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

DefCon 23 Video Demo: MAC Flooding on Gentoo/Xen with 802.1d Bridging

---

Disabling User List in GDM (Gnome 3)

In a multi-user environment you really do not want a full list of usernames being displayed in the GDM login screen for security reasons. In order to disable this feature on my Gentoo lab systems running Gnome 3 and GDM I performed the following actions.

First delete the following file:

Then create the directory

Create a new file in that directory called:

With the following content:

Then create a new file:

With the following content:

Run the following command:

Then change over to TTY1 *(Ctrl+F1)* and restart GDM with:

Use *(Ctrl+F7)* to change back to the GDM login screen and you will notice that the user list is now gone. Each user will now be required to type their username before they are prompted for a password.

Reference:
How do I disable user list in GDM

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

Gentoo – FTP Server

If you have the need for a FTP server on your Gentoo system, Pure-ftp is a good choice. It is a lightweight, standards compliant, and production quality FTP server that is available in portage.

To install pure-ftp do the following:

The default use flags should be fine for most installations.

After it is done installing you need to edit the config file located in /etc/conf.d/pure-ftpd

Make sure to uncomment the following line:

So that it looks like this:

Now you need to figure out how you want users to authenticate to the FTP server.

To allow login to the FTP server with local user accounts use the following setting for AUTH:

For PAM authentication use the following:

And finally for built in virtual user support use the following:

The virtual user method stores the user information in the file /etc/pureftpd.pdb. In order to setup FTP virtual users do the following:

First setup the ftpgroup and the ftpuser system user

Now virtual users can be created by using the following syntax:

The -d flag specifies the user's home directory. This can be changed to any location you like as long as the user has sufficient privileges on the folder.

The following commands can be used to manage virtual users:

To delete a user:

To change a user's password:

To view a user's status:

To commit changes *(NOTE: changes are commited automatically when using the [-m] flag)*:

Lastly we need to start the pure-ftpd service and add it to the default runlevel:

NOTE: If you are running iptables you will want to add the following rule to allow FTP traffic:

Gentoo – Zoneminder

The Zoneminder ebuild in the portage tree has been broken for a couple of years now. Here is the process I use to build Zoneminder from source on a Gentoo x86 install:

1. Get Root Access

2. Make The Source Directory

3. Get The Source Package

Stable: *1.25.0*

Development: *svn*

4. Unpack The Sources

5. Install Required Dependencies

Add the following to /etc/portage/package.keywords:

Emerge all dependencies:

Install Sys::Mmap with cpan:

6. Configure and Build Zoneminder

Configure the sources (note this is for a localhost install, you can change it to whatever vhost you want)

Make sure to change the zm db username and password to reflect your setup.

Setup the DB

Import the Initial DB

Build it

After all that Zoneminder should be installed on your system and ready to run. Here are is an init script that I created to get Zoneminder started, stopped and reset as well as allow it to start up and shutdown on a machine power cycle.

Save the file as zoneminder and copy it to /etc/init.d/zoneminder then make sure to run

Now you can start and stop your server by doing:

If you want it to start on power up add it to the default runlevel

Thats it! You should now be able to access the Zoneminder web interface at http://serverip/zm.

Now go buy some cameras!

Gentoo – Framebuffer Splash Image

If you are tired of staring at a black console screen on your Gentoo box you can trick it out a bit using Gensplash. Gensplash or "fbcondecor" allows you to use different background images, fonts and colors to decorate your console so you don't have the standard black background and white text. It also allows you to have nice boot and shutdown screens. (Think of the boot process and console on the install CD).

First we have to configure the kernel to support the framebuffer devices. You will need the following options enabled in your kernel config:

Next we need to setup some use flags in /etc/portage/package.use:

Then we need to unmask and install the necessary packages:

Now start the fbcondecor service and add it to the default runlevel:

Then create an initrid image and add it's entry to grub.conf. Note: themes are located in /etc/splash.

grub.conf example:

Grub Notes:

- splash=verbose – means you will see text scroll by, set to silent to just see a fancy progress bar. (Depends on

theme)

- fadein – will fade into the framebuffer on boot
- theme:theme_name – the theme you are using
- video=vesafb:mtrr:3,ywrap – vesa framebuffer options
- vga=0x361 – The resolution supported by your video card in hex. Set to Auto to see a list of supported resolutions and their codes for your monitor, experiment, then set it to the one that works.
- CONSOLE=/dev/tty1 – what console do we start the framebuffer on

Reboot and behold the wonders of your new framebuffer!

Gentoo – Deny Hosts

If you find your ssh server is getting hit by a lot of brute force attempts from the internet, and want to do something to defend yourself against them then denyhosts is for you! It helps to alleviate some of the stress on your server that occurs when someone or lots of someones are trying to hack their way into your ssh server. Basically the service watches your ssh traffic, and if it sees an IP address hitting a threshold of failed attempts it adds the address to your /etc/hosts.deny file so that it is blocked from future access attempts.

Simply install the package through emerge:

The initial configuration in /etc/denyhosts.conf should suffice, however it is well commented and you can edit it to suit your needs. You may want to add your email address in the ADMIN_EMAIL = variable so that denyhosts can email you alerts.

Once it is installed and configured start it up and add it to the default runlevel:

Thats it! Your server will now block an IP address after 3 failed ssh login attempts.

But what do I do if I accidentally lock out a valid IP address? Glad you asked!

You need to remove the wrongfully accused IP address from the following files using this process:

First stop the denyhosts service:

Then remove the IP from the following files:

/etc/hosts.deny
/var/lib/denyhosts/hosts
/var/lib/denyhosts/hosts-restricted
/var/lib/denyhosts/hosts-root
/var/lib/denyhosts/hosts-valid

Now restart the service:

You should now be able to ssh from the blocked IP address once again.

Gentoo – IRC Server

Want to host your own IRC server on your gentoo box? Well here is how:

Make a backup copy of the original config file:

Now we need to edit the config file so type

Under the [Global] section uncomment and edit the following lines:

Then uncomment the following and leave them at their default settings.

Next under the [Operator] section you can add your server op info:

Now lets add a channel under the [Channel] section:

Save the file and exit. You can now test your install by typing which should give you no errors and display your config. If there are errors correct them and try the config test again. Now make sure your forward ports 6667 and 6668 to your server on your router then start your server and add it to the default runlevel by doing the following:

You should now be able to connect to your new IRC server with an IRC client such as XChat or Pidjin!