# DefCon 23 Presentation: Exploring Layer 2 Network Security in Virtualized Environments

---

# DefCon 23 Video Demo: Rogue DHCP/DNS server gaining root access to target

---

# DefCon 23 Video Demo: Rogue DHCP/DNS server ShellShock exploit proof of concept

---

# DefCon 23 Video Demo: Rogue DHCP/DNS server on Citrix XenServer 6.2 with Open vSwitch 1.4.6

---

# DefCon 23 Video Demo: MAC Flooding on Citrix XenServer 6.2 with Open vSwitch 1.4.6

---

# DefCon 23 Video Demo: MAC Flooding on Gentoo/Xen with Open vSwitch 2.0.0

---

# Open vSwitch on Gentoo – Setting Up Your First vSwitch

In my last post I outlined how to get Open vSwitch installed on Gentoo from source for version 1.11.0 as well as from portage using version 2.0.0. I also described how to associate Open vSwitch with with Xen based virtual machines. This guide will detail how to build your first virtual switch from scratch and create virtual switch ports associated with the virtual switch that persist upon a reboot of the host machine. The virtual ports can then be used with VirtualBox VMs and other generic applications that can make use of them.

If you followed my last post you should have Open vSwitch installed and the respective services running. You should also have created an initial bridge interface, this was called xenbr0 and was created with the following commands:

where enp4s0 is your physical interface that is bound to the bridge. This is necessary since this will be the “switch” port that “uplinks” the vSwitch to the rest of the physical network.

You should now be able to view the current bridge setup by using the following command:

Which should display output similar to this:

This shows that the bridge is created and has a single port attached to it labeled enp4s0 that is bound to the physical interface enp4s0.

At this point you should be fine if you were using this for a Xen based setup since the hotplug scripts provided with xen-

tools will take care of the virtual interface setup for each VM. However if you want to use Open vSwitch with VirtualBox based VMs on a Gentoo host read on …

In order to create virtual switch ports that you can bind to your virtual machines a tun/tap interface must be created for each port that will be needed. So if you wish to create a 16 port switch 16 tap interfaces will be necessary. These interfaces can be added to the vSwitch as follows:

First verify that the tun module is loaded

If this does not run verify that you have tun compiled as a module in your kernel.

And that the tun module is set to load on boot up in /etc/conf.d/modules:

Next create the tap interfaces for each virtual switch port, in this example I will create an 8 port switch.

Now check that they were added:

Which should display:

This indicates that the virtual ports have been associated with the vSwitch xenbr0, however the interfaces still need to be created. In order to do this entries need to be made in /etc/conf.d/net that describes each tap interface and how they will be configured at bootup. In the case of these tap interfaces we want them to be brought up but not configured. So in /etc/conf.d/net add the following entries:

Notice how they are all set to “null” this indicates that on bootup they will not be assigned an IP address nor will they

poll a DHCP server for an address.

Next create the device symlinks in /etc/init.d/ as you would for any other network interface so that they can be started on bootup:

Then add them to the default runlevel:

Now reboot the server and everything should start up correctly.

After the server comes back online verify that the new ports are all associated with the vSwitch.

Which should return output similar to this:

Finally fire up VirtualBox and create a new VM or go to the settings of an old VM and set the network interface to "bridged mode" and choose one of the new switch ports in the list. Fire up the VM and you should have a connection!

References:

Xen Networking – Xen
Openvswitch with Virtualbox

---

# Installing Open vSwitch on Gentoo (Xen Hypervisor)

The Gentoo ebuild for Open vSwitch does not seem to work with the latest available kernel as of this writing *(3.10.7-gentoo-r1)*. This post is documentation of the process that I

performed in order to successfully install Open vSwitch on a Gentoo server running the Xen hypervisor. This guide assumes that you already have a Gentoo environment configured and running with the Xen hypervisor available in portage.

*Note: See the update in the comment section below for how to install openvswitch-2.0.0 from portage!*

First make sure the following kernel settings are enabled for full Open vSwitch compatibility:

After you rebuild the kernel and reboot the machine you can load the openvswitch module by typing:

Next add an entry for the openvswitch module to /etc/conf.d/modules so it loads on each reboot:

In order to successfully install Open vSwitch it must be downloaded and installed from source. The latest source code can be downloaded here.

In this guide the *openvswitch-1.11.0.tar.gz* file was downloaded and extracted to /usr/src/openvswitch. Perform the following commands to build and install Open vSwitch from the downloaded source code.

Open vSwitch should now have files installed in /usr and /var

The ovs-* commands should also now be available in your path

Next it is necessary to create the openvswitch DB

Startup the Open vSwitch database server

Initialize the database

Then start up openvswitch

In order to have Xen use Open vSwitch as its default virtual interface add the following entry to /etc/xen/xl.conf

The physical Ethernet interface that will be used with Open vSwitch has to be set to null in /etc/conf.d/net

Finally create the first Open vSwitch bridge called xenbr0

*Note: See the update in the comment section below for how to install openvswitch-2.0.0 from portage!*

References:

How to Install Open vSwitch on Linux, FreeBSD and NetBSD

Xen Networking – Xen (Setting up Open vSwitch networking)

QEMU with Open vSwitch network