

This site is no longer the primary site. All the content has been migrated to the new G-Suite site. Updates to content are only being applied to the new site. To go to the new site click on the blue "More Details" button below.

[Ok, got it](#) [More details](#)

Home

[Brother MFC-456CN Ubuntu Driver](#)
[Resume-Short](#)

AWS

CentOS7

[CentOS7 Desktop](#)
[CentOS7 General](#)

KVM and OpenvSwitch

[KVM Add PV Guest VG](#)
[KVM DG Migration](#)
[KVM DNS OEL 6.5](#)
[KVM DNS UL 14.04](#)
[KVM LXC DNS UL 14.04](#)
[KVM MAIL OEL 6.5](#)
[KVM Oracle Clone](#)
[KVM OVS OEL 6.5](#)
[KVM OVS OEL 6.5](#)
[KVM OVS UEL 14.04](#)
[KVM Resize Swap LV](#)
[Remove KVM Disk](#)
[Remove KVM VBDP Disk](#)
[Resize KVM Disk](#)
[Solaris 10 KVM OVS 12c GI RAC](#)

Lenovo ThinkPad P70

[A Brief Guide to Running Multiple OS on Lenovo ThinkPad P70 Mobile Workstation](#)
[Bootting to Graphical UI Desktop Oracle UEK4 Kernel on Lenovo P70 Thinkpad](#)
[Configure Wired Network UEK3 Lenovo P70 ThinkPad](#)
[Configure Wireless Network UEK3 Lenovo P70 ThinkPad](#)
[Configuring External Monitors on Oracle Linux UEK4 Kernel on Lenovo P70 ThinkPad](#)

[OpenvSwitch \(OVS\)](#) >

Configuring VirtualBox VMs for OpenvSwitch Networking

Contents

[1 Summary](#)
[2 Getting OpenvSwitch Onto Oracle Linux](#)
[3 Setup Dynamic DNS DHCP](#)
[3.1 Configure /etc/nsswitch.conf on the VM](#)
[3.2 VLAN the OpenvSwitch Ports](#)
[3.3 Configure the VirtualBox VM to use the OVS](#)
[3.4 Configure /etc/dhcp/dhclient.conf](#)
[3.5 Edit /etc/ssh/sshd_config in VM \(optional\)](#)
[4 Troubleshooting](#)

Summary

I use OpenvSwitch as the networking solution for my project <https://github.com/gstanden/orabuntu-lxc> and it comes bundled with Orabuntu-LXC. But what if you install Orabuntu-LXC and you want to also run some VirtualBox VMs and you want those VMs to be on the same network as the LXC Linux containers and you want them to get their dynamic DHCP and DNS from the same source as the LXC containers? Actually, this is

Configuring WIFI on Oracle Linux UEK4 Kernel on Lenovo P70 ThinkPad

Fixing BooBoos with Multiple Linux Distros on Lenovo P70 ThinkPad Mobile Workstation

Flash Installation Oracle Linux UEK4 Lenovo P70 Mobile Workstation

Lenovo ThinkPad P70 BIOS Flash Upgrade

Samsung 256 Gb M.2 950 Pro NVME

Samsung [(evo950 nvme) vs (pm951 nvme)] vs. Toshiba [OCZ RD400 1Tb nvme] M.2

Ubuntu 17.10 Artful Aardvark Lenovo P70 Mobile Workstation

LXC Linux Containers

Install Oracle-Supported LXC Version on Oracle Linux 7 UEK4

LXC Environment OEL6

LXC in Windows 10 Bash Shell

LXC iproute2

LXC Oracle RDBMS

LXC OVS BIND DNS

LXC VLC OEL6.5

Source Build LXC Ubuntu 16.04

Veth Pair Cleanup LXC

LXD the Lighter-visor

Building Oracle Enterprise in LXD

OpenvSwitch (OVS)

Build NetworkManager 1.11.1 on Ubuntu 17.10

Configuring VirtualBox VMs for OpenvSwitch Networking

Install OpenvSwitch OEL UEK3

Networking Problem: "The Lawnmower Man"

OpenvSwitch Patch Ports

OVS Source Build Oracle Linux 6 UEK3 Kernel

OVS Source Build Oracle Linux 7 UEK4 Kernel

OVS Source Build Ubuntu Linux 16.04

OVS Switch-As-A-Service Oracle Linux 7 UEK4

Orabuntu-LXC

Configuring systemd-resolved in a NetworkManager Environment

LXC Containerized DNS-DHCP

relatively easy to do and this blog is intended to show how that is done. This was all done and tested on Oracle Linux 7 UEK4 OS. On other OS YMMV but this should work the same for CentOS7 and RedHat7. All bets are off if you go to Linux6 however - typically procedures are needed there. This is Linux7 specific here.

Getting OpenvSwitch Onto Oracle Linux 7

Use my guide [here for building OpenvSwitch from source](#) on Oracle Linux 7 UEK4.

Setup Dynamic DNS DHCP

Sometime I will add details on this but right now it's outside the scope of this particular blog post for time constraint reasons. I have some posts though that include how to do that for example [here](#). Once dynamic DHCP/DNS solution is in place the VirtualBox VM's can be hooked up.

Configure /etc/nsswitch.conf on the Host

This is needed just to make sure that /etc/hosts doesn't override your DNS values. If you find everything is working ok (DHCP is issuing an IP address and DNS lookup works for that IP address and DNS name) but ssh is not working using the DNS name,

Configurations (Ubuntu dnsmasq bind9)**Oracle ASM and ASMLib****ASMLib Source Build RHEL5****ASMLib Source Build RHEL6****Rename ASM SYSDG****Oracle in LXD****Oracle Linux UEK Kernel****Configuring IPMI Console Redirection****Getting Kernel Headers Oracle Linux 7 UEK****LXC Source Build Oracle Linux 7 UEK4****Oracle RAC****Re-IP RAC****Oracle RAC in LXC Linux Containers****12c ASM Flex Cluster on Violin Memory Array****Oracle 12c 6-node ASM GNS Flex RAC on LXC Linux Containers Ubuntu 15.04****Oracle 12c RAC ASM Flex Cluster on LXC Linux Containers Ubuntu 14.10****Oracle 12c RAC ASM Flex Cluster on LXC Linux Containers Ubuntu 14.10****Oracle 12c RAC ASM Flex Cluster on LXC Linux Containers Ubuntu 15.04****Oracle 12c RAC in GCE on Ubuntu 14.04.1****Oracle EE ASM 12c LXC Ubuntu 14.04****Oracle EE DB 11gR2 Docker Ubuntu 14.04****Oracle RAC 6-node 12c GNS ASM Flex Cluster Ubuntu 15.04 Automated****Oracle RAC 6-node 12c GNS ASM Flex Cluster Ubuntu 15.04 Install****Ubuntu 14.04.1 Oracle GI 12c ASM Flex Cluster on LXC Containers****Ubuntu LXC OEL OVS ASM GNS RAC (12c)****Oracle RMAN****RMAN Active Duplicate Script****Projects****11gR2 -> 12cR2 Dataguard Upgrade in OL LXC on Ubuntu 17.04****VMOS6 Oracle Snapshots****Robin Systems****Configuring HugePages for Oracle on CentOS 7 Robin Hosts****Robin 12c RAC ASM Flex Cluster**

then check to see if /etc/hosts has some different IP for that same DNS name and then remove it from /etc/hosts and edit /etc/nsswitch.conf to make sure resolver uses DNS first.

```
[root@ol172 openvswitch]# cat /etc/nsswitch.conf
#
# /etc/nsswitch.conf
#
# An example Name Service Switch config file. This file should be
# sorted with the most-used services at the beginning.
#
# The entry '[NOTFOUND=return]' means that the search for an
# entry should stop if the search in the previous entry turned
# up nothing. Note that if the search failed due to some other
# reason
# (like no NIS server responding) then the search continues with
# the
# next entry.
#
# Valid entries include:
#
# nisplus                Use NIS+ (NIS version 3)
# nis                    Use NIS (NIS version 2), also called YP
# dns                    Use DNS (Domain Name Service)
# files                  Use the local files
# db                     Use the local database (.db) files
# compat                Use NIS on compat mode
# hesiod                 Use Hesiod for user lookups
# [NOTFOUND=return]     Stop searching if not found so far
#
# To use db, put the "db" in front of "files" for entries you
# want to be
# looked up first in the databases
#
# Example:
#passwd:    db files nisplus nis
#shadow:    db files nisplus nis
#group:     db files nisplus nis

passwd:    files sss
shadow:    files sss
group:     files sss
#initgroups: files

#hosts:     db files nisplus nis dns
hosts:     dns files myhostname

# Example - obey only what
```

[Robin Server Install CentOS 7 VM](#)[Robin SLOB2 PDB Protocols](#)[Robin SLOB2 Testing Notes](#)[Robin VirtualBox OVS](#)[The Surfing Life](#)

SCST

[SCST Create Deb Package and Install](#)[SCST Debian \(DKMS\) Package Build from Source \(Ubuntu 15.04-18.04+\)](#)[SCST Debian \(DKMS\) Package Build from Source \(Ubuntu 14.04\)](#)[SCST Debian Package Build from Source \(Ubuntu 15.04-17.04+\)](#)[SCST Debian Package Build from Source \(Ubuntu 17.04\)](#)[SCST KVM OEL 6.5](#)[SCST KVM UEK 3.8](#)[SCST Linux SAN](#)[SCST LXC UL 14.04](#)[SCST OL UEK 3](#)[SCST Package Build Install \(Ubuntu 14.x-17.x\)](#)[SCST Source Build Oracle UEK](#)[SCST Ubuntu 15.04](#)[SCST Ubuntu 16.04](#)[Update GPG1 Key Email Address, Add UID, Secret Key](#)

The Surfing Life

Tools

[AWR Analyzer v3.02](#)[Cisco AnyConnect VPN CentOS 7](#)[DTrace Linux](#)[iotop](#)[iscsiadm](#)[lynx HTML CLI](#)[nupdate](#)[Orion](#)[perf_stat](#)[pkexec visudo \(no session for cookie\)](#)[QEMU Monitor Command](#)[sflow](#)[SLOB2](#)[SLOB2 \(12c RAC PDB Version\)](#)[Tunnels](#)

```

nisplus tells us...
#services:  nisplus
[NOTFOUND=return] files
#networks:  nisplus
[NOTFOUND=return] files
#protocols: nisplus
[NOTFOUND=return] files
#rpc:       nisplus
[NOTFOUND=return] files
#ethers:    nisplus
[NOTFOUND=return] files
#netmasks: nisplus
[NOTFOUND=return] files

```

```

bootparams: nisplus
[NOTFOUND=return] files

```

```

ethers:      files
netmasks:   files
networks:    files
protocols:   files
rpc:         files
services:    files sss

```

```

netgroup:    files sss

```

```

publickey:   nisplus

```

```

automount:   files
aliases:     files nisplus

```

```

[root@ol172 openvswitch]#

```

VLAN the OpenvSwitch Ports

I like to use VLAN on my OpenvSwitch to partition network traffic. You don't have to use VLAN. If you do use VLAN the script which configures the OpenvSwitch at boot needs to have the correct VLAN tags for the ports the VM and/or the LXC containers will use as shown below. Also, there is more detail about this script [here](#). Note that you can add more VLAN's to your "trunks" parameter (comma separated) and then you can have VM's on different VLAN using the same switch just in separate trunks.

Just a side note that I prefer using these iptables rules to establish WAN rather than bonding the

VDBench**virt-alignment-scan****Ubuntu 16.04****Canonical Ubuntu 16.04 Livepatch Service****Cisco AnyConnect VPN Ubuntu 16.04****Sitemap**

openvswitch bridge directly to a physical interface because this way whatever networking was already in place on the desktop or server should be unaffected and continue to work. I've built my openvswitch solutions with this goal in mind of making them additive not disruptive. That being said YMMV so test first in a VM to be sure this is compatible with your existing networking setup if you want to preserve that networking.

```
[root@ol72 openvswitch]# cat
cr1 ovs_sw1.sh
#!/bin/bash

ip tuntap add s1 mode tap
ip tuntap add s2 mode tap
ip tuntap add s3 mode tap
ip tuntap add s4 mode tap
ip tuntap add s5 mode tap
ip link set s1 up
ip link set s2 up
ip link set s3 up
ip link set s4 up
ip link set s5 up
ovs-vsctl add-br sw1
ovs-vsctl add-port sw1 s1
ovs-vsctl add-port sw1 s2
ovs-vsctl add-port sw1 s3
ovs-vsctl add-port sw1 s4
ovs-vsctl add-port sw1 s5

ip link set up dev sw1
ip addr add 10.207.39.1/24 dev sw1
ip route replace 10.207.39.0/24 dev sw1

ovs-vsctl set port sw1 trunks=10
ovs-vsctl set port sw1 tag=10
ovs-vsctl set port s1 tag=10
ovs-vsctl set port s2 tag=10
ovs-vsctl set port s3 tag=10
ovs-vsctl set port s4 tag=10
ovs-vsctl set port s5 tag=10

# GLS 20140825 Get active
external interface dynamically at
boot. Tested & works with
{wlan0, eth0, bnep0} on Ubuntu
14.04.1 Desktop x86_64.
# GLS 20140825 Interface "bnep0"
is Blackberry Z30 OS10 Bluetooth
Tether.

### BEGIN Get Active EXTIF
Dynamcially. ###
function GetInterface
{
ifconfig | egrep -B1 'inet' |
```

```

egrep -A1 'enp|wlp|wlan|eth|bnep'
| sed '$!N;s/\n/ /' | sed 's/ */
/g' | cut -f1,7 -d' ' | sed 's/
addr//' | head -1 | cut -f1 -d':'
}
Interface=$(GetInterface)
function GetIP
{
ifconfig | grep -A1 $Interface |
grep inet | sed '$!N;s/\n/ /' |
sed 's/ */ /g' | cut -f3 -d' '
}
### END Get Active EXTIF
Dynamically. ###

echo '          IP: '$(GetIP)
echo 'Interface: '$(GetInterface)

INTIF="sw1"
EXTIF=$(GetInterface)

echo 1 > /proc/sys/net/ipv4
/ip_forward

# clear existing iptable rules,
set a default policy
iptables -P INPUT ACCEPT
iptables -F INPUT
iptables -P OUTPUT ACCEPT
iptables -F OUTPUT
iptables -P FORWARD DROP
iptables -F FORWARD
iptables -t nat -F

function CheckIptablesRulesCount
{
sudo iptables -S | grep FORWARD |
grep sw1 | wc -l
}
IptablesRulesCount=$(CheckIptablesRi
while [ $IptablesRulesCount -ne 0
]
do
iptables -D FORWARD -i $EXTIF -o
$IINTIF -j ACCEPT > /dev/null 2>&1
iptables -D FORWARD -i $INTIF -o
$EXTIF -j ACCEPT > /dev/null 2>&1
IptablesRulesCount=$(CheckIptablesRi
done

# set forwarding and nat rules
iptables -A FORWARD -i $EXTIF -o
$IINTIF -j ACCEPT
iptables -A FORWARD -i $INTIF -o
$EXTIF -j ACCEPT
iptables -t nat -A POSTROUTING -o
$EXTIF -j MASQUERADE
# iptables -t nat -A POSTROUTING
-o wlp4s0 -j MASQUERADE
service dhcpd start
service named restart

[root@ol72 openvswitch]#

```

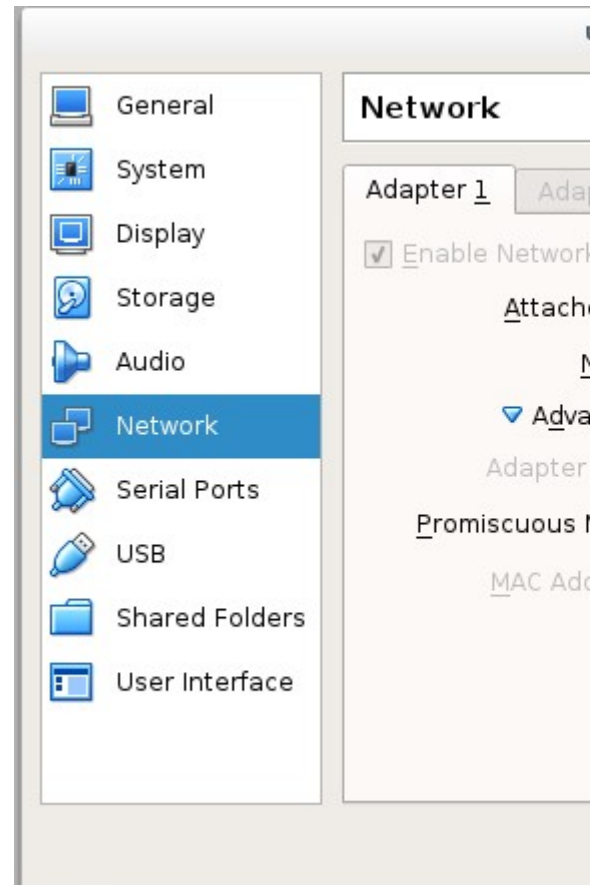
Configure the VirtualBox VM to use OpenvSwitch

This part is done simply with the VirtualBox GUI (unless you have more than 4 network interfaces to

configure with OpenvSwitch on VirtualBox). The switch itself should look like this if using VLAN tags.

```
[root@ol72 openvswitch]# ovs-vsctl show
05eaf5dd-98f5-422d-8de7-966049e48c00
    Bridge "sw1"
        Port "s4"
            tag: 10
            Interface "s4"
        Port "sw1"
            tag: 10
            trunks: [10]
            Interface "sw1"
                type: internal
        Port "s5"
            tag: 10
            Interface "s5"
                error: "could not open network device s5 (No such device)"
        Port "s3"
            tag: 10
            Interface "s3"
        Port "s1"
            tag: 10
            Interface "s1"
        Port "s2"
            tag: 10
            Interface "s2"
    ovs_version: "2.5.1"
[root@ol72 openvswitch]#
```

Networking the VirtualBox VM to OpenvSwitch is done as shown below. Be sure to select one of the ports on the switch (in this case "s2") that has the correct VLAN tag and also be sure to set "Promiscuous Mode" to "Allow All" and be sure to use "Bridged Adapter". (The fields that are greyed out are only greyed out because I took this screenshot when the VirtualBox VM was running).



Configure /etc/dhcp /dhclient.conf in VM (optional)

If for any reason the VM does not configure /etc/resolv.conf correctly at boot, it is possible to tell the resolver to use a different nameserver. This is done by configuring /etc/dhcp /dhclient.conf on the VM (not the host) as shown below.

```
[root@uekulele1 ~]# cat /etc/dhcp
/dhclient.conf
prepend domain-name-servers
10.207.39.1;

[root@uekulele1 ~]# cat
/etc/resolv.conf
# Generated by NetworkManager
search orabuntu-lxc.com
nameserver 10.207.39.1

[root@uekulele1 ~]#
```

Edit /etc/ssh/sshd_config in VM (optional)

If you find that login to the VM is very slow (you can confirm this by using `ssh -vvv username@vm-dns-name`) then it may be the issue with GSSAPI authentication. In that case you can optionally edit `/etc/ssh/sshd_config` and set these two parameters as shown below to make login lightening fast. Note this has nothing to do with OpenvSwitch per se, it's just a login thing but it's included here anyway. Also note that UseDNS has nothing to do with the DNS name resolution - your VM will still use your DNS to resolve hostnames. You can lookup what **UseDNS does if interested**.

```
[root@uekulele1 ~]# cat /etc/ssh/sshd_config
# $OpenBSD: sshd_config,v 1.93
2014/01/10 05:59:19 djm Exp $

# This is the sshd server system-
wide configuration file. See
# sshd_config(5) for more
information.

# This sshd was compiled with
PATH=/usr/local/bin:/usr/bin

# The strategy used for options
in the default sshd_config
shipped with
# OpenSSH is to specify options
with their default value where
# possible, but leave them
commented. Uncommented options
override the
# default value.

# If you want to change the port
on a SELinux system, you have to
tell
# SELinux about this change.
# semanage port -a -t ssh_port_t
-p tcp #PORTNUMBER
#
#Port 22
#AddressFamily any
#ListenAddress 0.0.0.0
#ListenAddress ::

# The default requires explicit
activation of protocol 1
#Protocol 2
```

```
# HostKey for protocol version 1
#HostKey /etc/ssh/ssh_host_key
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
#HostKey /etc/ssh
/ssh_host_dsa_key
HostKey /etc/ssh
/ssh_host_ecdsa_key
HostKey /etc/ssh
/ssh_host_ed25519_key

# Lifetime and size of ephemeral
version 1 server key
#KeyRegenerationInterval 1h
#ServerKeyBits 1024

# Ciphers and keying
#RekeyLimit default none

# Logging
# obsoletes QuietMode and
FascistLogging
#SyslogFacility AUTH
SyslogFacility AUTHPRIV
#LogLevel INFO

# Authentication:

#LoginGraceTime 2m
#PermitRootLogin yes
#StrictModes yes
#MaxAuthTries 6
#MaxSessions 10

#RSAAuthentication yes
#PubkeyAuthentication yes

# The default is to check both
.ssh/authorized_keys and
.ssh/authorized_keys2
# but this is overridden so
installations will only check
.ssh/authorized_keys
AuthorizedKeysFile
.ssh/authorized_keys

#AuthorizedPrincipalsFile none

#AuthorizedKeysCommand none
#AuthorizedKeysCommandUser nobody

# For this to work you will also
need host keys in /etc/ssh
/ssh_known_hosts
#RhostsRSAAuthentication no
# similar for protocol version 2
#HostbasedAuthentication no
# Change to yes if you don't
trust ~/.ssh/known_hosts for
# RhostsRSAAuthentication and
HostbasedAuthentication
#IgnoreUserKnownHosts no
# Don't read the user's ~/.rhosts
and ~/.shosts files
#IgnoreRhosts yes

# To disable tunneled clear text
passwords, change to no here!
#PasswordAuthentication yes
#PermitEmptyPasswords no
PasswordAuthentication yes

# Change to no to disable s/key
```

```
passwords
#ChallengeResponseAuthentication
yes
ChallengeResponseAuthentication
no

# Kerberos options
#KerberosAuthentication no
#KerberosOrLocalPasswd yes
#KerberosTicketCleanup yes
#KerberosGetAFSToken no
#KerberosUseKuserok yes

# GSSAPI options
# GLS 2016-10-12 turning off
GSSAPI    <-- Set GSSAPI to no
# GSSAPIAuthentication yes
GSSAPIAuthentication
no                <-- Set to no
here
GSSAPICleanupCredentials no
#GSSAPIStrictAcceptorCheck yes
#GSSAPIKeyExchange no
#GSSAPIEnablek5users no

# Set this to 'yes' to enable PAM
authentication, account
processing,
# and session processing. If this
is enabled, PAM authentication
will
# be allowed through the
ChallengeResponseAuthentication
and
# PasswordAuthentication.
Depending on your PAM
configuration,
# PAM authentication via
ChallengeResponseAuthentication
may bypass
# the setting of "PermitRootLogin
without-password".
# If you just want the PAM
account and session checks to run
without
# PAM authentication, then enable
this but set
PasswordAuthentication
# and
ChallengeResponseAuthentication
to 'no'.
# WARNING: 'UsePAM no' is not
supported in Red Hat Enterprise
Linux and may cause several
# problems.
UsePAM yes

#AllowAgentForwarding yes
#AllowTcpForwarding yes
#GatewayPorts no
X11Forwarding yes
#X11DisplayOffset 10
#X11UseLocalhost yes
#PermitTTY yes
#PrintMotd yes
#PrintLastLog yes
#TCPKeepAlive yes
#UseLogin no
UsePrivilegeSeparation sandbox
    # Default for new
installations.
#PermitUserEnvironment no
#Compression delayed
#ClientAliveInterval 0
#ClientAliveCountMax 3
```

```

#ShowPatchLevel no
#UseDNS yes
# GLS 2016-10-12 Be sure UseDNS
is no.  <-- Turn off UseDNS
UseDNS
no
<-- Here.
#PidFile /var/run/sshd.pid
#MaxStartups 10:30:100
#PermitTunnel no
#ChrootDirectory none
#VersionAddendum none

# no default banner path
#Banner none

# Accept locale-related
environment variables
AcceptEnv LANG LC_CTYPE
LC_NUMERIC LC_TIME LC_COLLATE
LC_MONETARY LC_MESSAGES
AcceptEnv LC_PAPER LC_NAME
LC_ADDRESS LC_TELEPHONE
LC_MEASUREMENT
AcceptEnv LC_IDENTIFICATION
LC_ALL LANGUAGE
AcceptEnv XMODIFIERS

# override default of no
subsystems
Subsystem sftp /usr/libexec
/openssh/sftp-server

# Example of overriding settings
on a per-user basis
#Match User anoncvs
#    X11Forwarding no
#    AllowTcpForwarding no
#    PermitTTY no
#    ForceCommand cvs server
[root@uekulele1 ~]#

```

The pool of switch ports s1, s2, s3, s4 ... can be used to connect 4 different VirtualBox VM network interfaces. If the VMs aren't running at the same time, you can even re-use ports for different machines although I don't recommend this. You can add more ports as you need them by just editing crt_ovs_sw1.sh and adding more ports. You can also give the switch ports meaningful names like "myserverhostname" although don't let them get too long I think there may be some character length limits on names of ports (but not sure about that).

One side note is that VMs work differently with OpenvSwitch than LXC containers do. With the VM's we can setup the ports ahead of time and then just use them by bridging the VM adapter to an OpenvSwitch port. In the case of LXC containers, the containers create the port 'on-the-fly' when starting so we don't pre-create them like this for LXC containers (but that's another story).

Troubleshooting

If you can ssh to the VirtualBox VM ok from the host using the DNS name, then that means your DNS solution is working properly (but you can do nslookup on both the DNS name and IP address of the VirtualBox VM to be sure). Then if your /etc/resolv.conf looks ok in the VM (points to the DNS/DHCP nameserver and has the right domain) and of course is similar to the /etc/resolv.conf on the host, then that should be OK too. But if still you cannot resolve names using "nslookup" or "ping" in the VM (but works properly on the host) then the problem is probably iptables rules on the host needed for the VM to network properly if "nslookup" and "ping" are not resolving in the VM, or it could be also sysctl.conf parameters as shown below. Be sure you have the following set on the HOST (not the VM) as shown below. This is discussed in more

detail at [**Venu Murthy's excellent post here**](#).

```
[root@ol172 Downloads]# sysctl -p
net.ipv4.conf.default.rp_filter =
0
net.ipv4.conf.all.rp_filter = 0
net.ipv4.ip_forward = 1

[root@ol172 Downloads]# cat
/etc/sysctl.conf
# System default settings live in
/usr/lib/sysctl.d/00-system.conf.
# To override those settings,
enter new settings here, or in an
/etc/sysctl.d/<name>.conf file
#
# For more information, see
sysctl.conf(5) and sysctl.d(5).
# GLS 20141226
http://thenewstack.io/solving-
a-common-beginners-problem-when-
pinging-from-an-openstack-
instance/
net.ipv4.conf.default.rp_filter=0
net.ipv4.conf.all.rp_filter=0
# GLS 20141226
http://thenewstack.io/solving-
a-common-beginners-problem-when-
pinging-from-an-openstack-
instance/
net.ipv4.ip_forward=1

[root@ol172 Downloads]#
```

Be sure you have iptables rule set correctly on the host also. The crt_ovs_sw1.sh script should have set the iptables rules that are at boot (see my post on [**openvswitch-as-a-service on Oracle Linux 7 here**](#)) required but if not, you can try a rule of this form as shown below which should be run on the HOST (not on the VM). In this example the "External Interface" is a wireless interface "wlp4s0" but it could also be a wired interface enpxxx or a bluetooth tether interface bnepxxx etc.. As noted, the crt_ovs_sw1.sh script should take care of identifying which interface has an ip address and assign this automatically and run the iptables rules onboot, but on a server with

multiple active IPs it might need some manual adjustment to get the `crt_ovs_sw1.sh` script set to what is required and so replacing variables with literal interface names in `crt_ovs_sw1.sh` is of course just fine if needed for a specific requirement.

```
iptables -t nat -A POSTROUTING -o  
wlp4s0 -j MASQUERADE
```

If you want to install LXC Linux container software as a next step, use my [**guide for building LXC on Oracle Linux 7**](#) from source here.

Comments

You do not have permission to add comments.