NOKIA

Centralized License Manager

Release 18.12

Installation and Upgrade Guide

3HE-14321-AAAC-TQZZA

Issue 1

December 2018

Legal notice

Contents

List of tables

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:

- Customer Documentation Welcome Page
- Technical support

How to comment

Documentation feedback

- Documentation Feedback

1 Safety information

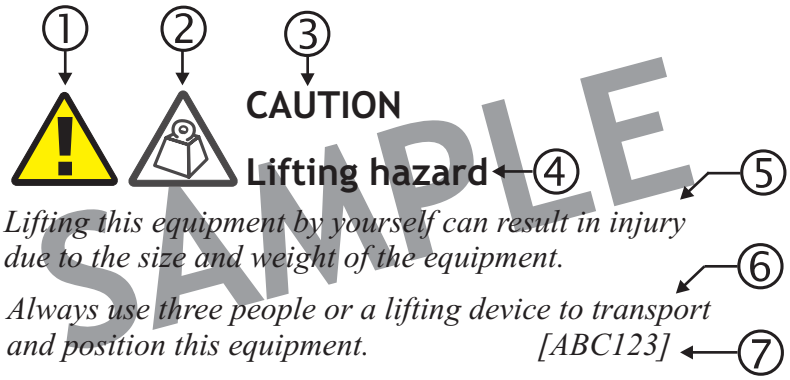
1.1 Structure of safety statements

1.1.1 Overview

This topic describes the components of safety statements that appear in this document.

1.1.2 General structure

Safety statements include the following structural elements:



Item	Structure element	Purpose
1	Safety alert symbol	Indicates the potential for personal injury (optional)
2	Safety symbol	Indicates hazard type (optional)
3	Signal word	Indicates the severity of the hazard
4	Hazard type	Describes the source of the risk of damage or injury
5	Safety message	Consequences if protective measures fail
6	Avoidance message	Protective measures to take to avoid the hazard
7	Identifier	The reference ID of the safety statement (optional)

1.1.3 Signal words

The signal words identify the hazard severity levels as follows:

Signal word	Meaning
DANGER	Indicates an extremely hazardous situation which, if not avoided, will result in death or serious injury.
WARNING	Indicates a hazardous situation which, if not avoided, could result in death or serious injury.
CAUTION	Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.
NOTICE	Indicates a hazardous situation not related to personal injury.

2 Pre-installation

2.1 Introduction

2.1.1 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.

2.2 Operating system specifications

2.2.1 Red Hat Enterprise Linux (RHEL) description and recommendations

The CLM is supported on Red Hat Enterprise Linux Server Edition 7.3, 7.4, and 7.5 (x86-64). 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.

2.2.2 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.

2.3 RHEL OS installation requirements

2.3.1 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.

2.3.2 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.

2.3.3 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 2.3.4 “RHEL OS packages to install” (p. 11).
3. Remove specific OS packages, as described in 2.3.5 “RHEL OS packages to remove” (p. 15).

4. Upgrade or install specific OS packages, as required, depending on the RHEL version; see 2.3.6 “Special RHEL OS package requirements” (p. 17).
5. Optionally, install one or more packages listed in 2.3.7 “Optional RHEL OS packages” (p. 17).

2.3.4 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. 11).

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. 14).

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 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 rsync.x86_64 tcpdump.x86_64 unzip.x86_64 which
yum -y install xinetd.x86_64 xz-devel.x86_64 zip.x86_64
```

Table 2-1 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
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

Table 2-1 Required OS packages from default RHEL repository or ISO image (continued)

Package	Description
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
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

Table 2-1 Required OS packages from default RHEL repository or ISO image (continued)

Package	Description
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
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
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 2-2, “Required OS packages from RHEL optional package repository” (p. 15) 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 2-2 Required OS packages from RHEL optional package repository

Package name	Description
compat-libstdc++-33.i686	Compatibility standard C++ libraries
compat-libstdc++-33.x86_64	Compatibility standard C++ libraries

2.3.5 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.

The packages that you remove depend on the RHEL version, as described below.

RHEL 7.3 or 7.4

For RHEL 7.3 or 7.4, you must remove the following.

- the packages described in "All RHEL 7 versions" (p. 15)
- the additional packages listed in Table 2-3, "Additional OS packages to remove, RHEL 7.3 or 7.4" (p. 15); 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
```

Table 2-3 Additional 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

All RHEL 7 versions

For all RHEL 7 versions, you must remove the packages listed in Table 2-4, "RHEL OS packages to remove, all RHEL versions" (p. 16). 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 2-4 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

2.3.6 Special RHEL OS package requirements

The CLM requires specific versions of some packages, as described in "Specific package version requirements" (p. 16), and requires the installation of specific packages after an upgrade to RHEL 7.5, as described in "Upgrading to RHEL 7.5" (p. 17).

Specific package version requirements

The CLM requires the version of each RHEL 7 package quoted in Table 2-5, "Required RHEL OS package versions" (p. 16), 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.

Table 2-5 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

Upgrading to RHEL 7.5

If you upgrade from RHEL 7.3 or 7.4 to RHEL 7.5, you must install the packages listed in Table 2-6, "Additional OS packages required after upgrade to RHEL 7.5" (p. 17).

Table 2-6 Additional OS packages required after upgrade to RHEL 7.5

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

2.3.7 Optional RHEL OS packages

Table 2-7, "Optional RHEL OS packages" (p. 17) 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 telnet.x86_64 vsftpd.x86_64
```

Table 2-7 Optional RHEL OS packages

Package	Description
nfs-utils	NFS utilities and supporting clients and daemons for the kernel
telnet.x86_64	The client program for the Telnet remote login protocol
vsftpd.x86_64	Very Secure Ftp Daemon

2.4 Virtual machine requirements

2.4.1 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, or 7.5 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.

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.

2.5 VMware Virtualization

2.5.1 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 2-8 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
Advanced CPU	Hyperthreaded Core Sharing Mod	Set to None
Memory	Shares	Set to High
	Reservation	Slider set to the size of the memory allocated to the VM
	Limit	Check box checked for unlimited
Advanced Memory	NUMA Memory Affinity	No affinity
Disk	Shares	Set to High
	Limit — IOPs	Set to Unlimited

2.6 KVM virtualization

2.6.1 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, 2.3.0, 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.

2.6.2 Configuration

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

Table 2-9 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

2.7 OpenStack requirements

2.7.1 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.

2.7.2 Hypervisor

The only hypervisor supported within an OpenStack environment is KVM. For details about the KVM hypervisor supported versions, see 2.6 “KVM virtualization” (p. 19).

2.7.3 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.

2.7.4 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.

2.7.5 CPU pinning

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

2.7.6 Availability zones/affinity/placement

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

2.7.7 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.

2.7.8 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.

2.7.9 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.

2.8 Platform requirements

2.8.1 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 2-10 CLM hardware platform requirements

Hardware	Requirement
CPU cores	2 (minimum 2.4 GHz)
Memory	minimum 16 GB
Disk	1 SAS 10K RPM drive, 200 GB or more

2.9 Partitioning

2.9.1 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 2-11, “CLM servers partitioning scheme” (p. 22) lists the partitioning requirements for CLM components in both live and lab deployments.

Table 2-11 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

2.10 Securing the CLM

2.10.1 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.

Communications is secured using TLS. The CLM supports TLS versions TLSv1.2, TLSv1.1, and TLSv1.0.

2.11 Operating system security for CLM workstations

2.11.1 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.

2.11.2 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.

2.12 CLM firewalls

2.12.1 Overview

A firewall can be deployed to protect the CLM from different networks and applications.

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

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.

2.12.2 Firewall port requirements for CLM deployments

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

The CLM deployment types are:

- standalone
- redundant

Table 2-12 Listening ports for all communications with CLM

Applica-tion	Default port(s)	Type	Encryption	Description	CLM deployment
All	22	TCP	Dynamic Encryption	SSH/SCP/SFTP Used for remote access and secure file transfer	All

Table 2-12 Listening ports for all communications with CLM (continued)

Applica-tion	Default port(s)	Type	Encryption	Description	CLM deployment
CLM	8105	TCP	None	Java Tomcat Local port to the host	All
	8223	TCP	None	Java Tomcat	Redundant
	8224	TCP	Dynamic, SSL/TLS	Java Tomcat Local port to the host	All
	8225	TCP	Dynamic, SSL/TLS	Java Tomcat Local port to the host	All
	8543	TCP	Dynamic, SSL/TLS	Java Tomcat, secure HTTPS port for GUI and REST API	All
	10800	TCP	None	Java Tomcat	All
	11211	TCP	None	Ignite cache	All
	47100–47199	TCP	None	Ignite cache	All
	47500–47599	TCP	None	Ignite cache	All
	48100–48199	TCP	None	Ignite cache Local port to the host	All

Table 2-12 Listening ports for all communications with CLM (continued)

Application	Default port(s)	Type	Encryption	Description	CLM deployment
nspOS	80	TCP	None	HTTP port for nspOS common applications, redirect to 443	All
	443	TCP	Dynamic, SSL/TLS	Secure HTTPS port for nspOS common applications	All
	2181	TCP	None	Zookeeper	All
	2390	TCP	Dynamic, SSL/TLS	nspdctl	All
	2391	TCP	None	PKI server	Only where PKI server is installed and running
	6432	TCP	None	PostgreSQL database	All
	7889	TCP	None	Telemetry Local port to the host	All
	8195	TCP	None	tomcat shutdown port Local port to the host	All
	8196	TCP	None	app1-tomcat shutdown port Local port to the host	All
	8544	TCP	Dynamic, SSL/TLS	HTTPS port for app1-tomcat	All
	9000	TCP	None	gRPC server Local port to the host.	All
	9092	TCP	None	Kafka server	All
	47100–47199	TCP	SSL/TLS	CAS ignite cache	All
	47500–47599	TCP	SSL/TLS	CAS ignite cache	All
	48500–48599	TCP	SSL/TLS	session-manager ignite cache	All
	48600–48699	TCP	SSL/TLS	session-manager ignite cache	All

Table 2-13 Ports used in communication between the active and standby CLM in a redundant deployment

Protocol	From port	To port
TCP	>32768	22

Table 2-13 Ports used in communication between the active and standby CLM in a redundant deployment (continued)

Protocol	From port	To port
TCP	22	>32768
TCP	>32768	2390
TCP	2390	>32768
TCP	>32768	5001
TCP	5001	>32768
TCP	>32768	5002
TCP	5002	>32768
TCP	>32768	5007
TCP	5007	>32768
TCP	>32768	6007
TCP	6007	>32768
TCP	>32768	6017
TCP	6017	>32768
TCP	>32768	6018
TCP	6018	>32768
TCP	>32768	6432
TCP	6432	>32768

Table 2-14 Ports used in communication between CLM and client (GUI/REST) applications

Protocol	To port	To module	Purpose
TCP	80	CLM / nspOS	for Launchpad redirect
TCP	443	CLM / nspOS	for Launchpad
TCP	8543	CLM	for CLM GUI, REST API
TCP	8544	CLM / nspOS	nspOS web applications
TCP	9092	CLM / nspOS	External notifications (messaging)

3 Standalone installation and upgrade

3.1 Introduction

3.1.1 Overview

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

3.1.2 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.

Note: A sample hosts file can be found in the */<loadpath>/NSD_NRC_<R_r>/examples/hosts* directory. It is highly recommend that a modified copy of this file be used during installation.

Table 3-1 Hosts file components

Deployed component	Required hosts file entry
nspOS + CLM (standalone)	**[nspos]** ***IPaddress*** **[clm]** ***IPaddress*** Where *IPaddress* is the IP address of the server where the software will be installed.

Table 3-1 Hosts file components (continued)

Deployed component	Required hosts file entry
nspOS + CLM (1+1 redundancy)	**[nspos]** **<primary server address>** **dc=<location>** **<standby server address>** **dc=<location>** **[clm]** **<primary server address>** **dc=<location>** **<standby server address>** **dc=<location>** 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

3.1.3 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 3.1.4 “SSO configuration file parameters” (p. 29) for a list of parameters available in the SSO block of the configuration file.

Note: A sample configuration file can be found at*/<loadpath>/NSD_NRC_<R_r>/examples/config.yaml*. It is highly recommend that a modified copy of this file be used during installation.

Table 3-2 Configuration file parameters

Parameter	Definition
auto_start	Specifies whether or not the CLM starts once installation is complete
ean — External applications notifications parameters	
max_subscribers	The maximum number of subscribers who can receive external applications notifications
tls — Used to customize TLS security	
pki_server	The IP address or hostname of the PKI server
pki_server_port	The port used for connecting to the PKI server. Default is 2391.

Table 3-2 Configuration file parameters (continued)

Parameter	Definition
pki_org	The organization name to use in your TLS certificate
pki_cn	The common name to use in your TLS certificate
custom_keystore_path	Allows you to provide your own TLS keystore
custom_truststore_path	Allows you to provide your own TLS truststore
custom_keystore_password	The password to your TLS keystore
custom_truststore_password	The password to your TLS truststore
custom_key_alias	The alias of the key to use in your TLS keystore

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.

3.1.4 SSO configuration file parameters

The SSO block of the configuration file is used to define CLM user authentication sources and login features. If the SSO block of the configuration file has no enabled authentication sources, then a local authentication source (PostgreSQL) will be enabled.

session	
concurrent_limits_enabled	Specifies whether or not maximum session limits are enabled - true or false
max_sessions_per_user	Specifies the maximum number of concurrent sessions per user
max_sessions_for_admin	Specifies the maximum number of concurrent sessions for users of the admin group
local	
enabled	Specifies whether or not locally defined users from the nspOS database are used - true or false
ldap	
enabled	Specifies whether or not LDAP is used for authentication - true or false
type	Specifies the server type - AUTHENTICATION, AD, or ANONYMOUS

url	Specifies the server URL with IP/hostname and port
security	Specifies the type of security used for the server - SSL, STARTTLS, or NONE
timeout	Specifies the timeout period (in seconds) for receiving an authentication response
user_base_dn	Specifies the user base dn value
user_filter	Used to filter by user name
group_base_dn	Specifies the group base dn value
min_pool_size	Specifies the minimum pool size
max_pool_size	Specifies the maximum pool size
use_entry_resolver	Specifies whether or not an entry resolver will be used - true or false
bind	
dn	User with authority to bind to LDAP server
credential	Password for bind user **Note:** The password must be enclosed in double quotation marks.
radius	
enabled	Specifies whether or not RADIUS users will be used for authentication - true or false
address	Specifies the hostname/IP address of one or more servers
secret	Specifies the shared secret for servers **Note:** The shared secret value must be enclosed in double quotation marks.
protocol	Specifies the protocol to be used - PAP/CHAP
retries	Specifies the maximum number of retries when attempting to reach server
timeout	Specifies the timeout period (in seconds) when attempting to reach server
failover_on_exception	Specifies whether or not, if first server fails with an exception, a second server is tried - true or false

failover_on_rejection	Specifies whether or not, if first server fails with a rejection, a second server is tried - true or false
authentication_port	Specifies the RADIUS port
vendor_id	Specifies the vendor ID used for VSA search
role_VSA_id	Specifies the VSA ID used to identify group property
tacacs	
enabled	Specifies whether or not TACACS users will be used for authentication - true or false
address	Specifies the hostname/IP address of one or more servers
secret	Specifies the shared secret between servers **Note:** The shared secret value must be enclosed in double quotation marks.
protocol	Specifies the protocol to be used - PAP/CHAP
timeout	Specifies the timeout period (in seconds) when attempting to reach server. Default is 7.
failover_on_exception	Specifies whether or not, if first server fails with an exception, a second server is tried - true or false
failover_on_rejection	Specifies whether or not, if first server fails with a rejection, a second server is tried - true or false
authentication_port	Specifies the TACACS port. Default is 49.
default_group	The default group to be assigned if none are defined within the server
VSA_enabled	Specifies whether or not VSA search is enabled - true or false
role_VSA_id	Specifies the role used for VSA search
VSA_service_id	Specifies the service used for VSA search
throttling	
enabled	Specifies whether or not to enable login throttling - true or false
rate_threshold	Specifies the login failure rate, calculated as seconds/threshold. Lockout occurs when this rate is exceeded. Default is 3.

rate_seconds	Specifies the login failure rate, calculated as seconds/threshold. Exceeded if second login attempt comes within 3 seconds of a previous failed login attempt. Default is 9.
lockout_period	Specifies the number of seconds for which to prevent the same user/IP combo from attempting login after threshold is exceeded. Default is 5.

Note: Any certificates required for secure LDAP communications should be copied to <*nsp installer directory*>/ssl/ldap/. If a n 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.

Login throttling

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.

Following a failed login attempt, if another login attempt is made by that same user from the same source IP address within a defined threshold period, that login attempt will be blocked and a lockout period will be applied during which no further logins (from that user/IP combination) will be accepted.

The threshold period is defined by two parameters. The *rate_seconds* parameter defines an interval of time in seconds, while the *rate_threshold* parameter defines the number of logins within that period of time. If a user fails an authentication attempt and then tries to login again from the same source IP address within the threshold period, that login attempt will be blocked. If a user fails a login, but their next login attempt exceeds the threshold period, then that login attempt will be processed. The *lockout_period* parameter defines the number of seconds that the user/IP are blocked from further login attempts if the login threshold has been exceeded.

3.2 To install a standalone CLM system

3.2.1 Purpose

Use this procedure to install a standalone CLM system.

3.2.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 2, “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 2.3.6 "Special RHEL OS package requirements" (p. 17) for the required package versions.

1

Download the NSP_CLM_18_12.tar.gz from OLCS (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM. Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

Note: When performing remote operations, SSH connections are used between the system where the CLM installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords, which requires the configuration of SSH keys, or the *--ask-pass* argument to be used when running the install.sh or uninstall.sh utilities, which requires that all systems share the same root user SSH password.

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*.

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 3.1.2 "Hosts file" (p. 27) for more information.

Note: A sample hosts file can be found in the */<loadpath>/NSD_NRC_<R_r>/examples/hosts* directory. It is highly recommend that a modified copy of this file be used during installation.

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. See 3.1.3 "Configuration file" (p. 28) for more information.

Note: A sample configuration file can be found at */<loadpath>/NSD_NRC_<R_r>/examples/config.yaml*. It is highly recommend that a modified copy of this file be used during installation.

5

Copy the appropriate license file(s) into the license directory where the CLM installer bundle was extracted.

6

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

7

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

8

Perform 6.4 “To configure and enable a PKI server” (p. 52) to enable the configuration of TLS in the system.

9

Install the CLM. Execute the following commands:

```
cd bin ↵

./install.sh ↵
```

10

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

3.3 To upgrade a standalone CLM server

3.3.1 Purpose

Use this procedure to upgrade a standalone CLM server.

3.3.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 2, “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.

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 2.3.6 “Special RHEL OS package requirements” (p. 17) 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_18_12.tar.gz from OLCS (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM. Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

Note: When performing remote operations, SSH connections are used between the system where the CLM installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords, which requires the configuration of SSH keys, or the *--ask-pass* argument to be used when running the install.sh or uninstall.sh utilities, which requires that all systems share the same root user SSH password.

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*.

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 3.1.2 “Hosts file” (p. 27) for more information.

Note: A sample hosts file can be found in the */<loadpath>/NSD_NRC_<R_r>/examples/hosts* directory. It is highly recommend that a modified copy of this file be used during installation.

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 3.1.3 “Configuration file” (p. 28) for more information.

Note: A sample configuration file can be found at */<loadpath>/NSD_NRC_<R_r>/examples/config.yaml*. It is highly recommend that a modified copy of this file be used during installation.

7

Copy the appropriate license file(s) into the license directory where the CLM installer bundle was extracted.

8

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

9

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

10

Perform 6.4 “To configure and enable a PKI server” (p. 52) to enable the configuration of TLS in the system.

11

Install the CLM. Execute the following commands:

```
cd bin ↵

./install.sh ↵
```

12

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

4 Redundant installation and upgrade

4.1 Introduction

4.1.1 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.

4.2 To install a redundant CLM system

4.2.1 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.

Note: The CLM server instances will not initialize without a redundant license, which must be obtained from Nokia personnel.

4.2.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 2, “Pre-installation”.

4.2.3 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 2.3.6 "Special RHEL OS package requirements" (p. 17) for the required package versions.

1

Download the NSP_CLM_18_12.tar.gz from OLCS (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM. Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

Note: When performing remote operations, SSH connections are used between the system where the CLM installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords. Otherwise, the *--ask-pass* argument must be used when running the install.sh or uninstall.sh utilities, which will require that all systems share the same root user SSH password.

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*.

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 3.1.2 "Hosts file" (p. 27) for more information.

Note: A sample hosts file can be found in the */<loadpath>/NSD_NRC_<R_r>/examples/hosts* directory. It is highly recommend that a modified copy of this file be used during installation.

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. See 3.1.3 "Configuration file" (p. 28) for more information.

Note: A sample configuration file can be found at */<loadpath>/NSD_NRC_<R_r>/examples/config.yaml*. It is highly recommend that a modified copy of this file be used during installation.

5

Copy the appropriate license file(s) into the license directory where the CLM installer bundle was extracted.

6

Perform 6.4 “To configure and enable a PKI server” (p. 52) to enable the configuration of TLS in the system.

7

Install the CLM. Execute the following commands:

```
cd bin ↲

./install.sh ↲
```

8

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.

9

Close the open console windows.

END OF STEPS

4.3 To upgrade redundant CLM servers

4.3.1 Purpose

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

Note: The CLM servers will not initialize without a redundant license, which must be obtained from Nokia personnel.

4.3.2 Before you begin

Before executing the CLM installer, ensure that your system meets the hardware and software requirements described see Chapter 2, “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.

4.3.3 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 2.3.6 "Special RHEL OS package requirements" (p. 17) 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_18_12.tar.gz from OLCS (delivered under the Centralized License Manager product hierarchy) to use the NSP installer utility for CLM. Extract it as the CLM installer bundle on any system running a supported version of RHEL 7.

Note: When performing remote operations, SSH connections are used between the system where the CLM installer bundle was extracted and the system(s) on which it will execute its tasks. Therefore, SSH connections must be possible between these systems without the use of passwords. Otherwise, the *--ask-pass* argument must be used when

running the install.sh or uninstall.sh utilities, which will require that all systems share the same root user SSH password.

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*.

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 3.1.2 “Hosts file” (p. 27) for more information.

Note: A sample hosts file can be found in the */<loadpath>/NSD_NRC_<R_r>/examples/hosts* directory. It is highly recommend that a modified copy of this file be used during installation.

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 3.1.3 “Configuration file” (p. 28) for more information.

Note: A sample configuration file can be found at */<loadpath>/NSD_NRC_<R_r>/examples/config.yaml*. It is highly recommend that a modified copy of this file be used during installation.

7

Copy the appropriate license file(s) into the license directory where the CLM installer bundle was extracted.

8

Perform 6.4 “To configure and enable a PKI server” (p. 52) to enable the configuration of TLS in the system.

9

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

10

Install the CLM servers. Execute the following commands:

cd bin ↵

./install.sh ↵

The CLM servers are automatically deployed on both servers.

11

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.

12

Close the open console windows.

END OF STEPS

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

4.4.1 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.

4.4.2 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 3.1.2 “Hosts file” (p. 27) for more information.

3

Copy the appropriate license file(s) into the *license/* folder where the CLM installer bundle was extracted.

4

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

5

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.

6

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

```
cd bin ↲
```

```
./install.sh ↲
```

7

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.

8

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

5 Post-installation activities

5.1 Introduction

5.1.1 Overview

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

5.2 To uninstall an CLM system

5.2.1 Purpose

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

5.2.2 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 3.2 “To install a standalone CLM system” (p. 32), 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

6 Security

6.1 Introduction

6.1.1 Overview

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

6.1.2 Automated TLS deployment using an 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 6.4 “To configure and enable a PKI server” (p. 52) for information about using an 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.

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

6.2 Data privacy

6.2.1 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 6-1 CLM data privacy

Category	Description
Local user data (local authentication)	
Type of data	• Username and password • IP address
Purpose	• Authentication of local NSP users • IP address provides accountability of individual product access
Storage	• Local database • Logs
Retention	Data is retained in the database until an authorized user deletes it. Log retention time can vary based on log file size and the number of log backups.
Processing	Local user data is processed for the stated purpose.
Access	Authorized users

Table 6-1 CLM data privacy (continued)

Category	Description
Safeguards	• Additional local users must be created by an authorized user. • Database access is restricted to authorized users. • TLS secures data in transit. • Passwords for local users are hashed before they are stored. • Log file access is restricted to authorized users.
Comments	Local authentication is performed using a local database of users and a local security scheme.
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
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

6.3 To configure the NSP security statement

6.3.1 Purpose

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

6.3.2 Steps

Install the CLM and start the nspOS

1

Perform one of the following:

- Install your standalone CLM system, as described in 3.2 “To install a standalone CLM system” (p. 32).
- Install your redundant CLM system, as described in 4.2 “To install a redundant CLM system” (p. 39).

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, go to More → Settings → NSP System Settings → Security Statement.

5

Add the appropriate security statement.

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

END OF STEPS

6.4 To configure and enable a PKI server

6.4.1 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.

6.4.2 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 bundle 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 bundle 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 15.

8

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

```
**********************************************************************************
No Root CA detected on the filesystem. This should only happen on
initial installation!
**********************************************************************************
Create new 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.
date time Using Root CA from disk, and serving requests on port port
```

14

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

15

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*`]`

16

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.

17

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

18

Close the console window.

END OF STEPS

6.5 To migrate to the PKI server

6.5.1 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.

6.5.2 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 bundle 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

6.6 To generate a keystore

6.6.1 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.

6.6.2 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 NSP 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

6.7 To suppress security warnings in CLM browser sessions

6.7.1 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.

6.7.2 Steps

1

Perform one of the following.

a. If you deployed TLS using an 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

7 Backup and restore

7.1 Introduction

7.1.1 Overview

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

7.2 To manually backup the PostgreSQL database

7.2.1 Purpose

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

Note: Backups of the database is taken automatically each day through a cron job and 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.

7.2.2 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

7.3 To restore the PostgreSQL database

7.3.1 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.

7.3.2 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.

7.3.3 Steps

1

Backup the PostgreSQL database as described in 7.2 “To manually backup the PostgreSQL database” (p. 59).

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 Overview

CLM software is delivered to registered customers through the Electronic Delivery→Downloads portal of Nokia Online Customer Support, or OLCS. If you are a new customer and require access, contact your sales or support representative for information about registration.

After you register, you can download the CLM software from OLCS.

Note: It is strongly recommended that you verify the checksum of each software package or file that you download from OLCS. 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.1.2 Software delivery

The CLM software on the Electronic Delivery > Downloads portal 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.

Once you have selected items for download and clicked Next, you must choose your download method. Click Help for information about the various download methods available.

A.2 Documentation

A.2.1 Overview

CLM documentation consists of:

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

A.2.2 Documentation delivery

CLM documentation is available on the Documentation Center of the OLCS site. If you are a new user and require access to this service, contact your Nokia support representative.

From the NSP product documentation page on the OLCS Documentation Center, you can:

- filter by release, category, and content type
- 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, whereas Release Notices and Release Descriptions are filed under “Release Information.”

Application help

Help is available from the Launchpad. The application help contains the same content as the *CLM User Guide*.

Documentation alerts

To receive an email when new or reissued CLM customer documents are posted to OLCS, subscribe to the notification service for documentation alerts.