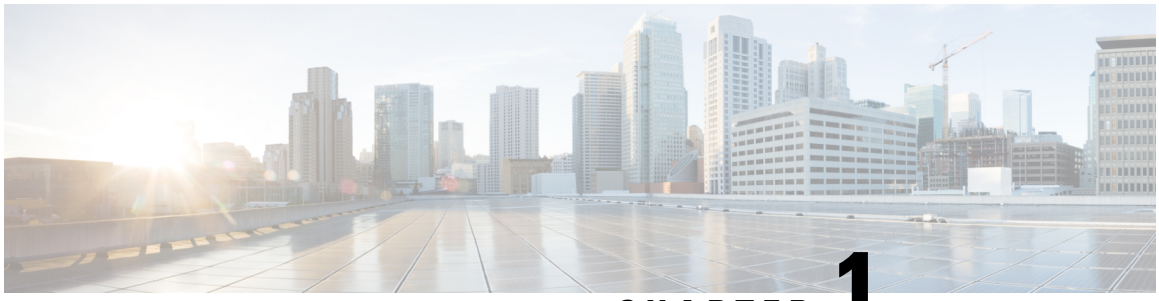
Cisco WAE 7.6.x Installation Guide

First Published: 2022-06-06

Last Modified: 2025-01-13

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883



CHAPTER 1

Cisco WAE Overview

Cisco WAE Overview

The Cisco WAN Automation Engine (WAE) platform is an open, programmable framework that interconnects software modules, communicates with the network, and provides APIs to interface with external applications.

Cisco WAE provides the tools to create and maintain a model of the current network through the continual monitoring and analysis of the network and the traffic demands that is placed on it. At a given time, this network model contains all relevant information about a network, including topology, configuration, and traffic information. You can use this information as a basis for analyzing the impact on the network due to changes in traffic demands, paths, node and link failures, network optimizations, or other changes.

Some of the important features of Cisco WAE platform are:

- Traffic engineering and network optimization—Compute TE LSP configurations to improve the network performance, or perform local or global optimization.
- Demand engineering—Examine the impact on network traffic flow of adding, removing, or modifying traffic demands on the network.
- Topology and predictive analysis—Observe the impact to network performance of changes in the network topology, which is driven either by design or by network failures.
- TE tunnel programming—Examine the impact of modifying tunnel parameters, such as the tunnel path and reserved bandwidth.
- Class of service (CoS)-aware bandwidth on demand—Examine existing network traffic and demands, and admit a set of service-class-specific demands between routers.

Cisco WAE Applications

Cisco WAE applications work with the Cisco WAE platform software, providing greater insight into your network.

- Cisco WAE Design—The WAE Design GUI provides graphical layouts of the network, showing views of different regions and layers, as well as utilizations and routings. It lets you model, simulate, and analyze

failures, changes, and impact of traffic growth, as well as optimize your network for maximum efficiency. Simulation tools let you perform interactive what-if simulations by:

- Failing objects
- Modifying the network topology
- Creating and changing traffic demands (which simulate traffic flows)
- Modifying routing configurations

For installation instructions, see the Cisco WAE Design GUI Installation Guide.

- Cisco WAE Live—Cisco WAE Live provides immediate and easy access to both current and historical network data. Combined, the Explore, Analytics, and Map tools offer a flexible and interactive means of finding summarized aggregate views or quickly narrowing the search on network data to only relevant details.

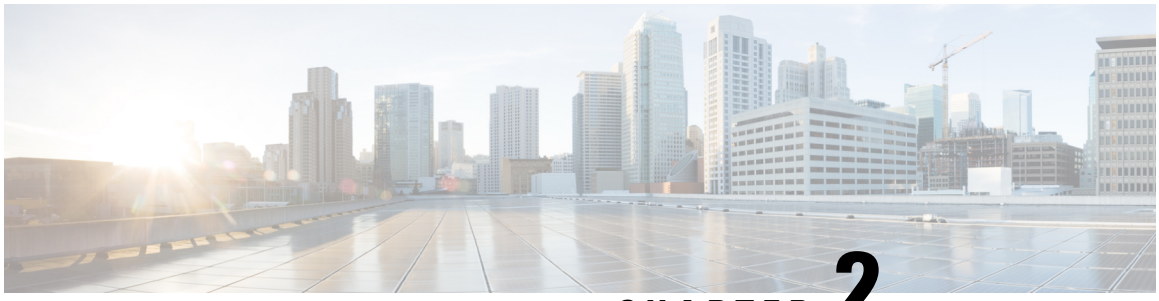
  For installation instructions, see Install Cisco WAE Live, on page 29.

- Bandwidth on Demand—The Bandwidth on Demand (BWoD) application utilizes the near real-time model of the network offered by WMD to compute and maintain paths for SR policies with bandwidth constraints delegated to WAE from SR-PCE. In order to compute the shortest path available for a SR policy with a bandwidth constraint and ensure that path will be free of congestion, a Path Computation Element (PCE) must be aware of traffic loading on the network. The WAE BWoD application extends the existing topology-aware PCE capabilities of SR-PCE by allowing delegation of bandwidth-aware path computation of SR policies to be sub-delegated to WAE through a new SR-PCE REST API. Users may fine-tune the behavior of the BWoD application, affecting the path it computes, through selection of application options including network utilization threshold (definition of congestion) and path optimization criteria preferences.

  For information about enabling, configuring, and properly shutting down the BWoD application, see the "Automation Applications" chapter in the Cisco WAE User Guide.

- Bandwidth Optimization—The Bandwidth Optimization application is an approach to managing network traffic that focuses on deploying a small number of LSPs to achieve a specific outcome in the network. Examples of this type of tactical traffic engineering are deploying LSPs to shift traffic away from a congested link, establishing a low-latency LSP for priority voice or video traffic, or deploying LSPs to avoid certain nodes or links. WAE provides the Bandwidth Optimization application to react and manage traffic as the state of the network changes.

  For information about enabling, configuring, and properly shutting down the Bandwidth Optimization application, see the "Automation Applications" chapter in the Cisco WAE User Guide.



CHAPTER 2

Cisco WAE Installation Requirements

Cisco WAE requirements vary depending on which components are installed together. This section provides general guidelines and minimum requirements for individual components installed on a single server, unless otherwise specified.

This section contains the following topics:

Cisco WAE Server Requirements

You can install Cisco WAE on a server that meets the following requirements.

Operating System	Software	CPU	Memory	Hard Drive
Linux-x86_64	Centos/RHEL 8.1 with latest patches **Note** Cisco WAE 7.6.4 is validated on RHEL 9.4.	16+ Core	64 GB	Minimum: 500 GB Recommended: 1TB

Important Notes

- Cisco WAE 7.6.0 software is qualified on CentOS/RHEL 8.1. Cisco WAE VM is compatible with the ESXi versions which support RHEL 8.1.
- Cisco WAE 7.6.4 is validated on RHEL 9.4.

- If you use a hard disk with greater than 2048 bytes physical sector size for data store filesystem, you may get the following error on startup:

  Failed to initialize the system ..

  For the WAE Live database to function as designed, the recommended block size is equal to or less than 2048 bytes for data store filesystem.

- Only Linux distributions available in English are supported.

- Modify the `/etc/security/limits.conf` file by adding or updating the following lines to make sure your hardware supports sufficient number of threads for starting poller:

  ```
  [user] soft stack 8192
  [user] soft nproc 257805
  ```

  Where `[user]` is the userid which starts the WAE process.

  Set the file descriptor limits:

  ```
  [user]     soft    nofile  1000000
  [user]     hard    nofile  1000000
  # End of file
  ```

Note Reboot the server after modifying the `limits.conf` file.

- In case of VM, there is no specific requirement for resource allocation in terms of provisioning or reservation, as this is not an OVA based installation. The provisioning and reservation depend on the customer environment and network.

Required Software Packages

Software	Version
JDK/JRE	OpenJDK 11.0.7 64-bit **Note** If multiple versions of Java are installed on the setup, set JAVA_HOME/JRE_HOME to Java 11 in **Environment** section of the **wae.ini** file.
Perl	5.16.3
fontconfig	2.13.1
Python	Cisco WAE 7.6.4 supports Python 3.7.6, while earlier versions of Cisco WAE support Python 3.6.x. **Note** `/usr/bin/python3` must point to the installed python.
python3-paramiko.noarch	2.7.2
python3-lxml	4.6.2

Software	Version
python3-requests	2.20.0
redhat-lsb	4.1 This is required for the License Server. For more information, see the "WAE Design Floating License Server" chapter in the Cisco WAE Design GUI Installation Guide.
Supervisor	Cisco WAE 7.6.4 supports version 4.2.5, while earlier versions of Cisco WAE support version 4.2.1.
Which	2.21
ncurses-compat-libs	6.1.7

Note `python3-paramiko`, `python3-lxml`, `python3-requests` and `supervisor` must be installed after adding `epel-release` repository.

NetFlow Requirements

NetFlow Collection - (Exclusive, apart from WAE server requirements) Memory size and CPU per server

Centralized NetFlow **(Server where snapshot resides)**		
	Memory	CPU
pmacct	32 GB	
flow_cluster_broker	2 GB	
flow_cluster_controller	2 GB	
flow_cluster_agent	8 GB	
flow_collector_ias flow_collector_dmd	8 GB	
TOTAL	**52 GB**	**8+ Cores**

Distributed NetFlow **(Server where the agent resides)**		
	Memory	CPU
pmacct	32 GB	

Distributed NetFlow **(Server where the agent resides)**		
flow_cluster_agent	8 GB	
TOTAL	**40 GB**	**8+ Cores**

Distributed NetFlow **(Server where the snapshot resides)**		
	Memory	**CPU**
flow_cluster_broker	2 GB	
flow_cluster_controller	2 GB	
flow_cluster_ias flow_cluster_dmd	8 GB	
TOTAL	**12 GB**	**8+ Cores**

Note

- One flow collection server (pmacct) is required per 100 Mbps of flow export bandwidth.
- Only English Linux is supported.
- Flow collection requires Linux Kernel 2.6.32 or greater.
- The memory requirement listed above per collection server instance is based on the assumption of an approximate figure of 100 Mbit/s of NetFlow traffic.

Optical Collection Agents

Vendor	Supported Node Version	Software
Cisco	Cisco Network Convergence System (NCS) 2000 Series Routers, Releases 11.3	Cisco Evolved Programmable Network Manager 5.1.4
Cisco	Cisco Network Convergence System (NCS) 2000 Series Routers, Releases 12.3	Cisco Evolved Programmable Network Manager 6.0

Collection from Network Service Orchestrator

Software/Driver	Version
IOS NED	6.66.1

Software/Driver	Version
IOS-XR NED	7.30.1
Junos NED	4.6.17
Network Service Orchestrator	5.4.2
Traffic Engineering	Contact your Cisco WAE representative.

Scale Support

Parameters	Scale
Total number of Network Devices	3000
Total number of Interfaces	100000
Total number of Demands	100000
Total number of Policies (SR or RSVP or Both)	5000

Cisco WAE Design Requirements

WAE Design is a 64-bit installation on the Linux operating systems.

Operating System	Software	CPU	Memory
Linux-x86_64	RHEL 8.4 with latest patches **Note** Cisco WAE 7.6.4 is validated on RHEL 9.4.	Minimum: 8 Cores Recommended: 16+ Cores	Minimum: 16 GB Recommended: 64 GB

Important Notes

- A standalone WAE Design system does not require the use of WAE Collector.
- Only Linux distributions available in English are supported.
- A Perl (5.10+) installation is required for some WAE features. See Required Software Packages, on page 4.
- A Python installation is required for some WAE features. See Required Software Packages, on page 4.
- FlexLM license: For more information, see *Cisco WAE Design GUI Installation Guide*.
- Only English Linux is supported.

WAE Live Requirements

Note WAE Live must be installed on a separate server than the WAE server.

Requirement	~1000 Node Network	~2000 Node Network
Supported operating system	RHEL 8.4 **Note** Cisco WAE 7.6.4 is validated on RHEL 9.4.	RHEL 8.4 **Note** Cisco WAE 7.6.4 is validated on RHEL 9.4.
CPU	8 cores, 16 threads (or 16 vCPUs)	16 cores, 32 threads (or 32 vCPUs)
Memory	24 GB	48 GB
Disk speed	200 MB/s	320 MB/s
Disk size	3 TB	10 TB
Number of network objects	100,000	500,000

Note
- Only English Linux is supported.
- Other Red Hat Enterprise Linux distributions should work, but are not officially supported.

Kernel Parameters

Kernel Parameters	Value
SHMALL	4294967296
SHMMAX	4398046511104
SHMMNI	4096
SEMMNS	32000
SEMMSL	250
SEMOPM	32
Maximum number of file descriptors	65535

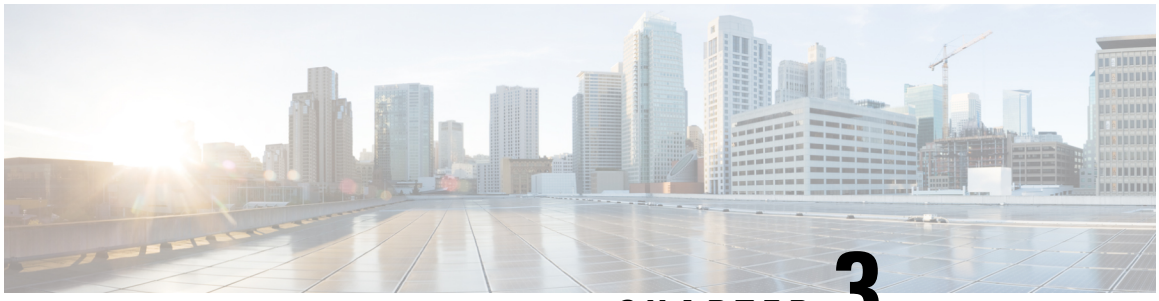
Supported Web Browsers

Browser	Version
Google Chrome	62 or later
Firefox	56 or later

Cisco WAE Ports

Port	Protocol	Type	Description
*:8080	TCP	Listening	Cisco WAE Server
*:8443	TCP	Listening	Cisco WAE Server, Live Server
2022 - 2023	TCP	Listening and outgoing	Cisco WAE Server
*:2024	TCP	Listening	Cisco WAE Server
4569	TCP	Listening and outgoing	Cisco WAE Server
4570	TCP	Listening	Cisco WAE Server HA
8080	TCP	Outgoing	XTC collection
164	UDP	Outgoing	SNMP-based NIMOs
22	TCP	Outgoing	Collection via Telnet
23	TCP	Outgoing	Collection via SSH
*:2181	TCP	Listening	Message broker
*:9092 - 9094	TCP	Listening	Message broker
8443	TCP	Listening and outgoing	Cisco WAE Optical EPNM Agent
8161	TCP	Listening	NetFlow JMS OOB
61616	TCP	Listening	NetFlow JMS IB
9090	TCP	Listening	NetFlow HTTP
2100	UDP	Listening	NetFlow
179	TCP	Listening	NetFlow BGP

Port	Protocol	Type	Description
*:8843	TCP	Listening	Cisco WAE Coordinated Maintenance (standalone or as part of WAE Server)



CHAPTER 3

Install Cisco WAE

This section contains the following topics:

Install and Configure Supervisor

Install and configure supervisor before installing WAE.

Note

- The following configuration steps work only when supervisor is installed using yum. If supervisor is installed using any other method, it has to be configured to run **supervisorctl** as a non root user.
- For installing and configuring supervisor on RHEL 9.4 (Cisco WAE 7.6.4), see Install and Configure Python and Supervisor on RHEL 9.4, on page 12.

Procedure

Step 1 Install supervisor and verify.

```
sudo yum install -y epel-release
sudo yum install -y supervisor
```

```
supervisord -version
4.2.5
```

Note
Cisco WAE 7.6.4 supports supervisor 4.2.5.

Step 2 Create directories with write permissions for the OS user running WAE.

```
sudo mkdir -p /opt/supervisor/run
sudo mkdir -p /opt/supervisor/log
sudo chown -R [USER-NAME]:[GROUP-NAME] /opt/supervisor
```

Step 3 Update supervisor configuration to not run as a root user.

Point the pid file to `/opt/supervisor/run/supervisor.pid` and user as the OS user running WAE.

Open `/etc/supervisord.conf` as root and edit.

- In the `[unix_http_server]` section:

  - Change `;file=/var/run/supervisor/supervisor.sock` to `file=/opt/supervisor/run/supervisor.sock`

  - Change `;chown=nobody:nogroup` to `chown=[USER-NAME]:[GROUP-NAME]`

- In the `[supervisord]` section:

  - Change `;logfile=/var/log/supervisor/supervisord.log` to `logfile=/opt/supervisor/log/supervisord.log`.

  - Change `;pidfile=/var/run/supervisord.pid` to `pidfile=/opt/supervisor/run/supervisord.pid`

  - Change `;minfds=1024` to `minfds=1000000`

  - Change `;minprocs=200` to `minprocs=257805`

  Note
  Do not set the user under the `[supervisord]` section.

- In the `[supervisorctl]` section:

  - Change `;serverurl=unix:///var/run/supervisor/supervisor.sock` to `serverurl=unix:///opt/supervisor/run/supervisor.sock`

Step 4 Start Supervisor.

```
sudo systemctl start supervisord
sudo supervisorctl status all
```

Step 5 Enable supervisor to start during system startup.

```
sudo systemctl enable supervisord
sudo systemctl status supervisord
```

Install and Configure Python and Supervisor on RHEL 9.4

Cisco WAE 7.6.4 supports the installation of Python 3.7.6 and supervisor 4.2.5 on RHEL 9.4.

Install Python 3.7.6 on RHEL 9.4

Follow these steps to install Python 3.7.6 on RHEL 9.4.

Before you begin

Install the dependencies needed for Python 3.7.6 using this command:

```
sudo dnf install \
      gcc gcc-c++ gdb lzma glibc-devel libstdc++-devel openssl-devel \
      readline-devel zlib-devel libffi-devel bzip2-devel xz-devel \
      sqlite sqlite-devel sqlite-libs libuuid-devel gdbm-libs perf \
      expat expat-devel mpdecimal
```

Procedure

Step 1 Install Python 3.7.6.

```
sudo dnf install python3.7.6
```

If the above command does not work, then follow these steps.

a) Download the Python 3.7.6 tar file from the official Python website.

```
wget https://www.python.org/ftp/python/3.7.6/Python-3.7.6.tgz
```

b) Extract the contents of the downloaded tar file.

```
tar xzf Python-3.7.6.tgz
```

c) Navigate to the Python-3.7.6 directory created.

```
cd Python-3.7.6
```

d) Run the configure script to prepare the build environment.

```
sudo ./configure --enable-optimizations --prefix=/usr
```

e) Compile the Python source code.

```
sudo make
```

f) Install Python 3.7.6.

```
sudo make altinstall
```

Step 2 Update the symbolic link named `python3` in the `/usr/bin/` directory to point to `python3.7`.

```
sudo ln -sf /usr/bin/python3.7 /usr/bin/python3
```

Step 3 Update the symbolic link named `python` in the `/usr/bin/` directory to point to `python3`.

```
sudo ln -sf /usr/bin/python3 /usr/bin/python
```

Step 4 If the `/usr/lib/python3.7/` folder does not have the `lib-dynload` directory, run this command to create a symbolic link and point it to `lib-dynload` which is in the `lib64` folder.

```
sudo ln -sf /usr/lib64/python3.7/lib-dynload/ /usr/lib/python3.7/lib-dynload
```

Install and configure Supervisor on RHEL 9.4 using pip

Follow these steps to install and configure supervisor on RHEL 9.4 using pip.

Procedure

Step 1 Install the `supervisor` package.

```
sudo pip3.7 install supervisor
```

Step 2 Check the version of the `supervisord` program installed.

```
supervisord -version
4.2.5
```

Step 3 Create the `/usr/lib/systemd/system/supervisord.service` file manually with root user.

```
sudo vi /usr/lib/systemd/system/supervisord.service
```

Step 4 Add this content in the `/usr/lib/systemd/system/supervisord.service` file.

```
[Unit]
Description=Process Monitoring and Control Daemon
After=rc-local.service nss-user-lookup.target

[Service]
Type=forking
ExecStart=/usr/bin/supervisord -c /etc/supervisord.conf
RuntimeDirectory=supervisor
RuntimeDirectoryMode=755

[Install]
WantedBy=multi-user.target
```

Step 5 Generate the default supervisord configuration file and save it to `/etc/supervisord.conf`.

```
sudo su
echo_supervisord_conf > /etc/supervisord.conf
exit
```

Step 6 Create directories with write permissions for the OS user running Cisco WAE.

```
sudo mkdir -p /opt/supervisor/run
sudo mkdir -p /opt/supervisor/log
sudo chown -R [USER-NAME]:[GROUP-NAME] /opt/supervisor
```

Step 7 Update supervisor configuration to not run as a root user.

Point the pid file to `/opt/supervisor/run/supervisor.pid` and user as the OS user running WAE.

Open `/etc/supervisord.conf` as root and edit.

- In the `[unix_http_server]` section:
  - Change `;file=/var/run/supervisor/supervisor.sock` to `file=/opt/supervisor/run/supervisor.sock`
  - Change `;chown=nobody:nogroup` to `chown=[USER-NAME]:[GROUP-NAME]`
- In the `[supervisord]` section:
  - Change `;logfile=/var/log/supervisor/supervisord.log` to `logfile=/opt/supervisor/log/supervisord.log`

- Change `;pidfile=/var/run/supervisord.pid` to `pidfile=/opt/supervisor/run/supervisord.pid`

- Change `;minfds=1024` to `minfds=1000000`

- Change `;minprocs=200` to `minprocs=257805`

Note
Do not set the user under the `[supervisord]` section.

- In the `[supervisorctl]` section:

  - Change `;serverurl=unix:///var/run/supervisor/supervisor.sock` to `serverurl=unix:///opt/supervisor/run/supervisor.sock`

- Uncomment the `[include]` section and replace the line under it with `files = supervisord.d/*.ini`

Step 8 Create the `/etc/supervisord.d` directory.

```
sudo mkdir /etc/supervisord.d
```

Step 9 Start Supervisor.

```
sudo systemctl start supervisord
sudo supervisorctl status all
```

Step 10 Enable supervisor to start during system startup.

```
sudo systemctl enable supervisord
sudo systemctl status supervisord
```

Verify WAE Image

Procedure

Step 1 Download the Cisco WAE 7.6.x software package from Cisco Download Software site.

Step 2 The certificate and digital signature are both embedded in the downloaded file - `wae-linux-v7.6.x.signed.bin`.

Step 3 Run the self-extracting signed binary. This extracts the Release Binary and validates using the signature file.

Verification of signed image

```
[admin@wae-vm-21 workspace.signed]$ ./wae-linux-v7.6.x.signed.bin
Unpacking...
Verifying signature...
Downloading CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...
Successfully downloaded and verified crcam2.cer.
Downloading SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...
Successfully downloaded and verified innerspace.cer.
Successfully verified root, subca and end-entity certificate chain.
Successfully fetched a public key from WAE-CCO_RELEASE.cer.
Successfully verified the signature of wae-linux-v7.6.x.bin using WAE-CCO_RELEASE.cer
```

Step 4 The generated `wae-linux-v7.6.x.bin` is the Linux installer for WAE.

Install Cisco WAE

Before you begin

Note If you want to upgrade from an older WAE 7.x release to WAE 7.6.x, see Upgrade from Cisco WAE 7.x, on page 21

- If one does not yet exist, create a UNIX user (assigned to a group). You must be this UNIX user to run installation.
- Make sure Java 11 and Python 3.6.x are installed on the system. `JAVA_HOME` environment variable is pointing to jdk-11.0 and `/usr/bin/python3` must point to the installed python.

Note Cisco WAE 7.6.4 supports Python 3.7.6, while earlier versions of Cisco WAE support Python 3.6.x.

- Make sure supervisor is installed and configured. See Install and Configure Supervisor, on page 11.
- Download and verify the digitally signed Cisco WAE 7.6.x image. See Verify WAE Image, on page 15.
- Make sure that `requests.auth` python package is installed for the BW-OPT application to function in WAE.

Procedure

Step 1 Stop WAE if running.

Step 2 Change permission of the install file using the command:

```
chmod +x wae-linux-v7.6.x.bin
```

Step 3 Run the installer specifying the target directory.

```
./wae-linux-v7.6.x.bin <wae-dir>
```

Step 4 Navigate to installation directory to source `waerc`. Setup environment and create a runtime directory specifying the path.

```
cd  <wae-dir>
source waerc
wae-setup --dest <target-runtime-dir>
```

Step 5 You are prompted to set the Cisco WAE admin password.

```
WAE admin password:
Confirm password:
```

Step 6 After installing and setting up wae (i.e. after running wae-setup), create a soft link to the `wae.ini` file from inside `/etc/supervisord.d/` and add WAE config to supervisor.

```
sudo ln -sf <target-runtime-dir>/wae.ini /etc/supervisord.d/
```

Note

- Execute this step only after supervisor is installed and configured.
- If you want to use an external-executable-nimo based network which needs `JAVA_HOME/JRE_HOME` to be set, edit the section `[program:waectl]` inside `target-runtime-dir/wae.ini` file to include `JAVA_HOME="valid_jdk_path"` inside environment.

  For example, under `[program:waectl]` edit to add:

  ```
  JAVA_HOME:environment=HOME="/home/wae", NCS_JAVA_VM_OPTIONS="-Xmx32G -Xms16G -XX:+UseG1GC
  -XX:+HeapDumpOnOutOfMemoryError -XX:HeapDumpPath=/home/wae/test/run/logs/
  -Djava.io.tmpdir=/home/wae/test/run/work/", TMPDIR="/home/wae/test/run/work/", JAVA_HOME="/usr/"
  ```

- For the new wae.ini changes to come into effect, execute **`supervisorctl update`**.

Step 7 Update supervisor configuration.

```
sudo supervisorctl update
```

Step 8 Start WAE process

```
sudo supervisorctl start wae:*
wae:zookeeper: started
wae:waectl: started
wae:kafka: started
wae:wae-monitor: started
```

Note

- `wae:waectl` is the WAE program.
- `wae:kafka` and `wae:zookeeper` are required for traffic collection and internal messaging.
- `wae:wae-monitor` is the monitoring service.
- `wae:logrotate` is for log rotation.

Step 9 Check status of WAE process

```
sudo supervisorctl status
wae:kafka RUNNING pid 1540, uptime 28 days, 14:03:40
wae:logrotate RUNNING pid 1178, uptime 28 days, 15:10:11
wae:wae-monitor RUNNING pid 11520, uptime 0:00:12
wae:waectl RUNNING pid 1177, uptime 28 days, 15:10:11
wae:zookeeper RUNNING pid 1736, uptime 28 days, 14:03:39
```

Note
To stop all WAE process, use the command:

```
sudo supervisorctl stop wae:*
```

Step 10 To migrate configurations from a WAE 7.x.x release to WAE 7.6.x release, use the Cisco WAE upgrade script from Cisco Download Software site.

Note
If the server/VM is restarted, all the WAE services are not restarted automatically and they will be in the stopped state. They can be started using the command mentioned in Step 8.

Install Multi WAE

Before you begin

Important Cisco WAE 7.6.4 does not support Multi WAE.

- Install Ansible version 2.10.7 or higher based on python3. Use the following command:

  ```
  sudo yum install ansible
  ```

- Install Java 11 on all remote hosts.
- Install Python3 on all remote hosts as well as in the host where playbooks are run.

Note

- On RHEL 8.4, run the playbook from a terminal where **waerc** is not sourced.
- Restart WAE in the scale primary whenever the number of splits increases.

- Enable passwordless ssh between servers participating in multi WAE (including self ssh).

Procedure

Step 1 Export the ansible.cfg. A custom ansible.cfg file is provided at **playbooks/ansible.cfg**. Use the command:

```
export ANSIBLE_CONFIG=<path-to-the-ansible-config-file>
```

Step 2 On the machine where you intend to run the playbook from, add the entry to self in **playbooks/known_hosts** file by doing an SSH to yourself. The Multi WAE installation only supports single **username** and **wae_dir** across different machines. You can also pass **ansible_ssh_user** from CLI while invoking the **ansible-playbook** command by passing **-u** flag.

```
ansible-playbook wae_install.yml -u <username> --ask-pass
```

Step 3 Add the following line at the end of the **playbooks/visudo** file to make sure you can run the sudo commands without password

```
<username> ALL=(ALL) NOPASSWD:ALL
```

Step 4 Modify the **playbooks/hosts** file to include the IP addresses of the machines. The **hosts** file has 3 groups: **[remote]**, **[primary]** and **[secondary]**.

```
[remote]
'element-1' ansible_ssh_user='TARGET_SSH_USER'
'element-2' ansible_ssh_user='TARGET_SSH_USER'
'element-3' ansible_ssh_user='TARGET_SSH_USER'

[primary]
'element-1' ansible_ssh_user='TARGET_SSH_USER'
```

```
[secondary]
'element-2' ansible_ssh_user='TARGET_SSH_USER'

where,

[remote]    - indicate the set of hosts in which the playbooks are to be run
[primary]   - is the host which should be set as primary when configuring HA. Must be one of the host
 present in [remote] group.
[secondary] - is the host which should be set as secondary when configuring HA. Must be one of the
host present in [remote] group.
```

Note

- **[remote]** group is compulsory for every playbook execution.
- **[primary]**, and **[secondary]** groups are required only for **ha_config** playbook execution.

Step 5 Set the input parameters required by playbooks in **group_vars/all** file. The file is present in **playbooks/group_vars/all** (refer to the following table) and execute the playbook. The following table lists the details of the available playbooks:

Table 1: Ansible Playbook Details

Playbook	Description	Parameters	Usage
wae_install.yml	The **wae_install.yml** playbook installs WAE on remote machines by copying the WAE binary and performing relevant checks and tasks that are needed to get the server up and running using supervisor.	• **WAE_USER_NAME**: WAE user (sudo capable) preexisting on all the remote machines. • **WAE_BIN_PATH**: Absolute path to the WAE binary located on the machine where **ansible-playbook** is run. • **WAE_DIR**: Absolute path of the WAE directory which will hold **wae-install** and **wae-run** directories. • **DELETE_SIGNED**: Flag used to indicate if we need to clean up signed WAE image after install is complete. Default value is **false**.	`ansible-playbook wae_install.yml -i <path_to_inventory_file> --ask-pass`
kafka_config.yml	The **kafka_config.yml** playbook deploys kafka on remote machines by setting the right configurations for the internal and external listeners.	• **WAE_DIR**: Absolute path of the WAE directory which will hold **wae-install** and **wae-run** directories	`ansible-playbook kafka_config.yml -i <path_to_inventory_file> --ask-pass`

Playbook	Description	Parameters	Usage
ha_config.yml	The **ha_config.yml** playbook deploys HA between two nodes given WAE is running.	• **WAE_USER_NAME**: WAE user (sudo capable) preexisting on all the remote machines. • **WAE_DIR**: Absolute path of the WAE directory which will hold **wae-install** and **wae-run** directories. • **WAE_HA_XML_** **TEMPLATE**: XML template containing WAE HA config to be loaded on CDB of the two nodes.	`ansible-playbook ha_config.yml -i <path_to_inventory_file> --ask-pass`

Playbook	Description	Parameters	Usage
load_config.yml	The **load_config.yml** playbook is intended to load the WAE configs on the remote WAE server.	• **WAE_DIR**: Absolute path of the WAE directory which will hold **wae-install** and **wae-run** directories. • **WAE_CFGS_SRC_DIR**: Absolute path of the directory where the configs are present on the machine where ansible-playbook is run. • **WAE_CFGS**: List of names of the config files. The files should be present at **WAE_CFGS_SRC_DIR**. • **WAE_TMP_CFGS_** **DEST_DIR**: Absolute path to a directory in remote machines where the config files will be copied. Directory will be created if it does not exist. Default value is **/tmp/wae_cfgs**.	`ansible-playbook load_config.yml -i <path_to_inventory_file> --ask-pass`

Upgrade from Cisco WAE 7.x

Before you begin

- Download the Cisco WAE upgrade script from Cisco Download Software site.
- Download and verify the digitally signed Cisco WAE 7.6.x image. See Verify WAE Image, on page 15.
- Make sure Java 11 and Python 3.6.x are installed on the system. `JAVA_HOME` environment variable is pointing to jdk-11.0 and `/usr/bin/python3` must point to the installed python.
- Install `pexpect` using the following command:

  ```
  sudo pip3 install pexpect
  ```

- Make sure supervisor is installed and configured. See Install and Configure Supervisor, on page 11.

- Disable HA before doing an upgrade. Upgrade script does not handle any configurations related to specific functional packs present in the previous WAE installation. You can:
  - Remove the configurations related to functional packs before doing the upgrade, or
  - Install WAE manually (See Install Cisco WAE, on page 16), install the functional packs in the new WAE installation and then import the configurations (see Migrate Configurations from Cisco WAE 7.x, on page 24).

Procedure

Step 1 Log in to the machine where 7.x is installed.

Step 2 Deregister Cisco WAE from Cisco Smart Software Manager (CSSM). To do this, click **Deregister** from the drop-down list available on the top right of the Smart Software Licensing page.

Step 3 Take a backup of the 7.x configuration. To do this, run the `wae_upgrade` script with `--export` option.

```
# ./wae_upgrade --export --install-dir <WAE_7.x_INSTALL_DIR> --run-dir <WAE_7.x_RUN_DIR>
--cfg-dir <dir_to_save_exported_config>

Where:
      --install-dir   indicates the directory where 7.x WAE is installed
      --run-dir       indicates the directory where the run time for 7.x WAE resides
      --cfg-dir       indicates the folder where backup of 7.x configuration must reside
```

Step 4 Install WAE 7.6.x. For details, see Install Cisco WAE, on page 16.

Step 5 Run the `wae_upgrade` script.

```
# ./wae_upgrade --upgrade --old-install-dir <WAE_7.x_INSTALL_DIR> --old-run-dir
<WAE_7.x_RUN_DIR> --new-install-dir <WAE_7.6.x_INSTALL_DIR> --new-run-dir
<WAE_7.6.x_RUN_DIR> --cfg-dir <dir_to_save_config>  --wae-bin <WAE_7.6.x_INSTALLATION_FILE>

where
--old-install-dir    indicates the directory where 7.x WAE is installed
--old- run-dir       indicates the directory where the run time for 7.x WAE resides
--new-install-dir    indicates the directory where 7.6.x WAE must be installed
--new-run-dir        indicates the directory where the run time for 7.6.x WAE will reside
--cfg-dir            indicates the folder where the config is to be saved. This config
                     will be changed to match 7.6.x and pushed to 7.6.x
--wae-bin            indicates the path to WAE 7.6.x installation file.
```

Note
The installation file passed as `--wae-bin` option is the image obtained after verifying the digitally signed Cisco WAE 7.6.x image.

Step 6 If using Smart Licensing, follow the steps in the *"Smart Licensing Configuration Workflow"* and *"Enable Smart Licensing in Cisco WAE"* sections under the *"Cisco Smart Licensing"* chapter of the Cisco WAE User Guide.

Upgrade from Cisco WAE 7.x to Multi WAE

Important Cisco WAE 7.6.4 does not support Multi WAE.

Procedure

Step 1 Upgrade your WAE installation to Cisco WAE 7.6.x. See Upgrade from Cisco WAE 7.x, on page 21.

Step 2 Use ansible playbook **load_config.yml** to configure agent and Nimo and manually configure Multi WAE on the upgraded WAE instance. See Install Multi WAE, on page 18

Step 3 Run upgrade script with export option to collect the config from the updated WAE instance. See Migrate Configurations from Cisco WAE 7.x, on page 24.

Step 4 Use the **wae_install** and **load_config** playbooks to install and configure WAE in the other WAE instances

Install Cisco WAE License

A license is required to use all the features in Cisco WAE. If you have questions about obtaining a license, contact your Cisco support representative or system administrator.

Cisco WAE supports both Cisco Smart Licensing and traditional licensing. If you would like to convert from a traditional license to Smart Licensing, see your Cisco WAE account representative. For information on the differences between the two types of licensing, refer to the Cisco Smart Licensing Overview on Cisco.com.

For information on Cisco Smart Licensing, see "Smart Licensing" chapter in *Cisco WAE User Guide.*

Install Traditional License

To install a traditional license:

Procedure

Step 1 Run `license_install` tool, and pass the name of license file (with `.lic` extension). By default, the tool merges all features that are granted by the new license with those features in an existing license.

```
license_install -file <path>/<license_name>.lic
```

Step 2 When prompted, enter the number that is associated with the directory in which you want to install the license.

Note

- If option 2 (<wae-dir>/etc) is selected, you need to reinstall the license when a new build is installed.
- If option 1 (/.cariden/etc) is selected, reinstalling the license is not necessary unless the license is expired.

- Once the license is installed, you can verify the installed licenses by running the `license_check` command.

Step 3 Stop and start WAE for the installed license to be picked up.

Install Smart License

To install a smart license:

Procedure

Step 1 See "Smart License" section in User Guide, to configure Smart License.

Step 2 Stop and start WAE for the installed license to be picked up.

Start and Stop Cisco WAE

From the Cisco WAE run-time directory, enter the relevant Cisco WAE CLI command to start or stop Cisco WAE services:

- Start WAE

```
sudo supervisorctl start wae:*
wae:zookeeper: started
wae:waectl: started
wae:kafka: started
wae:wae-monitor: started
```

- Stop WAE

```
sudo supervisorctl stop wae:*
```

Migrate Configurations from Cisco WAE 7.x

You can use the Cisco WAE upgrade script utility to migrate configurations from WAE 7.x.

Before you begin

- Download the Cisco WAE upgrade script for migrating configurations from WAE 7.x to WAE 7.6.x package from Cisco Download Software site.
- Install WAE 7.6.x and start the WAE process before you proceed with migrating configurations. See Install Cisco WAE, on page 16
- Install `pexpect` using the following command:

```
sudo pip3 install pexpect
```

- Disable HA before doing an upgrade.

- Upgrade script does not handle any configurations related to specific functional packs present in the previous WAE installation. You can:
  - Remove the configurations related to functional packs before exporting them, or
  - Install the functional packs in the new WAE installation before importing the configurations.

Procedure

Step 1 Deregister Cisco WAE from CSSM. To do this, click **Deregister** from the drop-down list available on the top right of the Smart Software Licensing page.

Step 2 To take a backup of the 7.x configuration, log in to the machine where 7.x is installed, and run `wae_upgrade` script with `--export` option.

```
# ./wae_upgrade --export --install-dir <WAE_7.x_INSTALL_DIR> --run-dir <WAE_7.x_RUN_DIR>
--cfg-dir <dir_to_save_exported_config>

Where:
      --install-dir   indicates the directory where 7.x WAE is installed
      --run-dir       indicates the directory where the run time for 7.x WAE resides
      --cfg-dir       indicates the folder where backup of 7.x configuration must reside
```

Step 3 To restore the 7.x configuration to 7.6.x, log in to the machine where 7.6.x is installed, and run `wae_upgrade` script with `--import` option.

```
# ./wae_upgrade --import  --install-dir <WAE_7.6.x_INSTALL_DIR> --run-dir
<WAE_7.6.x_RUN_DIR> --cfg-dir <dir_to_import_saved_config>

Where:
      --install-dir   indicates the directory where 7.6.x WAE is installed
      --run-dir       indicates the directory where the run time for 7.6.x WAE resides
      --cfg-dir       indicates the folder where backup of 7.x configuration resides
```

Step 4 If using Smart Licensing, follow the steps in the *"Smart Licensing Configuration Workflow"* and *"Enable Smart Licensing in Cisco WAE"* sections under the *"Cisco Smart Licensing"* chapter of the Cisco WAE User Guide.

Update Packages or Templates

If any packages or templates are updated or added in the *<wae_run_time_directory>*`/packages` directory, request a package reload using the Cisco-style WAE CLI:

```
$ packages reload
```

For example, perform a package reload when you edit the `wae.conf` file.

Troubleshoot a Cisco WAE Installation

To check the status of Cisco WAE, enter `sudo supervisorctl status`.

Cisco WAE comes with standard logging features in the YANG run time. Cisco WAE logs to multiple log files in the *<wae-run-time>*/logs directory.

The LDAP authentication logs are logged in *[wae-run-time]/logs/wae-ldap-auth.log* file. The tool located in *[wae-install-dir]lib/exec/test-java-ssl-conn* is useful to test SSL connectivity for java applications like LDAP Authentication and EPNM notifications which provide useful information to debug certification issues.

The most useful log is *<wae-run-time>*/logs/wae-java-vm.log. Most Cisco WAE packages log to this file. Some Cisco WAE packages also log to *<wae-run-time>*/logs/wae-python-vm-*<package-name>*.log. The following example shows Python-VM based logs:

```
[wae@wae logs]$ pwd
/home/wae/wae-run/logs
[wae@host logs]$ ls -ltr wae-python-vm*
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-opm-tte.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-get-plan.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-dmdmesh-creator-nimo.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-layout-nimo.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-opm-load-plan.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-dmddeduct-nimo.log
-rw-rw-r-- 1 wae wae    0 Feb 26 07:50 wae-python-vm-cisco-wae-archive.log
-rw-rw-r-- 1 wae wae 2238 Feb 26 07:50 wae-python-vm.log
-rw-rw-r-- 1 wae wae  270 Feb 26 08:20 wae-python-vm-nso_wae_nodes_insert.log
```

By default, the log level is set to INFO. You can configure logging in the following ways:

- Define the log level of various logs in the run-time directory wae.conf file. For information about the wae.conf file, see the *Cisco WAE User Guide*.
- Use the Expert Mode to set logging capabilities for some network interface modules (NIMOs). For example, you can set logging capabilities such as topology NIMOs and the lsp-snmp-nimo module. For information about the Expert Mode, see the Cisco WAE User Guide.
- Use the Cisco WAE CLI to define the log level for various NIMO components. To define the log level, enter the following command at the command line:

  ```
  admin@wae% set java-vm java-logging logger <nimo-component> level <level-x>
  ```

  Level types are level-info, level-debug, and level-all. The logs are saved to wae-java-vm.log and can be used for troubleshooting.

  The following table lists basic NIMO components.

NIMO Component	Description
com.cisco.wae	General debugging
com.cisco.wae.nimo.topo	Topology-based NIMO debugging
com.cisco.wae.nimo.lspconfig	LSP configuration through NED debugging
com.cisco.wae.nimo.lsp	LSP debugging
com.cisco.wae.nimo.snmptrafficpoller	SNMP traffic poller debugging
com.cisco.wae.dare	Aggregation debugging

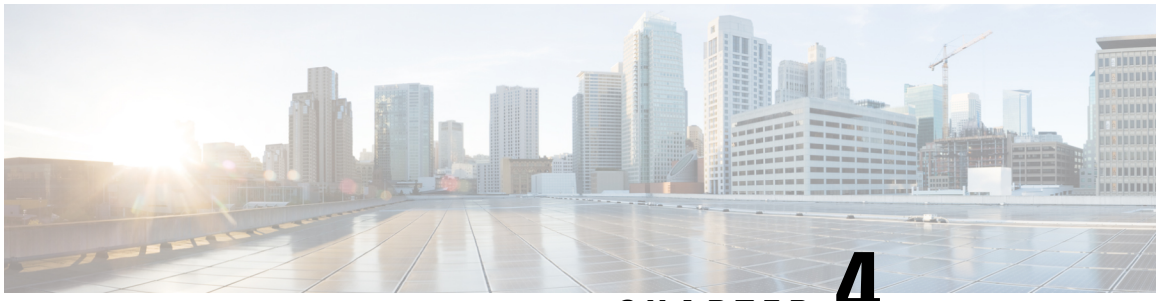
NIMO Component	Description
com.cisco.wae.nimo.optical	Optical NIMO debugging

ssh: symbol lookup error

When the waerc file is sourced, ssh and scp commands may fail due to WAE specific openssl libraries set in the LD_LIBRARY_PATH environment variable by waerc. The error message `ssh: symbol lookup error: /lib64/libk5crypto.so.3: undefined symbol: Camellia_cbc_encrypt, version OPENSSL_1_1_0` may appear.

To resolve this issue, use any of the following steps:

- Use ssh, scp, or any other operation which uses openssl on a terminal session where waerc is not sourced.
- If you are using ssh or scp commands after sourcing waerc, then set the LD_LIBRARY_PATH environment variable to the value it had before sourcing waerc.



CHAPTER 4

Install Cisco WAE Live

This section contains the following topics:

Verify WAE Live Image

Procedure

Step 1 Download the Cisco WAE Live 7.6.x software package from Cisco Download Software site.

Step 2 The certificate and digital signature are both embedded in the downloaded file - `WAE-Live-7.6.x-Linux_x86-64.signed.bin`.

Step 3 Run the self-extracting signed binary. This extracts the Release Binary and validates using the signature file.

Verification of signed image

```
[admin@wae-vm-21 workspace.signed]$ ./WAE-Live-7.6.x-Linux_x86-64.signed.bin
Unpacking...
Verifying signature...
Downloading CA certificate from http://www.cisco.com/security/pki/certs/crcam2.cer ...
Successfully downloaded and verified crcam2.cer.
Downloading SubCA certificate from http://www.cisco.com/security/pki/certs/innerspace.cer ...
Successfully downloaded and verified innerspace.cer.
Successfully verified root, subca and end-entity certificate chain.
Successfully fetched a public key from WAE-CCO_RELEASE.cer.
Successfully verified the signature of WAE-Live-7.6.x-Linux_x86-64.bin using WAE-CCO_RELEASE.cer
```

Step 4 The generated `WAE-Live-7.6.x-Linux_x86-64.bin` is the Linux installer for WAE.

Install Cisco WAE Live

Before you begin

- Cisco WAE Live cannot be installed on the same machine where the Cisco WAE 7.6.x server software is installed.
- Confirm that the Cisco WAE Live server requirements are met (see WAE Live Requirements, on page 8).
- Do not install Cisco WAE Live as a root user.
- Confirm that you have a Cisco WAE 7.6.x Live license on the server.
- Make sure Java 11 is installed on the system and `JAVA_HOME` environment variable is pointing to jdk-11.0.
- Enable `Automatic Update of Date and Time` for the server and select `Date and Time` format to be 24hr. This is not required if using NTP server.
- WAE Live is installed in `$CARIDEN_ROOT/software/live`. For example: `/opt/wae/software/live` or `/opt/cariden/software/live`.
- If you plan to migrate Cisco WAE Live 6.4.9 or older data, first upgrade to Cisco WAE Live 6.4.10. Then, enter the same installation directory that was used in Cisco WAE Live 6.4.x when prompted to install Cisco WAE Live 7.6.x. For example, if `$CARIDEN_ROOT` is defined as `/opt/cariden` in Cisco WAE Live 6.4.x, then confirm that `$CARIDEN_ROOT` in Cisco WAE 7.6.x is also defined as `/opt/cariden`.

Procedure

Step 1 Navigate to and download the Cisco WAE Live package from the Cisco Download Software site.

Step 2 Log in to the server, copy the Cisco WAE Live package `WAE-Live-7.6.x-Linux_x86-64.signed.bin` to a local directory, and start a bash shell.

Step 3 Unpack the signed package.

```
# ./WAE-Live-7.6.x-Linux_x86-64.signed.bin
```

Step 4 Provide executable access to the .bin file.

```
# chmod +x WAE-Live-7.6.x-Linux_x86-64.bin
```

Step 5 Install the Cisco WAE Live package.

```
# bash WAE-Live-7.6.x-Linux_x86-64.bin
```

Step 6 If prompted, install the required software packages using the yum command.

Step 7 Follow the installation prompts.

Step 8 After installation, set environment variables and source ~/.profile to get the necessary settings.

```
# source ~/.profile
```

Step 9 Install Cisco WAE Live data store. For more information, see Install WAE Live Data Store, on page 36.

Step 10 Start Cisco WAE Live services.

```
# wae-live-start
```

Note
The data store must be configured before starting Cisco WAE Live.

Step 11 Start one of the supported browsers and enter **`https://`***server-ip*`:8443`**, where *server-ip* is the IP address of the server on which you have WAE Live installed. The default password for the **`admin`** user is "admin". The default password for the **`user`** user is "user". You will be prompted to change the default login credentials upon first login.

Upgrade from Cisco WAE Live 7.1.x to Cisco WAE Live 7.6.x

Before you begin

You must have Cisco WAE 7.1 or later installed to perform this upgrade. For Cisco WAE 6.4.x installations, see Migrate WAE 6.4.10+ Live Data to WAE Live 7.1.x, on page 31.

Procedure

Step 1 Stop the web server and mld.

```
# wae-live-stop
```

Step 2 Install Cisco WAE Live 7.6.x. For more information, see Install Cisco WAE Live, on page 30.

Step 3 Execute the upgrade.

```
# mld -action upgrade
```

Step 4 Start the web server and mld.

```
# wae-live-start
```

Migrate WAE 6.4.10+ Live Data to WAE Live 7.1.x

On Live 7.1, install Live software to the same location as in live 6.4.10.

Before you begin

- ***You can only migrate data from WAE 6.4.10 or later to WAE 7.1.x.*** If you have an earlier WAE 6.x release installed, you must upgrade to at least WAE 6.4.10 before proceeding with the WAE 7.1.x upgrade.
- WAE Live 7.1.x and data store must be installed on a different machine than WAE 6.4.x. For installation steps, see Install Cisco WAE Live, on page 30. In addition, the WAE Live 7.1.x installation directory and data store (mld) options must use the same directory path and mld options that was used for the WAE

6.4.x installation. For example, if WAE 6.4.10 was installed on `/opt/cariden`, then you must also install WAE Live 7.1.x in `/opt/cariden` in another server. mld parameters, such as CPUs, memory, storage, and so forth, must also have the same values. To view existing mld parameters, you can look in the config.xml file.

- The WAE Live 7.1.x data store must be installed before doing this procedure. For data store installation instructions, see Install WAE Live Data Store, on page 36.
- You must continue to use the same WAE 6.4.x user ID (UID) and group ID (GID) after upgrading to WAE 7.1.x.

Procedure

Step 1 Install WAE Live 7.1 software to the same location as WAE Live 6.4.10.

If in WAE Live 6.4.10, `CARIDEN_ROOT = /opt/cariden`, install WAE Live 7.1 to `/opt/cariden`.

Make sure the license for Live 7.1 is setup.

Step 2 On WAE Live 7.1 server, install `mld` with the same parameters as in WAE Live 6.4.10.

Example:

parameters

```
-cpus
-mem
-storage
-mldata
-datastore
-backup
```

Step 3 From the WAE Live 7.1.x server, stop the web server.

```
embedded_web_server -action stop
```

Step 4 From the WAE Live 6.4.x Live server, do the following:

a) Stop the web server.

```
service wae-web-server stop
```

b) Back up the WAE Live data store. For example:

```
ml_backup -L 0
```

Step 5 From the WAE Live 7.1.x server:

a) Back up the WAE Live data store. For example:

```
ml_backup -L 0
```

b) Edit parameters in `$CARIDEN_ROOT/software/mld/current/scripts/sqlhosts.ml`.

Example:

On the WAE Live 6.4.x server, sqlhosts.ml has the following:

```
ml_remote onsoctcp 172.131.130.112 9089
mltcp     onsoctcp 127.0.0.1    9088
```

```
ml        onipcshm 127.0.0.1    dummy
```

On the WAE Live 7.1.x server, change the sqlhosts.ml file to the following:

```
ml_remote onsoctcp <Live71_mld_IP_address> <Live71_port>
mltcp     onsoctcp 127.0.0.1    9088
ml        onipcshm 127.0.0.1    dummy
```

Step 6 Confirm that the data store directory (attribute in config.xml) is the same. If there are missing files on the WAE 7.1.x server, then create zero size files with the same name using the touch command and change the file permission to 600 for the newly created files.

Example:

On the WAE Live 6.4.10 server:

```
[cariden@wodl114 archives]$ cd $CARIDEN_ROOT/software/mld/current/data/

[cariden@wodl114 data]$ ls -la

total 63591328

drwxr-xr-x  2 cariden caridenstaff       4096 Dec  5 12:55 .

drwxr-xr-x 10 cariden caridenstaff       4096 Dec  5 12:35 ..

-rw-------  1 cariden caridenstaff 2147483648 Dec  5 13:01 catdbs001

. . . . . .

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 14:31 sbspace000

-rw-------  1 cariden caridenstaff 2147483648 Dec  5 14:54 tempdbs000

-rw-------  1 cariden caridenstaff 2147483648 Dec  5 14:54 tempdbs001

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 14:54 tsdbs000

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 14:54 tsdbs001

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 14:31 tsdbs002000

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 14:31 tsdbs002001
```

Example:

On the WAE Live 7.1.x server, the missing files are `tsdbs002000` and `tsdbs002001`:

```
$CARIDEN_ROOT/software/mld/current/data

[cariden@wodl113 data]$ ls -la

total 46814024

drwxr-xr-x  2 cariden caridenstaff       4096 Dec  5 01:56 .

drwxr-x--- 10 cariden caridenstaff        143 Dec  5 01:44 ..

-rw-------  1 cariden caridenstaff 2147483648 Dec  5 22:52 catdbs001
. . . . . .

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 22:50 sbspace000
```

```
-rw-------  1 cariden caridenstaff 2147483648 Dec  5 22:52 tempdbs000

-rw-------  1 cariden caridenstaff 2147483648 Dec  5 22:52 tempdbs001

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 22:52 tsdbs000

-rw-------  1 cariden caridenstaff 4294967296 Dec  5 22:52 tsdbs001
```

You would then create zero size files so that WAE Live 7.1.x has the same contents as WAE Live 6.4.10.

```
touch tsdbs002000
touch tsdbs002001
chmod go-r tsdbs002000
chmod go-r tsdbs002001
```

Step 7 Copy the WAE Live 6.4.x data store backup file to the WAE Live 7.1.x server.

Example:

On WAE Live 6.4.x

```
scp $CARIDEN_ROOT/software/mld/current/backups/fullbackups/wodl114_1_L0
user@live71:$CARIDEN_ROOT/software/mld/current/backups/fullbackups/
```

Step 8 Rename the copied WAE 6.4.x data store backup file to the WAE 7.1.x backup file.

Example:

If WAE Live 7.1.x backup file is named `wodl113_1_L0`, then

```
mv $CARIDEN_ROOT/software/mld/current/backups/fullbackups/wodl114_1_L0
$CARIDEN_ROOT/software/mld/current/backups/fullbackups/wodl113_1_L0
```

Step 9 Stop WAE 7.1.x mld and restore the data store backup.

```
mld -action stop
ml_restore -directory $CARIDEN_ROOT/software/mld/current/backups/fullbackups
```

Step 10 Run a sanity check. This process may take awhile.

```
mld -sanity all
```

Step 11 Restart mld.

```
mld -action restart
```

Step 12 From the WAE Live 6.4.x server, do the following:

a) Check the config.xml file to see if the following attributes are set to specific directories. If not, it is specified under the `MLData` attribute and the default path is `$CARIDEN_ROOT/data/mldata/`:

- `AppData`
- `Backup`
- `Map.ArchivePath`
- `ReportData`

If these attributes are set, copy the data from the respective directory to the same directory in the WAE Live 7.1.x server.

b) If the preceding attributes are not set, then use the tar command to pack all the respective directories and copy `mldata.tar` to the same directory on the WAE Live 7.1.x server.

Example:

```
tar -cvf mldata.tar appdata/ archives/ customdata/ jobs/ plans/ reports/
```

Copy `mldata.tar` to the WAE Live 7.1.x server:

```
scp mldata.tar cariden@wodl113:$CARIDEN_ROOT/data/mldata/
```

On the WAE Live 7.1.x server, navigate to where the MLdata property values are located. For example:

```
cd $CARIDEN_ROOT/data/mldata/
tar -xvf mldata.tar
```

Step 13 From the WAE Live 6.4.x server, copy the following contents from `$CARIDEN_ROOT/etc` to the same path in WAE Live 7.1.x:

a) `$CARIDEN_ROOT/etc/config/config.xml`

b) `$CARIDEN_ROOT/etc/matelive`

c) `$CARIDEN_ROOT/etc/user_manager`

Example:

```
# tar -cvf etc.tar config/ matelive/ user_manager/
config/
config/config.xml
config/config.xml.bak
matelive/
user_manager/
user_manager/auth.db.properties
user_manager/auth.db.script

# scp etc.tar <WAE_64x_host>:$CARIDEN_ROOT/etc/.
```

From WAE Live 7.1.x server, copy the $CARIDEN_ROOT/etc directory.

```
# cd $CARIDEN_ROOT/etc
# tar -xvf etc.tar
```

Step 14 Start the WAE Live 7.1.x web server and data store:

```
wae-live-start
```

Cisco WAE Live Data Store

This section describes how to install, upgrade, back up, and restore a Cisco WAE Live data store. It also describes how to purge data using the `ml_purge` tool.

If the defaults were used during installation, `$CARIDEN_HOME` is the same as `/opt/cariden/software/live/current`.

Install WAE Live Data Store

The following procedure describes how to install the Cisco WAE Live Data Store using mld_tool. The `mld` tool installs both the mld server and an empty data store directory.

Before you begin

- For better performance, create a separate ext2 partition for the directory that is specified with the `-datastore` option.
- Understand what type of production environment you want to create.

Note

- After the data store is created, it is difficult to modify any of the installation options (including the user name).
- A 'demo' data store is just for pilot purpose. If you start with 'demo', then you must recreate the data store when it is time to move to production and the data in 'demo' data store will be lost.

- The `-demo` or `-storage`, `-cpu` and `-memory` options are required. For more information on the mld command and options, see mld Options, on page 37.

Procedure

Step 1 If WAE Live is running, stop the web server:

```
# wae-live-stop
```

Step 2 Enter appropriate mld command to install the data store. For mld commands and options see mld Options, on page 37.

Example 1: To be prompted through installation and obtain sizing recommendations:

```
# mld -installchk
```

Example 2: To install mld with a demo data store size:

```
# mld -action install -demo true
```

Example 3: To install a small mld server into $CARIDEN_ROOT/data/matelive, reserve 2 CPUs, reserve 542 GB of disk storage and allocate 2.2 GB (2200 MB) of memory:

```
# mld -action install -mldata $CARIDEN_ROOT/data/matelive -cpus 2 -storage 1:1:540 -memory 200:55:2000
```

Step 3 Start mld and the web server:

```
# wae-live-start
```

mld Options

Option	Description	Default
`-version`	Displays the data store version.	Default installation directory `$CARIDEN_ROOT/software/mld/current`
`-action`	`install` —Installs a new mld server and data store, and start the mld server. `upgrade` —Updates an existing mld server and start the mld server. `start` —Alternative way to start the mld server. `stop` —Alternative way to stop the mld server. `status` —Alternative way to show the status of the mld server. `restart` —Alternative way to stop and then restart the mld server.	
`-installchk`	Prompts you through installation and gives sizing recommendations.	
Use only with -action install (If an option is not given, the installation performs the same tasks as `-installchk` .)		
`-demo true`	Installs a demo data store. **Note** If both `-demo` and `-storage` options are used, `-demo` takes precedence.	
`-storage <n:n:n>`	Allocates the disk and memory based on the anticipated data store size, where <n:n:n> is data:indices:timeseries in GB. For details and recommended values, use `-installchk` and `-verbose` options. **Note** If the data store is larger than the demo size, this option is required when using `-action install`.	
`-memory <n:n:n>`	Allocates the requested memory of the data store, where <n:n:n> is data:indices:timeseries in MB. For details and recommended values, use `-installchk` and `-verbose` options.	
`-mldata <directory>`	Specifies directory where all application data is stored. This directory includes the data store, report output, and other application data.	`$CARIDEN_ROOT/data/mldata`

Option	Description	Default
`-datastore <directory>`	Specifies directory where the data store is initialized. Once set, this directory cannot be changed. You can, however, use symbolic links.	`$CARIDEN_ROOT/data/mldata/datastore`
`-cpus <#>`	Reserves the number of CPUs for the data store and the mld server.	Half of the total CPUs
Use only with -action install or -action upgrade		
`-mld <directory>`	Specifies directory where the mld server is installed. Once set, this directory cannot be changed. You can, however, use symbolic links.	`$CARIDEN_ROOT/software/mld/current`
`-backup <directory>`	Specifies directory for saving data store backups.	`$CARIDEN_ROOT/data/mldata/backup`

Back Up the Data Store

Cisco WAE Live backs up the time-series derived data from plan files. It does not back up transaction logs or other WAE Live data, such as application data and report data.

The required amount of space for backups depend on the installation size and how long a system has been running.

Best Practices

- Perform the backup to a different disk drive, or copy the backup to a different physical device after you finish the backup.
- Perform backups outside of peak traffic hours.
- Set up a backup directory that is on a different physical disk when you first install the mld server and data store. Doing so sets the default backup directory for all backups.

```
mld -action install -backup <backup_directory>
```

- The backup process makes a copy of the data store, but it does not back up other Cisco WAE Live data, such as application data and report data. Therefore, with some regularity, copy this other data to a safe location, such as to a different physical disk.
- Perform a full backup at least weekly or monthly, with numerous incremental backups in between them.
- Rather than running manual backups, call `ml_backup` from a cron job.
- Perform only 1 backup at a time so that their schedules do not overlap. Running simultaneous backups are not supported. Ensure that there is at least 1 hour between each backup. After it completes, verify that the backup was completed within the hour.

Backup Steps

Caution If you delete the previous mld installation directory, you may delete all the data. To check the current location, enter the following command: `mld -diag -c | egrep ROOTPATH`

The `ml_backup` tool enables you to perform multiple levels of backups to save disk space. An OS file system backup cannot be used to restore the data store. Use the `ml_backup` tool to perform a complete backup to use for data store restoration.

You can execute `ml_backup` to run a manual backup at any time. However, the first time you use backup levels, you must perform backups in this sequence.

Note Keep both the data store and the web server running.

Sequence	Enter	Description

1	`ml_backup`
	or
	`ml_backup -L 0`
		Level 0—Back up everything.

2	`ml_backup -L 1`	Level 1—Back up everything since the most recent Level 0 backup was performed.
3	`ml_backup -L 2`	Level 2—Back up everything since the most recent Level 1 backup was performed.

Note For larger systems that frequently run plan file processes, less incremental Level 1 and Level 2 backups are available in between Level 0 backups. The following error appears when a Level 1 or Level 2 backup is not available:

```
Archive failed - The existing level-0 backup for DBspace rootdbs is too old to allow any
      incremental backup.
```

When this error appears, run a Level 0 backup.

To run a backup using all defaults, enter `ml_backup`. The tool uses the default backup directory, and creates a full backup.

- To override the default backup directory, use the `-directory` option. The default backup directory is `$CARIDEN_ROOT/data/mldata/backup`.
- To set a different backup level, use the `-L` option.

The following example sets the backup directory to `$CARIDEN_ROOT/data/waelive/backups` and backs up only data that is new since the last level 0 backup was run. This assumes that you ran `ml_backup` one time using the default level of 0.

```
ml_backup -directory /data/waelive/backups -L 1
```

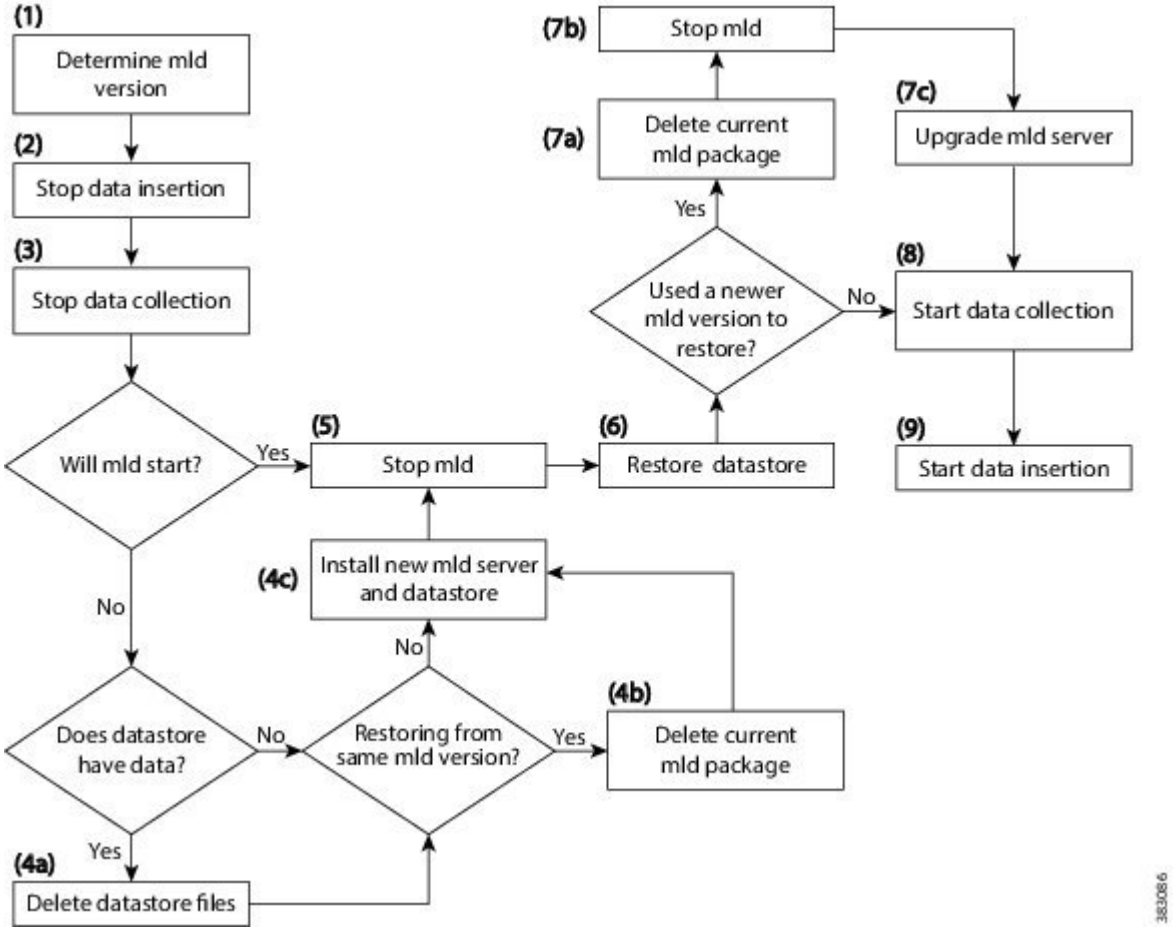
Restore the Data Store

Note the following before attempting to restore the data store:

- To restore a data store, you must have a backup of it. For information about backing up the data store, see the Backing Up the Data Store section.
- Ensure you have a proper disk and disk space. For example, if your data was corrupted, you would need a new disk. If the restoration is due to a space issue, then add more space to the existing disk.
- If you have a single device configuration, the collection of data will be interrupted during the restoration of a WAE Live data store.
- If the backup data store resides on a different device, confirm that the following prerequisites are met:
  - The username and user ID (uid) of both devices must be the same.
  - The backup data store name uses a hostname as a portion of its name. This hostname portion of the backup data store name must be changed to be the same as the hostname on the device to which it is being restored.

    Example: The backup data store name is akdobi.acme.com_1_L0. The hostname on the device on which the data store is being restored is akgudei.acme.com. In this case, change the backup data store name to akgudei.acme.com_1_L0.

Figure 1: Data Store Restoration Workflow



Delete Data from the Data Store

The `ml_purge` tool removes all data prior to the specified timestamp.

Before you begin

Before running `ml_purge`, confirm that there are no insertions running (`ml_insert_ctl -status`). Insertions might fail due to locks created by `ml_purge` while it is in operation. You might need to pause the scheduler to prevent scheduled insertions (`ml_insert_ctl -disable-scheduler`).

Procedure

To run `ml_purge`:

```
# ml_purge <timestamp>
```

where <*timestamp*> is in the following UTC format: *year-month-day* `T` *hour:minutes*. For example:

```
# ml_purge 2017-01-31T00:00
```

Install the Cisco WAE Live License

Cisco WAE Live supports Cisco Smart Licensing and traditional licensing. Use one of the following procedures based on the license type.

Install Traditional License

Before you begin

Confirm you have the Cisco WAE Live license on your server.

Procedure

Step 1 If Cisco WAE Live is not running, start it.

```
# embedded_web_server -action start
```

Step 2 Start the Cisco WAE Live UI in a supported browser: `https://<server_IP>:8443`. The default username is "admin" and the password is "admin".

Step 3 Choose **System** > **Licenses**.

Step 4 Click **Upload Traditional License**.

Step 5 Click **Select Licenses**.

Step 6 Browse to the location or enter the name of the license file (.lic extension), and click **Open**.

Step 7 Click **Upload License**.

Step 8 Restart Cisco WAE Live.

```
# embedded_web_server -action restart
```

Install Smart License

Before you begin

Confirm you have the Cisco smart license on WAE server.

Procedure

Step 1 If Cisco WAE Live is not running, start it.

```
# embedded_web_server -action start
```

Step 2 Start the Cisco WAE Live UI in a supported browser: `https://<server_IP>:8443`. The default username is "admin" and the password is "admin".

Step 3 Choose **System** > **Licenses**.

Step 4 Click **Enable Smart Software Licensing**.

Step 5 Provide WAE Server Name or IP address, Port, Username and Password, and click **Enable**.

Step 6 Restart Cisco WAE Live.

```
# embedded_web_server -action restart
```

What to do next

You can begin using Cisco WAE Live and collect plan files. To collect plan files from Cisco WAE 7.6.x, go to **Settings** > **Data Source** and click the 7.6.x Remote Archive option. Enter the appropriate Cisco WAE 7.6.x network and server details. For information about using Cisco WAE Live, see the Cisco WAE Live User Guide.

Troubleshoot Authentication Failure Error

If the admin password on the WAE collector is changed, the UI of WAE Live does not show the Live components since the license check fails due to password change. To resolve this issue, follow these steps:

Procedure

Step 1 Change the admin password on the WAE server (new password).

Step 2 On the WAE Live server, run the `license_install` tool with the new password.

For example:

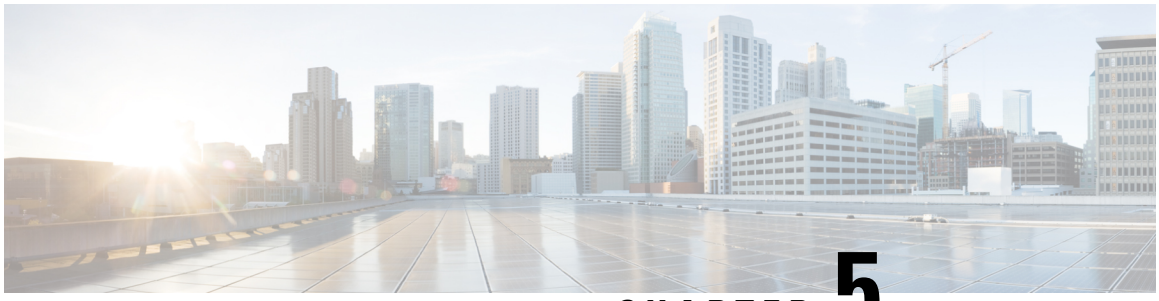
```
license_install -smart-lic-host <WAE-server-IP> -smart-lic-port 2022 -smart-lic-username admin
-smart-lic-password <new-password>
```

Step 3 On the WAE Live server, go to `$HOME/.cariden/etc` and move the **MATE_Smart.lic** file to `$CARIDEN_ROOT/etc`.

Step 4 Run the `license_check` command on the WAE Live server.

Note that the authentication error message is no longer displayed.

Step 5 From the WAE Live UI, go to **Settings > Data Source** and update the password.



CHAPTER 5

Security

Core Security Concepts

If you are an administrator and are looking to optimize the security of your product, you should have a good understanding of the following security concepts.

HTTPS

Hypertext Transfer Protocol Secure (HTTPS) uses Secure Sockets Layer (SSL) or its subsequent standardization, Transport Layer Security (TLS), to encrypt the data transmitted over a channel. Several vulnerabilities have been found in SSL, so now supports TLS only.

Note TLS is loosely referred to as SSL often, so we will also follow this convention.

SSL employs a mix of privacy, authentication, and data integrity to secure the transmission of data between a client and a server. To enable these security mechanisms, SSL relies upon certificates, private-public key exchange pairs, and Diffie-Hellman key agreement parameters.

SSL Certificates

SSL certificates and private-public key pairs are a form of digital identification for user authentication and the verification of a communication partner's identity. Certificate Authorities (CAs), such as VeriSign and Thawte, issue certificates to identify an entity (either a server or a client). A client or server certificate includes the name of the issuing authority and digital signature, the serial number, the name of the client or server that the certificate was issued for, the public key, and the certificate's expiration date. A CA uses one or more signing certificates to create SSL certificates. Each signing certificate has a matching private key that is used to create the CA signature. The CA makes signed certificates (with the public key embedded) readily available, enabling anyone to use them to verify that an SSL certificate was actually signed by a specific CA.

In general, setting up certificates involve the following steps:

1. Generating an identity certificate for a server.

2. Installing the identity certificate on the server.
3. Installing the corresponding root certificate on your client or browser.

The specific tasks you need to complete will vary, depending on your environment.

1-Way SSL Authentication

This authentication method is used when a client needs assurance that it is connecting to the right server (and not an intermediary server), making it suitable for public resources like online banking websites. Authentication begins when a client requests access to a resource on a server. The server on which the resource resides then sends its server certificate (also known as an SSL certificate) to the client in order to verify its identity. The client then verifies the server certificate against another trusted object: a server root certificate, which must be installed on the client or browser. After the server has been verified, an encrypted (and therefore secure) communication channel is established. At this point, the server prompts for the entry of a valid username and password in an HTML form. Entering user credentials after an SSL connection is established protects them from being intercepted by an unauthorized party. Finally, after the username and password have been accepted, access is granted to the resource residing on the server.

Note A client might need to store multiple server certificates to enable interaction with multiple servers.

To determine whether you need to install a root certificate on your client, look for a lock icon in your browser's URL field. If you see this icon, this generally indicates that the necessary root certificate has already been installed. This is usually the case for server certificates signed by one of the bigger Certifying Authorities (CAs), because root certificates from these CAs are included with popular browsers.

If your client does not recognize the CA that signed a server certificate, it will indicate that the connection is not secure. This is not necessarily a bad thing. It just indicates that the identity of the server you want to connect has not been verified. At this point, you can do one of two things: First, you can install the necessary root certificate on your client or browser. A lock icon in your browser's URL field will indicate the certificate was installed successfully. And second, you can install a self-signed certificate on your client. Unlike a root certificate, which is signed by a trusted CA, a self-signed certificate is signed by the person or entity that created it. While you can use a self-signed certificate to create an encrypted channel, understand that it carries an inherent amount of risk because the identity of the server you are connected with has not been verified.

Install Certificates

This section contains information about installing security certificates on the Cisco WAE server, Cisco WAE Coordinated Maintenance, and Cisco WAE Live.

Install a Certificate for the Cisco WAE Server

Cisco WAE comes with a default certificate. Because this certificate is not from a “trusted CA", the browser shows an unsecured connection warning. This is the expected behavior. The warning can be removed by applying an appropriate Certificate Authority (CA) issued certificate.

Procedure

Step 1 Create a private server key and store it in a secure location. For example:

```
# openssl genrsa -out server.key 2048
```

Step 2 Create the Certificate Signing Request (CSR). The CSR is used by CA to create a certificate that identifies your website as secure. For example:

```
# openssl req -sha256 -new -key server.key -out server.csr
```

Step 3 Submit the CSR to the Certificate Authority to obtain your Certificate (for example, server.crt).

Note
WAE supports server.crt in the PEM format only. To convert the server certificate from DER to PEM format, you can use the command

```
sudo openssl x509 -inform der -in <input certificate filename> -out <output certificate filename>
```

Step 4 Modify the `<WAE_run_directory>/wae.conf` by changing `<key-file/>` and `<cert-file/>` elements to point to the location of the server.key and server.crt files.

Step 5 Restart the Cisco WAE server.

```
# sudo supervisorctl stop wae:*
# sudo supervisorctl start wae:*
```

Install a Certificate for Cisco WAE Live

Cisco WAE Live includes a default certificate that causes the browser to indicate that the certificate is not trusted. This is the expected behavior. The warning can be removed by applying an appropriate CA issued certificate.

To install a CA certificate for Cisco WAE Live, do the following:

Before you begin

Note This procedure is only applicable for Cisco WAE Live 7.1.1 and later.

- You must be an administrator with Cisco WAE user privileges to perform this task.
- The tool `keytool` is deployed with jdk/jre. Make sure the `keytool path` is included in `PATH`.

Note The previous example is applicable if your shell is sh, ksh, or bash. Use equivalent commands for other shells.

- Log out and in again, or enter the following command using the appropriate profile filename.

```
# source ~/.profile
```

Procedure

Step 1 In order to obtain a certificate from the Certificate Authority (CA) of your choice, you have to create a Certificate Signing Request (CSR). To create a CSR follow these steps:

a) Delete the default certificate. For example:

```
# keytool -storepass changeit -delete -alias cisco -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

b) Create a local self-signed Certificate. For example:

```
# keytool -storepass changeit -genkey -alias tomcat -keyalg RSA -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

c) Create the CSR. For example:

```
# keytool -storepass changeit -certreq -keyalg RSA -alias tomcat -file certreq.csr -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore
```

d) Submit the CSR to a Certificate Authority to obtain your certificate.

e) (Optional) Restart Cisco WAE Live to use the new certificate immediately.

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

Step 2 Install the certificate.

a) Download a Chain Certificate (also called a Root Certificate) from the CA you obtained the certificate from.

b) Import the Chain Certificate into the keystore.

```
# keytool -storepass changeit -import -alias root -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -trustcacerts -file
<filename_of_the_chain_certificate>
```

c) Import the new certificate.

```
# keytool -storepass changeit -import -alias tomcat -keystore
$CARIDEN_HOME/lib/web/apache-tomcat-8.5.53/conf/keystore -file <your_certificate_filename>
```

d) Restart Cisco WAE Live.

```
# embedded_web_server -action stop
# embedded_web_server -action start
```

Install a Certificate for the LDAP Server

Cisco WAE supports authentication and authorization of foreign users using Lightweight Directory Access Protocol (LDAP).

To use LDAPS protocol, get the SSL certificate and add it to a keystore.

Procedure

Step 1 Save the self signed certificate to cert.pem file using the following command:

```
# openssl s_client -connect <ldap-host>:<ldap-ssl-port> </dev/null 2>/dev/null | sed -n
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

Step 2 Get the default key-store path by running the following command from `WAE_RUN` directory.

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <ldap-host> <ldap-ssl-port> 2>1 | grep "trustStore is:"
```

Running the above command helps you find the directory from where certs are picked up. It may be a directory similar to:

```
trustStore is: /usr/lib/jvm/java-1.8.0-openjdk-1.8.0.102-4.b14.el7.x86_64/jre/lib/security/cacerts
```

Step 3 Import cert into default key-store using following command:

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```

Install a Certificate for the EPN-M Server

Install the certificate when using the Cisco Evolved Programmable Network Manager (Cisco EPN Manager) agent for L1 collection.

Procedure

Step 1 Save the self signed certificate to cert.pem file using the following command:

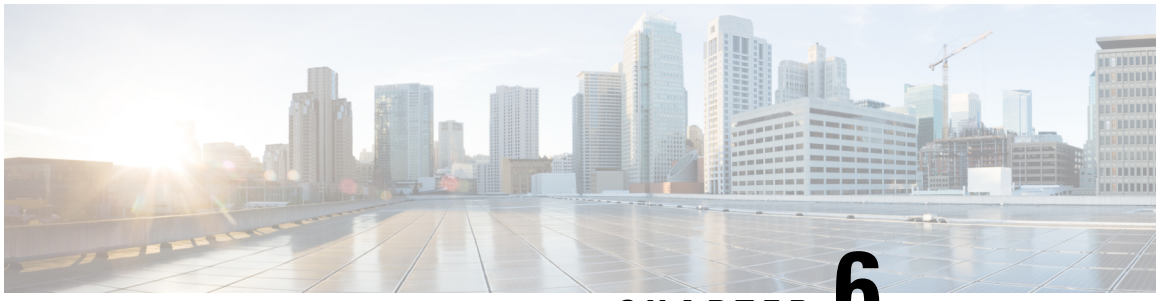
```
# openssl s_client -connect <epnm-host>:<epnm-port> </dev/null 2>/dev/null | sed -n
'/^-----BEGIN/,/^-----END/ { p }' > cert.pem
```

Step 2 Get the default key-store path using the following command. Typically the default key-store path is /etc/pki/java/cacerts.

```
# $WAE_ROOT/lib/exec/test-java-ssl-conn <epnm-host> <epnm-port> 2>1 | grep "trustStore is:"
```

Step 3 Import cert into default key-store using following command:

```
# sudo keytool -import -keystore <default-key-store-path> -storepass changeit -noprompt -file cert.pem
```



CHAPTER 6

Next Steps

The following topics describe the next steps you perform to get started with Cisco WAE. You access the WAE UI, WAE Expert Mode, or WAE CLI to perform operations. For detailed information, see the *Cisco WAE User Guide*.

Log In to Cisco WAE

This section describes how to log in to the available Cisco WAE interfaces: Cisco WAE UI, Expert Mode, and the Cisco WAE CLI. For more information about these interfaces, see the Cisco WAE User Guide.

Log In to the Cisco WAE UI

Follow these steps to log in to the Cisco WAE web UI.

Before you begin

Confirm that all the appropriate services are running. All services automatically start after installation. For information about how to start or stop Cisco WAE, see Start and Stop Cisco WAE, on page 24.

Procedure

Step 1 Start one of the supported browsers. See Cisco WAE Installation Requirements, on page 3.

Step 2 In the browser's address bar, enter `https://`*server-ip*`:8443`, where *server-ip* is the IP address of the server where Cisco WAE installed.

The Cisco WAE user interface displays the **Login** window.

Step 3 Enter the web UI username and password.

Step 4 Click **Login**.

The home page appears and you can now use the web UI.

What to do next

After you log in to Cisco WAE, you can start a network topology collection to create a network model. For information about creating a network model, see the Cisco WAE User Guide.

Log In to the Expert Mode

You must log in to the WAE UI before accessing the Expert Mode.

Before you begin

Confirm that all the appropriate services are running. All services automatically start after installation. For information about how to start or stop Cisco WAE, see Start and Stop Cisco WAE, on page 24.

Procedure

Step 1 Start one of the supported browsers. See Cisco WAE Installation Requirements, on page 3.

Step 2 In the browser's address bar, enter **`https://`***server-ip*`:8443`**, where *server-ip* is the IP address of the server where Cisco WAE is installed.

The Cisco WAE UI displays the **Login** window.

Step 3 Enter the Cisco WAE UI username and password.

Step 4 Click **Login**.

The home page appears and you can now use the web UI.

Step 5 In the top-right corner of the Cisco WAE UI, click the tool icon to access the Expert Mode.

What to do next

After you log in to Cisco WAE, you can start a network topology collection to create a network model. See the Cisco WAE User Guide.

Log In to the WAE CLI

To log in to the WAE CLI:

Procedure

Step 1 Navigate to the WAE run-time directory and enter `wae_cli`.

```
# wae_cli -u admin
admin@wae#
```

Note
You can enter `wae_cli --help` to view all the WAE CLI options.

Step 2 (Optional) To enable configuration operations, switch to the configuration mode.

```
admin@wae# config
admin@wae%#
```

Example

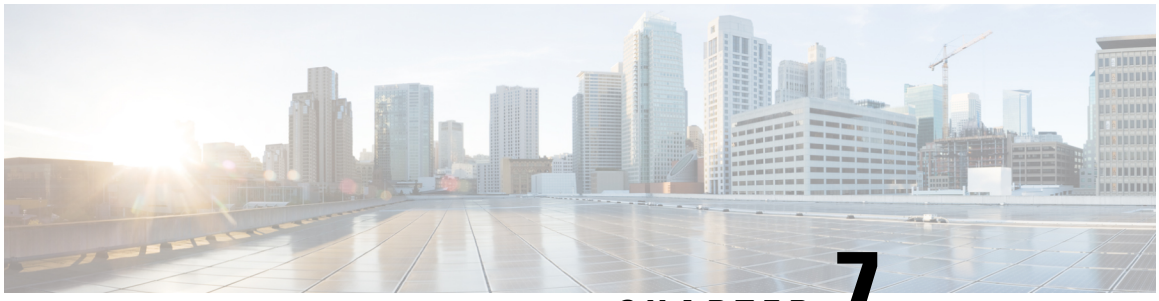
For example:

```
waerun# wae_cli -u admin
admin@wae# config
admin@wae%#
```

Build a Network Model

This topic gives a high-level description of tasks that are necessary to build a network model. For more detailed information, see the Cisco WAE User Guide.

1. Configure device authgroups, SNMP groups, and network profile access.
2. (Optional) Configure agents. This step is required only for collecting SR-PCE, LAG and port interface, multilayer, netflow, or telemetry information.
3. Configure an aggregated network and sources with a topology NIMO.
4. Configure additional collections such as demands, traffic, layout, inventory, and so on.
5. Schedule when to run collections.
6. Configure the archive file system location and interval at which plan files are periodically stored.
7. (Optional) View plan files in Cisco WAE applications.



CHAPTER 7

Uninstall Cisco WAE

Uninstall Cisco WAE

This procedure describes how to remove a Cisco WAE installation.

Note You can have more than one instance of Cisco WAE installed. When going through the uninstallation procedure, make sure you are removing the correct Cisco WAE installation and run-time directories.

Procedure

Step 1 To stop all WAE process, use the command:

```
# sudo supervisorctl stop wae:*
```

Step 2 Navigate to the parent directory and remove the Cisco WAE installation and run-time directories.

```
# rm -rf <wae_installation_directory>
# rm -rf <wae_run_time_directory>
```

Example

For example:

```
# sudo supervisorctl stop wae:*
# cd
# rm -rf waeinstall
# rm -rf waerun
```

Uninstall Cisco WAE