



Installing vWLC and Host Linux with SUSE Linux

This chapter contains the following topics:

- [Installing SUSE Linux, on page 1](#)
- [Install KVM and Supporting packages, on page 2](#)
- [Enabling SSH, on page 2](#)
- [Network Configuration, on page 2](#)
- [Installing vWLC Using VMM, on page 4](#)

Installing SUSE Linux

Download SLEs 12 - <https://www.suse.com>. (You must create a login)

- eth0—for uplink (service-port of WLC); no IP address is required to this interface but should be connected and up.
- eth1—for WLC Management interface; no IP address is required to this interface but should be connected and up.
- eth2 or 3— for Linux accessibility; provide IP address to this interface, so that there is a network connectivity for Linux box and internet from it.



Note Before working on any other package or KVM/vswitch, check the Linux kernel. Make sure the kernel version is 3.12.36-38 or above.

If the kernel version is not 3.12.36-38 or above, upgrade it by performing the following steps:

1. Install SLES 12 on the server.
2. Once the server comes up, copy the kernel rpm to the machine.
3. On a terminal, execute **rpm --ivh <kernel>.rpm**.
The rpm is installed and would take some time to configure. You need not do anything else.
4. Reboot the machine once the installation is complete, and verify that the latest kernel is loaded using **uname --a**.

Install KVM and Supporting packages

Install KVM and supporting packages using the following commands:

```
zypper install openvswitch openvswitch-switch
zypper install kvm libvirt libvirt-python qemu virt-manager
```

Enabling SSH

Execute the following commands:

```
systemctl enable sshd.service → enabling sshd daemon
systemctl start sshd.service → starting ssh
netstat -an | grep :22 → to see if port# 22 is listening
```

Network Configuration

Creating a Bridge and Mapping it to Port (Ethernet Interface)

```
ovs-vsctl add-br ov_10nw
ovs-vsctl add-port ov_10nw eth0
ovs-vsctl add-br ov_9nw
ovs-vsctl add-port ov_9nw eth1
```

The bridge name must be the same as created in the XML file.

Viewing the Bridge Mapping

```
ovs-vsctl show
```

Example

```
linux-f8es:~ # ovs-vsctl show
51600b63-b508-45b0-9d0c-9f74036114c5
    Bridge "ov_9nw"
        Port "ov_9nw"
            Interface "ov_9nw"
                type: internal
        Port "eth1"
            Interface "eth1"
    Bridge "ov_10nw"
        Port "ov_10nw"
            Interface "ov_10nw"
                type: internal
        Port "eth0"
            Interface "eth0"
    ovs_version: "2.1.2"
```

Creating XML Files

Create two XML files; one for service-nw (10nw) and the other for management (9nw).

Example

```
10nw_eth0_ov.xml
9nw_eth1_ov.xml
```

Both XML files contain VLAN information based on the network, or based on what you want to allow.

Example: To Allow All VLANs

```
<network>
  <name>10-nw</name>
  <forward mode='bridge'>
  <bridge name='ov_10nw'>
  <virtualport type='openvswitch'>
  <portgroup name='vlan-any' default='yes'>
  </portgroup>
</network>
```

The bridge name must be the same as created during "ovs-vsctl" command.

Starting Open vSwitch

```
service openvswitch-switch start
```

Configuring Open vSwitch to Start When the System Boots

```
chkconfig openvswitch-switch on
```



Note vSwitch must be started before creating the bridge using above command.

Starting libvirt

```
service libvirtd restart
```

Allowing CDP Packets to Forward from Open vSwitch

```
ovs-vsctl set bridge ov_9nw other-config:forward-bpdu=true
```

Viewing the Virtual Network

```
virsh net-list --all
```

Deleting the Default Network

```
virsh net-undefine default
```

Creating Virtual Network

```
virsh net-define <xml_file_name>
```

Viewing the Virtual Network

```
virsh net-list --all
```

Starting the Virtual Network

```
virsh net-start <network_name_that is in the list>
```

Example

```

linux-f8es:~ # virsh net-list --all
Name                State      Autostart  Persistent
-----
default             inactive  no         yes

linux-f8es:~ # virsh net-undefine default
Network default has been undefined

linux-f8es:~ # virsh net-define 10nw_eth0_ov.xml
Network 10-nw defined from 10nw_eth0_ov.xml

linux-f8es:~ # virsh net-define 9nw_eth1_ov.xml
Network 9-nw defined from 9nw_eth1_ov.xml

linux-f8es:~ # virsh net-list --all
Name                State      Autostart  Persistent
-----
10-nw               inactive  no         yes
9-nw                inactive  no         yes

linux-f8es:~ # virsh net-start 10-nw
Network 10-nw started

linux-f8es:~ #
linux-f8es:~ # virsh net-start 9-nw
Network 9-nw started

linux-f8es:~ # virsh net-list --all
Name                State      Autostart  Persistent
-----
10-nw               active     no         yes
9-nw                active     no         yes

```

Installing vWLC Using VMM

To install vWLC using VMM in SUSE Linux, perform the following steps:

-
- Step 1** Similar to Fedora, go to the terminal and type **virt-manager**.
The Virt Manager (VMM) pop-up appears.
- Step 2** Follow the steps covered in [Fedora installation](#) using VMM.
-