

# Prepare the hardware and OS

## Install VLAN Module

- We need to install the VLAN module to run VLANs on the server.

```
#Install the VLAN package
sudo apt-get update
sudo apt-get install vlan
#Permanently load the module during boot time
sudo su -c 'echo "8021q" >> /etc/modules'
#Reboot the server
sudo reboot
#Confirm that it is loaded
lsmod | grep 8021q
```

## Install Bridge Utilities

- We need to install the bridge utilities to use the bridging capabilities on the server.

```
#Install the bridge-utils package
sudo apt-get update
sudo apt-get install bridge-utils
```

## Configure the Interfaces

- If your server has **only one** interface card, edit the **/etc/rc.local** file to contain the following:



We assume here the eth0 port now has the public IP address and give the dummy interface the name of eth1. Please consider this in the configurations after this page and change accordingly.

rc.local

```
#!/bin/bash

#Set up the dummy interface
/sbin/modprobe dummy
/sbin/ip link add dummy0 type dummy
/sbin/ip link set name eth1 dev dummy0
/sbin/ip link set dev eth1 address 00:22:22:ff:ff:ff
/sbin/ip link set eth1 up promisc on

#Now add the VLAN
/sbin/ip link add link eth1 name eth1.101 type vlan id 101
/sbin/ip link set eth1.101 up promisc on
/sbin/brctl addbr br0.101
```

```
/sbin/brctl addif br0.101 eth1.101
/sbin/ip addr add 10.101.0.1/16 dev br0.101
/sbin/ip link set dev br0.101 up

/sbin/ip link add link eth1 name eth1.102 type vlan id 102
/sbin/ip link set eth1.102 up promisc on
/sbin/brctl addbr br0.102
/sbin/brctl addif br0.102 eth1.102
/sbin/ip addr add 10.102.0.1/16 dev br0.102
/sbin/ip link set dev br0.102 up

/sbin/ip link add link eth1 name eth1.103 type vlan id 103
/sbin/ip link set eth1.103 up promisc on
/sbin/brctl addbr br0.103
/sbin/brctl addif br0.103 eth1.103
/sbin/ip addr add 10.103.0.1/16 dev br0.103
/sbin/ip link set dev br0.103 up

exit 0
```

- If your server has two or more physical interfaces we do on need to first define a dummy interface.

## rc.local

```
#!/bin/bash

#Now add the VLAN
/sbin/ip link add link eth1 name eth1.101 type vlan id 101
/sbin/ip link set eth1.101 up promisc on
/sbin/brctl addbr br0.101
/sbin/brctl addif br0.101 eth1.101
/sbin/ip addr add 10.101.0.1/16 dev br0.101
/sbin/ip link set dev br0.101 up

/sbin/ip link add link eth1 name eth1.102 type vlan id 102
/sbin/ip link set eth1.102 up promisc on
/sbin/brctl addbr br0.102
/sbin/brctl addif br0.102 eth1.102
/sbin/ip addr add 10.102.0.1/16 dev br0.102
/sbin/ip link set dev br0.102 up

/sbin/ip link add link eth1 name eth1.103 type vlan id 103
/sbin/ip link set eth1.103 up promisc on
/sbin/brctl addbr br0.103
/sbin/brctl addif br0.103 eth1.103
/sbin/ip addr add 10.103.0.1/16 dev br0.103
/sbin/ip link set dev br0.103 up
```

```
exit 0
```



We don't use a Netplan file since it does not currently support things like setting a card in promiscuous mode.

## Add a Systemd Service for rc.local

- Ubuntu changed the startup system in the more recent releases to Systemd.
- In order for the rc.local file to be run during startup under Systemd, we need a couple of extra steps
- Create the file **/etc/systemd/system/rc-local.service** with the following content:

```
# /etc/systemd/system/rc-local.service
[Unit]
Description=/etc/rc.local Compatibility
ConditionPathExists=/etc/rc.local

[Service]
Type=forking
ExecStart=/etc/rc.local start
TimeoutSec=0
StandardOutput=tty
RemainAfterExit=yes
SysVStartPriority=99

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

- Then

```
sudo touch /etc/rc.local
sudo chmod +x /etc/rc.local
sudo systemctl enable rc-local
```

- Check with

```
sudo systemctl start rc-local.service
sudo systemctl status rc-local.service
```

## Confirm it is correct

- Reboot the server and confirm that it came up with these bridges configured.

```
#Issue the ip a command to confirm the br0.101, br0.102 and br0.103 are up and has the correct IP Address.
#Also use the brctl command to show you the bridges present
system@rd:~$ brctl show
```

bridge name	bridge id	STP enabled	interfaces
br0.101	8000.000c294aafdf	no	eth0.101
br0.102	8000.000c294aafdf	no	eth0.102
br0.103	8000.000c294aafdf	no	eth0.103

From:  
<https://www.radiusdesk.com/wiki/> - **RADIUSdesk**

Permanent link:  
<https://www.radiusdesk.com/wiki/technical/openvpn-bridges-prep-os>

Last update: **2023/10/23 14:47**

