy

Category	Description
Network element data	
Type of data	• Username and password • IP address
Purpose	• NE authentication • NE IP address for NE discovery/access
Storage	• Local database • Logs

Table 5-1 CLM data privacy (continued)

Category	Description
Retention	Data is retained in the database until an authorized user deletes it. Log retention can vary based on the log file size and number of log backups.
Processing	NE data is processed for the stated purpose.
Access	Authorized users
Safeguards	• NEs are configured by authorized users • Database access is restricted to authorized users • Secure transit option is available • Passwords for NE users are encrypted before being stored • Log file access is restricted to authorized users

5.11 To configure the CLM security statement

5.11.1 Purpose

Use this procedure to configure the security statement that is displayed on the CLM login page.

5.11.2 Steps

Install the CLM and start the nspOS

1

Perform one of the following:

- Install your standalone CLM system, as described in 2.2 “To install a standalone CLM system” (p. 37).
- Install your redundant CLM system, as described in 3.2 “To install a redundant CLM system” (p. 43).

2

Start the nspOS.

Configure the CLM security message

3

Sign in as an administrator user and launch the CLM application.

4

From the Launchpad, click User → NSP Settings.

5

Click System Settings.

6

Type a security statement in the text field, and then select the check box to enable the security statement.

Note: The security statement will not be displayed the first time that the CLM login page is accessed.

END OF STEPS

5.12 To update the supported CLM TLS versions and ciphers

5.12.1 Purpose

CAUTION

Service Disruption

This procedure involves a complete shutdown and restart of the CLM system.

It is strongly recommended that you perform this procedure only during a scheduled maintenance period.

CAUTION

Potential Service Outage

- A client that uses Java 7, update 94 or earlier, cannot connect to the CLM unless you enable TSLv1 on the CLM.
- A client that uses Java 7, update 95 or later, can connect to the CLM only if you enable TLSv1.1 or TLSv1.2, as required, on the client station.

  To enable TLSv1.1 or TLSv1.2 for a client, you must add the protocol to the list of supported protocols defined by the jdk.tls.client.protocols Java system property on the client station, for example:

  jdk.tls.client.protocols=TLSv1.2

Outdated TLS versions or ciphers may present a security risk. Perform this procedure to update the lists of supported ciphers and TLS versions (for example, to also enable TLS 1.0) in an CLM system.

Note: By default, the CLM only supports TLS 1.2; TLS 2.0 is not supported.

5.12.2 Steps

1

Log in to the standalone or primary CLM server station as the nsp user.

2

Enter the following:

```
bash$ cd /opt/nsp/scripts/security ↵
```

Prepare new cipher files

3

Enter the following to create the default cipher list file:

```
bash$ ./ciphers_and_tls_update.bash create -cdc default-ciphers-file ↵
```

4

Enter the following to copy the default ciphers file to a new file:

```
bash$ cp default-ciphers-file new_ciphers_file ↵
```

where *new_ciphers_file* is the name to assign to the new ciphers file

5

Open *new_ciphers_file* using a plain-text editor such as vi.

6

Remove the ciphers that are not to be supported.

7

Save and close the file.

Prepare new TLS versions files

8

Enter the following to create the default TLS list file:

```
bash$ ./ciphers_and_tls_update.bash create -cdt default-tls-file ↵
```

9

Enter the following to copy the default TLS versions file to a new file:

```
bash$ cp default-tls-file new_tls_file ↵
```

where *new_tls_file* is the name to assign to the new TLS versions file

10

Open *new_tls_file* using a plain-text editor such as vi.

11

Remove the TLS versions that are not to be supported.

Note: TLSv1.2 is mandatory and must not be removed.

12

Save and close the file.

Distribute files to system components

13

If the CLM system is redundant, distribute the required files to the standby CLM server station.

1. Log in to the standby CLM server station as the root user.
2. Enter the following:

   ```
   # cd /opt/nsp/scripts/security ↵
   ```

3. Copy the required file(s) from the primary CLM server station to the current directory. The file(s) that appear depend on whether you prepared a new ciphers file and/or new TLS versions file(s).
   - /opt/nsp/scripts/security/*new_ciphers_file*
   - /opt/nsp/scripts/security/*new_tls_file*

Stop CLM system

14

If the CLM system is redundant, stop the standby CLM server.

15

Stop the standalone or primary CLM server.

Apply new cipher and/or TLS lists

16

Perform the following steps on each CLM server station to apply the new TLS configuration.

1. Log in as the nsp user.
2. Open a console window.
3. Enter the following:

   ```
   bash$ cd /opt/nsp/scripts/security ↵
   ```

4. Enter the following:

Note: The -fo parameter is optional, and sets the cipher priority according to the order in the specified file. If the parameter is not included, the cipher priority is set to the default order.

```
bash$ ./ciphers_and_tls_update.bash apply -c new_ciphers_file -t
new_tls_file -fo ↵
```

where

new_ciphers_file is the updated ciphers file - if you are not updating the ciphers you can skip the -c argument

new_tls_file is the updated TLS versions file - if you are not updating the TLS list you can skip the -t argument

The script applies the new configuration, and backs up the previous configuration in the following file:

ciphers_and_tls_backup.*timestamp*.tar.gz

17

Close the open console windows.

Start CLM system

18

Start the standalone or primary main server.

19

If the CLM system is redundant, start the standby CLM server.

20

Close the open console windows.

END OF STEPS

6 Backup and restore

6.1 Introduction

6.1.1 Overview

This chapter describes the procedures that must be performed in order to preserve crucial system data in the case of a catastrophic failure.

6.2 To manually backup the PostgreSQL database

6.2.1 Purpose

Use this procedure to manually backup the contents of the PostgreSQL database.

Note: Scheduled database backups occur daily. The backup files are stored in the */opt/nsp/backup/scheduled* directory for up to seven days. A maximum of four backups taken on Wednesdays can be saved for up to one month. The */opt/nsp/scripts/db/nsp-backup.conf* file can be modified in order to customize this automated backup schedule.

6.2.2 Steps

1

Log in to the primary CLM server as the nsp user.

2

Enter the following:

```
nspdctl --host <IP_address> backup -d nspos_migration -f ↵
```

where *IP_address* is the IP address of the desired CLM server

3

Verify that the backup has completed successfully. Execute:

```
nspdctl --host <IP_address> backup status ↵
```

where *IP_address* is the IP address of the desired CLM server

4

As nsp user, transfer the backup files from */opt/nsp/backup/nspos_migration/* to the */tmp/nspos_migration* directory within the CLM server.

Note: If the CLM system was deployed in a redundant configuration, the backup files must be transferred to the active CLM server.

END OF STEPS

6.3 To restore the PostgreSQL database

6.3.1 Purpose

Use this procedure to restore the PostgreSQL database from backups following a catastrophic system failure.

Note: All commands presented in this procedure must be executed as nsp user.

6.3.2 Before you begin

Before restoring the databases, backups must be created using the **`nspdctl --host <IP_address> backup`** CLI command, or using the POST /backup/trigger/ REST API method. See the NSP Developer portal for more information.

6.3.3 Steps

1

Backup the PostgreSQL database as described in 6.2 “To manually backup the PostgreSQL database” (p. 71).

2

Copy all database backup files generated in Step 1 to the system where the CLM installer bundle was extracted.

3

Stop the CLM services. As nsp user, execute the following command on a standalone CLM server, or on both servers if the CLM system was deployed in a redundant configuration:

`nspdctl --host <IP_address> stop` ↵

where *IP_address* is the IP address of the desired CLM server

4

As root user, navigate to the *tools/database* directory on the system where the CLM installer bundle was extracted and execute the following command:

5

Enter the following:

`db-restore.sh` ↵

6

When prompted, specify the path to the database backup file to be restored.

7

Repeat Step 4 and Step 6 for each database backup file to be restored.

8

Restart the nspd agent. As nsp user, execute the following command on a standalone CLM server, or on both servers if the CLM system was deployed in a redundant configuration:

```
nspdctl --host <IP_address> start ↵
```

where *IP_address* is the IP address of the desired CLM server

END OF STEPS

A Obtaining CLM software and documentation

A.1 Software

A.1.1 Software download

As a registered customer, you can download CLM software from the Nokia Support portal. If you are a new customer and require access, contact your sales or support representative for registration information.

The CLM software on the Electronic Delivery→Downloads portal, also called ALED, is organized by release. You navigate through the hierarchy to select and download the packages you are licensed to use according to your purchase agreement.

After you select items for download and click Next, you must choose a download method. Click Help for information about the available download methods.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from Nokia Support portal. You can compare the checksum value on the download page with, for example, the output of the RHEL md5sum or sha256sum command. See the appropriate RHEL man page for information.

A.2 Documentation

A.2.1 Documentation architecture

CLM documentation consists of:

- application help
- product-level documents

Application help

The CLM has application help to guide operators in the use of the interface. Application help is delivered in the NSP Help Center, which can be opened from a ? button on the NSP Launchpad or the CLM application banner bar.

Product-level documents

CLM documentation consists of the following:

- *CLM User Guide*
- *CLM Installation and Upgrade Guide*
- *CLM Release Notice*

A.2.2 NSP Help Center

Starting with CLM Release 19.6, CLM user documentation is delivered in an on-product application called the NSP Help Center. During CLM installation, the NSP Help Center loads the information content associated with each application in your CLM deployment.

The Help Center can be opened from a ? button available in the CLM application banner bar, as well as the NSP Launchpad. You can browse the documentation from menus on the Help Center home page, or use the searching and filtering capabilities to isolate information quickly.

Searching

The Help Center application is centered on its robust search capabilities. Searches conducted from the home page search across documentation for all installed CLM and NSP components. As shown in the tooltip on the search bar, the boolean operators AND/OR/NOT are supported, as are the wildcard characters * (any string) and ? (any character). Exact-phrase search strings enclosed in quotation marks are also supported.

Note: Common, non-technical terms such as “the,” “and,” “on,” and others are ignored in all searches, including exact-string searches.

Search history is tracked as follows:

- The Recent Searches list on the home page is per-user, and the Popular Searches list shows the trend across all users of the system.
- When a search result link is clicked on the search results page, it is captured in the Recent Searches list and considered for forming the Popular Search list. Navigating to a page in any other way (for example, by browsing from the browse menu or following links within a browsed document) does not make the page eligible for capture in the Recent/Popular Searches list.

You can use the browser search to search within a page of content, or execute a new search from the search bar in the top right of your result page without having to return to the home page.

Filtering

You can filter your search results using filters in the left panel of the search results page.

You can filter by either or both of these facets:

- Application and NSP Guides
  Select one or more documentation sets to narrow your search results to those areas.
- Content Type
  Select one or more content types to narrow your search results to hits that match the content type. For example, if your search for a term and you only want to see procedural information, select “Procedure” as the content type.

Browsing

From the home page, you can browse documentation under APPLICATION GUIDES or NSP GUIDES. Once you have selected a guide or search result, you can browse within a guide using the table of contents tree in the left navigation panel.

Use the breadcrumbs in the search path to return to search results or the home page. Use the browser back button to return to any previously visited page.

A.2.3 Documentation delivery online

The documentation delivered on product in the NSP Help Center is also available online in PDF on the Nokia Documentation Center. If you are a new user and require access to the service, contact your support representative.

From the CLM product documentation page in the Documentation Center, you can:

- filter by release, category, content type, and format
- sort the results by title, document number, most accessed, or issue date
- search for documents
- search inside documents
- create a downloadable collection of your filtered documents

User documentation is filed under the “Manuals and Guides” content type; Release Notices and Release Descriptions are filed under “Release Information”.

Documentation alerts

To receive an e-mail when new or reissued CLM customer documents are available, subscribe to the notification service on the Documentation Alerts Subscription page.