Oracle Utilities Live Energy Connect

Installation Guide

Release 25.12.0.0.0
G49424-03
January 2026

ORACLE

Oracle Utilities Live Energy Connect Installation Guide, Release 25.12.0.0.0

G49424-03

Contents

1

Live Energy Connect Installation

The Oracle Utilities Live Energy Connect Installation Guide helps you understand the components and procedures necessary to install Live Energy Connect25.12.0.0.0.

Welcome to the Oracle Utilities Live Energy Connect (LEC) 25.12.0.0.0 Installation Guide. Use the information in this guide to understand the components and procedures for installing LEC.

Getting Started

Oracle Utilities Live Energy Connect 25.12.0.0.0 is a microservices architecture built using cloud-native tools and patterns and delivered as an on-premises solution. LEC 25.12.0.0.0 services run as a clustered application inside an Oracle Cloud Native Environment (OCNE) cluster. OCNE is a fully integrated suite for the development and management of cloud-native applications. The LEC 25.12.0.0.0 installer installs and deploys 25.12.0.0.0 resources and services on a running OCNE 2.x cluster. The LEC 25.12.0.0.0 installer currently supports a single-node OCNE cluster topology only.

> **Note**
>
> The term "node" in this document refers to a host being used in a Kubernetes cluster as a Kubernetes node. For more information about OCNE and Kubernetes terminology, refer to the OCNE documentation.

Oracle Utilities Live Energy Connect 25.12.0.0.0 must be installed on a system that is running an OCNE 2.x cluster. The Oracle Utilities Live Energy Connect 25.12.0.0.0 installer is distributed as a self-extracting TAR archive that will install product components (for example, configuration files, container images, Helm charts, and so on) to the running OCNE 2.x cluster. The TAR archive is installed from a machine that has network access to the running OCNE 2.x cluster (i.e., one that is running the OCNE 2.x CLI). If running a **libvirt** cluster, this will be the same host machine that the cluster is running on. For ***Bring Your Own*** model clusters, this will be a remote system with **SSH** access to the cluster.

2

Installation Prerequisites

Introduction to the requirements and setups needed prior to installing Oracle Utilities Live Energy Connect.

The sections in this chapter outline prerequisites that must be satisfied before installing Oracle Utilities Live Energy Connect (LEC) 25.12.0.0.0.

Hardware Requirements

Provides the Oracle Utilities Live Energy Connect host machine hardware requirements for use in an OCNE cluster.

The following are the minimum recommended hardware requirements for each host machine to be used in the OCNE cluster:

- **CPU**: 8 CPU
- **Memory**: 16 GB RAM
- **Hard Disk Space:**
  - At least 40 GB available the `/var` directory.
  - At least 256 GB available in the `/opt` directory.
- **Architecture:** x86-64
- **Network Interface:** 1 GB Ethernet NIC
- **File System:** The root file system should be XFS, which is the default Oracle Linux file system.

Creating the Oracle Cloud Native Environment Cluster

The Oracle Utilities Live Energy Connect installation in an OCNE cluster requirements.

Oracle Utilities Live Energy Connect 25.12.0.0.0 must be deployed to a cluster created using the Oracle Cloud Native Environment 2.x CLI.

> **Note**
>
> LEC 25.12.0.0.0 currently only supports deployment to a single-node cluster. See the Oracle Cloud Native Environment 2.x documentation on Oracle Help Center for details on setting up and deploying to OCNE.

OCNE Software Packages Dependencies

Oracle Utilities Live Energy Connect 25.12.0.0.0 requires the OCNE cluster to have preinstalled and configured software packages (`nginx`, `helm`, `yq`, `ocne and kubectl`),`curl`, and `jq`.

The following software packages must be present on each cluster host machine prior to installing Oracle Utilities Live Energy Connect 25.12.0.0.0:

- **nginx:** To install `nginx`, run the following shell command:

  ```
  sudo dnf -y install nginx
  sudo dnf -y install nginx-mod-stream
  sudo setsebool -P httpd_read_user_content 1
  sudo systemctl enable nginx
  sudo systemctl start nginx
  sudo mkdir -p /etc/nginx/conf.d/stream
  ```

- **helm:** To install `helm`, run the following shell commands:

  ```
  sudo dnf config-manager --enable ol9_olcne19
  sudosd
  ```

- **yq:** To install `yq`, run the following shell command (see https://github.com/mikefarah/yq):

  ```
  sudo wget
  https://github.com/mikefarah/yq/releases/latest/download/yq_linux_amd64
  -O /usr/bin/yq && sudo chmod +x /usr/bin/yq
  ```

- **ocne and kubectl:** These utilities should have been installed during the cluster install. For kubectl to run properly, add "`export KUBECONFIG=$(once show)`" to `~/.bashrc`, and source that file. But, if not, see the ocne documentation to install it. And, if kubectl is not installed, use the following:

  ```
  sudo dnf -y install kubectl
  ```

- **curl:**This should be installed along with Oracle Linux, but verify.
- **jq:** This should be installed along with Oracle Linux, but verify.

3
Installing Oracle Utilities Live Energy Connect

This chapter describes the installation and configuration steps for an LEC implementation.

The following sections describe how to install and configure Oracle Utilities Live Energy Connect 25.12.0.0.0.

Downloading and Running the Oracle Utilities Live Energy Connect Installer

The Oracle Utilities Live Energy Connect installer, which is distributed as a self-extracting archive, will install all needed Oracle Utilities Live Energy Connect 25.12.0.0.0 components to Oracle Cloud Native Environment 2.x.

To install Oracle Utilities Live Energy Connect 25.12.0.0.0:

1. Sign into My Oracle Support.
2. Click the **Patches & Updates** tab.
3. Find the **Patch Search** section, click the **Search** tab, and then select **Product or Family (Advanced)** from the left column.
4. In the **Product** field, enter

   ```
   Oracle Utilities Live Energy
               Connect
   ```

   .
5. In the **Release** drop-down list, select **Oracle Utilities Live Energy Connect**, and then click **Search**.
6. Check the **Updated** column to find the latest release, and then select the patch.
7. Click **Download** and then click the **TAR** link to download the archive; the archive will have a name of the form `lec_gz.run`.
8. Save the archive to a machine that has network access to the running OCNE 2.x cluster. If running a `libvirt` cluster, this will be the host machine - for *Bring Your Own* model clusters, this will be a remote box with SSH access to the cluster.
9. Move the archive to a directory from which you wish to run the installation. The directory should be owned by the user that will run the installer.
10. From a shell prompt, **cd** to the directory containing the archive, and enter `./<ARCHIVE>` to run the install (where `<ARCHIVE>` is the name of the archive, for example, `lec_gz.run`). The installer will detect the running OCNE 2.x cluster, and properly deploy LEC artifacts.

> **Note**
>
> You may need to adjust the permissions on the `gz.run` file to run it if you encounter any errors. To do this, run the following command:
>
> ```
> chown ${USER}: lec_.gz.run && chmod 700 lec_.gz.run
> ```

Troubleshooting Installation Problems

When there are problems with the installation process, Oracle Utilities Live Energy Connect 25.12.0.0.0 provides steps to determine what the problem is and remedy it.

The following is a list of troubleshooting steps you can use if you encounter certain problems during the installation process:

Problem: Uncompressing Oracle Utilities Live Energy Connect ... Extraction failed

When the self-extracting archive is run, it will use the tar program to untar the contents of the installer. If it cannot call the tar command it will print the following error:

Solution: The tar program can be installed with the following shell command:

```
sudo dnf install tar
```

Verifying the Oracle Utilities Live Energy Connect Installation

The Oracle Utilities Live Energy Connect 25.12.0.0.0 installation can be verified using the kubectl client CLI tool with the defined commands.

The Oracle Utilities Live Energy Connect 25.12.0.0.0 installation can be verified using the kubectl client CLI tool. To do this, run the following steps:

1. Reload the user's Bash environment variables by exiting the interactive shell and then starting a new one.

   Alternatively, you could run the command:

   ```
   source ~/.bashrc
   ```

2. At the command prompt, check the status of all the pods in the "lec" namespace of the Kubernetes cluster by running the command:

   ```
   kubectl get pods -n lec
   ```

   This command should return a list of the LEC-specific pods in the cluster with information about their running status. If the installation was successful the status should say `Running`. The following is an example output:

   ```
   [lec@node ~]$ kubectl get pods -n
                 lec
   ```

```
NAME READY STATUS RESTARTS AGE

fluentd-fld-fluentd-2g5qr 1/1 Running 1
          4m


fluentd-fld-fluentd-knzfb 1/1 Running 1
          4m


fluentd-fld-fluentd-m4fm7 1/1 Running 1
          4m


kafka-0 1/1 Running 4 (58m ago) 3m

kafka-zookeeper-0 1/1 Running 1 3m

opensearch-cluster-master-0 1/1 Running 1
            5m


os-dashboards-lec-747cdcd4cf-8874d 1/1 Running 1
            5m


[lec@node ~]$
```

3. For information about a specific pod use the ‘kubectl describe pod’ command:

```
[lec@node ~]$ kubectl describe pod
          kafka-0


Name: kafka-0

Namespace: lec

Priority: 0

Service Account: kafka

Node: node1/192.168.122.101

Start Time: Wed, 27 Mar 2024 04:45:24
            +0000


Labels:
          app.kubernetes.io/component=kafka

app.kubernetes.io/instance=kafka

app.kubernetes.io/managed-by=Helm

app.kubernetes.io/name=kafka

controller-revision-hash=kafka-57445667b

helm.sh/chart=kafka-20.0.2
```

```
statefulset.kubernetes.io/pod-name=kafka-0
Annotations: <none>
Status: Running
IP: 10.244.0.20
IPs:
IP: 10.244.0.20

Controlled By:
      StatefulSet/kafka
```

Configuring a Front-End Processor (FEP) on Your Live Energy Connect Cluster

Oracle Utilities Live Energy Connect can be configured to act as a Front-End Processor for Oracle Utilities Network Management System Flex SCADA system allowing it to send messages to the client.

The Oracle Utilities Live Energy Connect 25.12.0.0.0 message bus is configured by clients (for example, **Oracle Utilities Network Management System (NMS) Flex SCADA**) that send configuration information to LEC. From the client's perspective, Oracle Utilities Live Energy Connect 25.12.0.0.0 acts as a Front-End Processor (FEP) for the client.

Note: A single Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster can host multiple FEPs. For example, **NMS Flex SCADA** users will use the Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster's various applications by creating a FEP in **NMS** and configuring it to the connect to a particular FEP service running on the Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster. The FEP service (for example, "fep7-grpc-service") running on the Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster will then create and manage the resources requested by **NMS Flex SCADA**. By default, no FEP services are deployed on your Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster. The FEP service needs to be initialized with a specified, unique FEP ID that may vary depending on the client's configuration state.

> **Note**
>
> See the *Oracle Utilities Network Management System User Guide* Flex SCADA chapter for information on configuring FEPs with **NMS Flex SCADA**.

Use the following steps to create and configure a FEP on the Oracle Utilities Live Energy Connect 25.12.0.0.0 cluster so that a client, such as **NMS Flex SCADA**, can use the FEP service. These steps will install the FEP service on the cluster and configure the required `nginx` service configuration and `firewalld` configuration on the nodes in the cluster.

Add a FEP Service on Your Oracle Utilities Live Energy Connect 25.12.0.0.0 Cluster

On the OCNE localhost (the host from which the OCNE cluster was created), run the following steps:

1. Open a **Bash** shell session and change to the directory where the `lec run` file was previously unpacked. This process creates a directory with a name similar to

`installer_25.12....` Change into that directory, and then change to the `scripts` subdirectory.
The `create-fep-on-ocne2.sh` file should be present there. For example:

```
[lec@ugbu-phx-1147 ~]$ cd installer_<release-version>/scripts/
[lec@ugbu-phx-1147 scripts]$ ls -al
-rwxr--r--. 1 lec lec 25561 Dec 12 20:56 create-fep-on-ocne2.sh
```

2. Run the following command:

```
./create-fep-on-ocne2.sh --id <fep_id> --port <port_number>
```

Where `<fep_id>` is the unique numeric ID assigned to the FEP service by **NMS Flex SCADA**, and `<port_number>` is a currently unassigned port (configured in **NMS Flex SCADA**) on which the FEP service shall listen.

For example:

```
./create-fep-on-ocne2.sh --id 7 --port 50051
```

3. You will be prompted to press **Enter** to continue with this step.

> **Note**
>
> You can also exit by pressing **CTRL + C**.