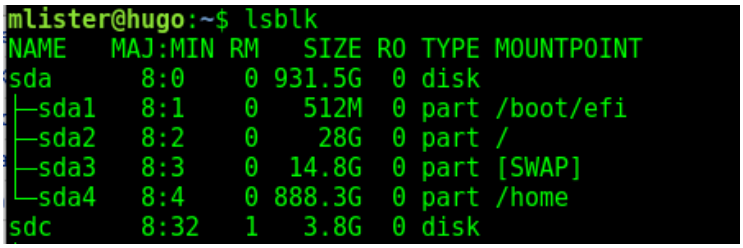
Everyone can be Root: Routing and Administration With Oracle VirtualBox
Matthew Lister
Ethan Payne

1. Download Ubuntu 18.04 LTS server iso and create a bootable USB. You can choose your download mirror from https://launchpad.net/ubuntu/+cdmirrors?_ga=2.186610729.1414317650.1563817197-1726790163.1553292886.

wget http://la-mirrors.evowise.com/ubuntu-releases/18.04/ubuntu-18.04.2-live-server-amd64.iso
lsblk

example output:



In this case sdc is our usb.

dd if=ubuntu-18.04.2-live-server-amd64.iso of=/dev/sdc

2. Install Ubuntu server on Physical machine. Make sure to select “Install OpenSSH server” so we will have remote access. This will give us a single user with sudo (administrator privledges).

3. Begin by adding some software we will need.

$ sudo apt update
$ sudo apt
$ sudo apt install virtualbox virtualbox-guest-additions-iso
$ sudo apt install xorg
$ sudo apt install nginx

Also download another copy of Ubuntu server.

4. Start X server and nginx. Also add a .Xauthority file to the user’s home directory (this can be an empty file but it must exist to use X11 forwarding).

$ sudo systemctl start graphical.target
$ sudo systemctl start nginx
$ sudo systemctl enable nginx
$ cd ~
$ touch .Xauthority

5. We can now use ssh to administer to our server and no longer need to be at the machine. If we want to use a GUI program while using ssh we use the ssh -X command.

$ssh -X user@server

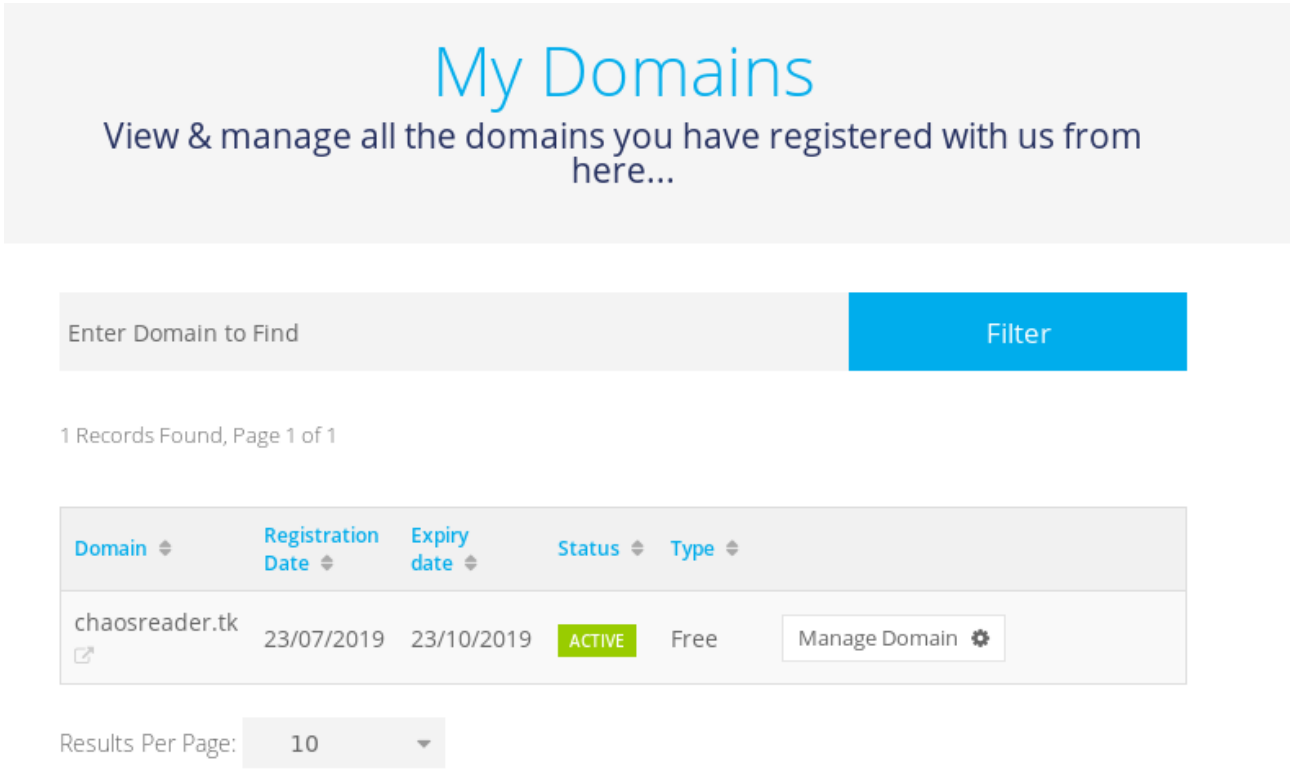
5.a Here is where we would create DNS records. We need an A record connect the name to the ip address. Then we can add any number of CN records to point to our server’s name. This can be emulated with the hosts file (/etc/host on a Linux machine).

Here is our hosts file:

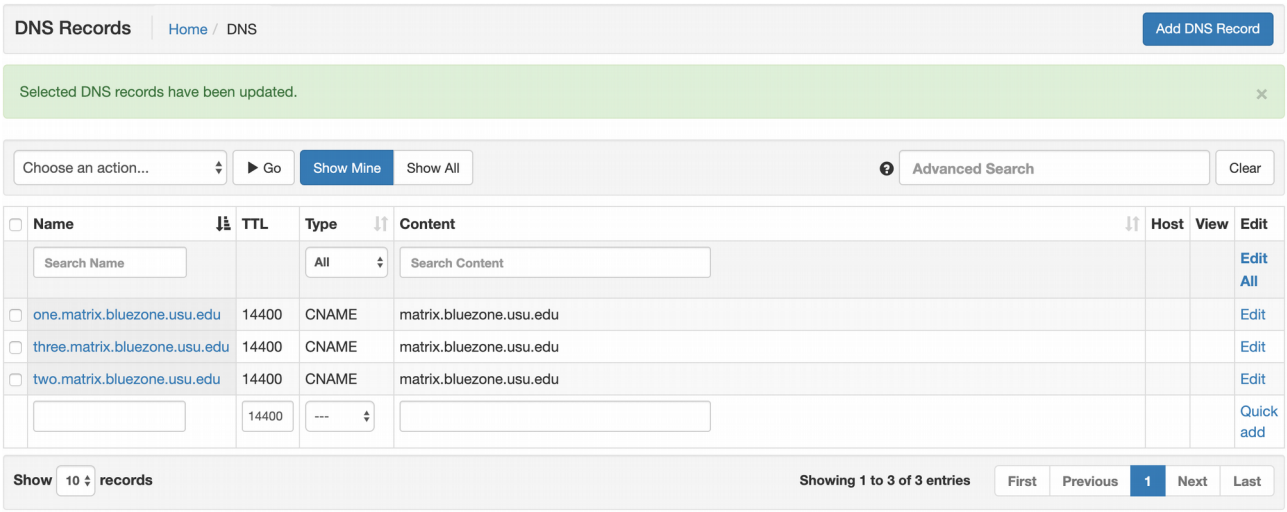
```
127.0.0.1       localhost
127.0.1.1       hugo.usu.edu    hugo
129.123.76.161  matrix.bluezone.usu.edu one     matrix.usu.edu
129.123.76.161  one.matrix.bluezone.usu.edu     one.matrix.usu.edu      one.usu.edu
129.123.76.161  two.matrix.bluezone.usu.edu     two.matrix.usu.edu      two.usu.edu
129.123.76.161  three.matrix.bluezone.usu.edu   three.matrix.usu.edu    three.usu.edu

# The following lines are desirable for IPv6 capable hosts
::1     localhost ip6-localhost ip6-loopback
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Notice how all those names resolve to the same IP address!

5b. We have set up ch*****r.ddns.net to point to my home router (as comcast sometimes changes my IP adress). This is the equivalent to an A record. We then registered ch*****r.tk to point the first domain name. This is a CN record. Both names resolve to the same IP but can be handled by nginx (or apache) separately. Furthermore, we can use nginx as a reverse poxy server to funnel requests into our private network.



5.c You may be self hosting or using you school's DNS, but, the end result is you can stack domain names onto the same IP address to route all the traffic you want to you outward facing IP.



A reverse proxy handles traffic to multiple websites behind a firfewall by receiving the layer 4 traffic and repackaging them to send on to the internal network. However, not all network services have good reverse proxies.

6. To lay the foundation of opening our internal network up to services besides http/https we will set up ip forwarding.

Modify the /etc/sysctl.conf file by adding

net.ipv4.ip_forward = 1

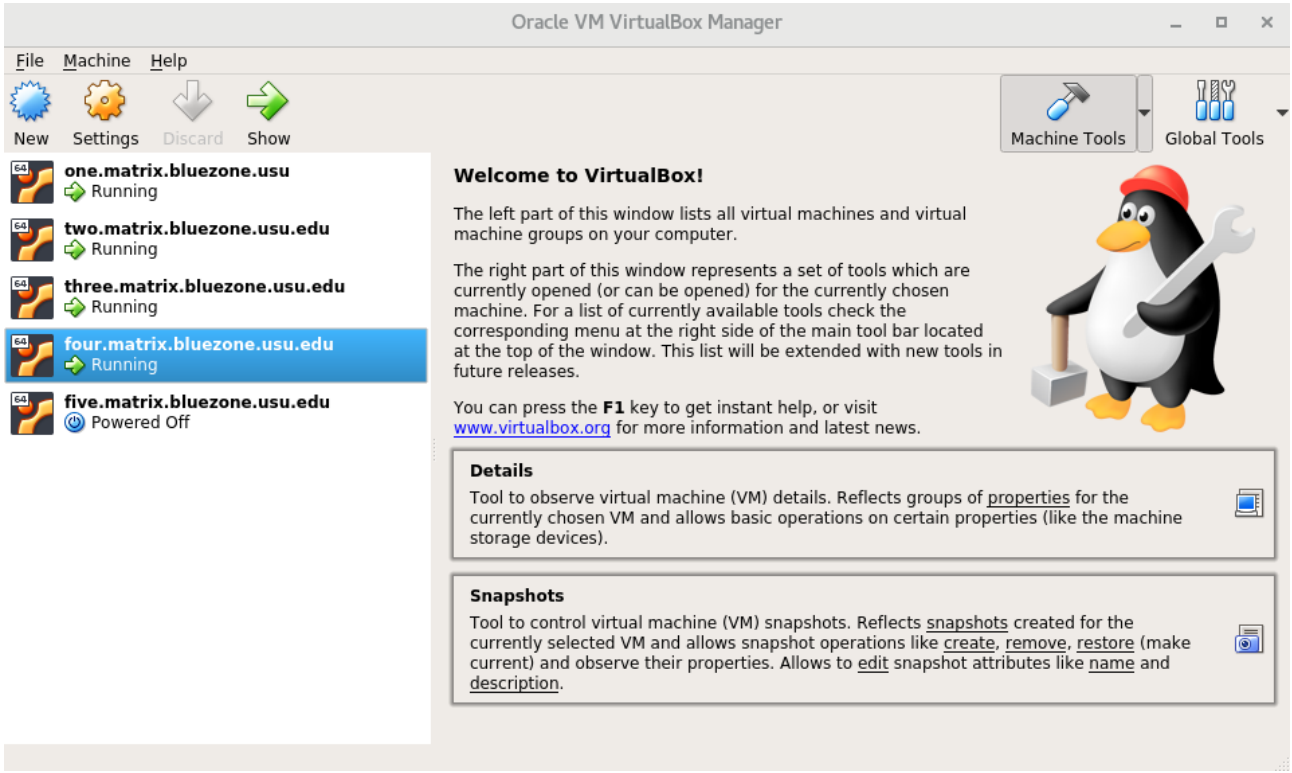
to the end of the file. Then run

sysctl -p

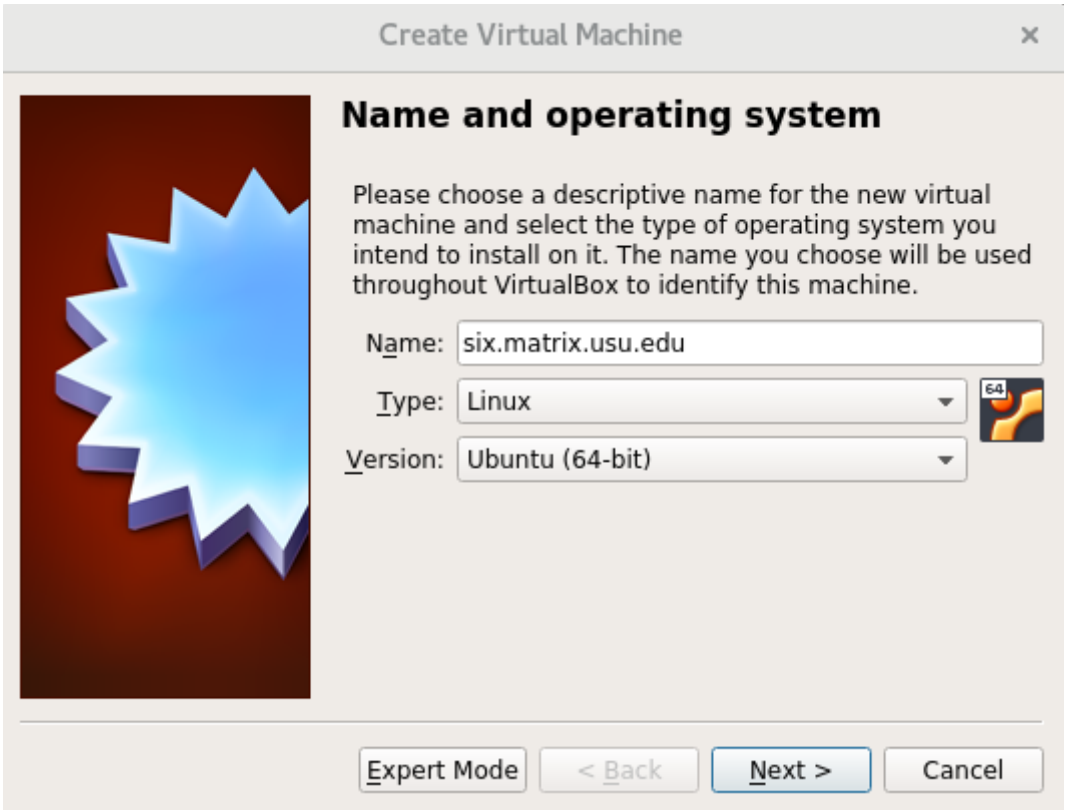
This will allow ip forwarding when we latter configure port forwarding (iptables) to provide services other than http/https.

7. Now we can built some virtual machines.

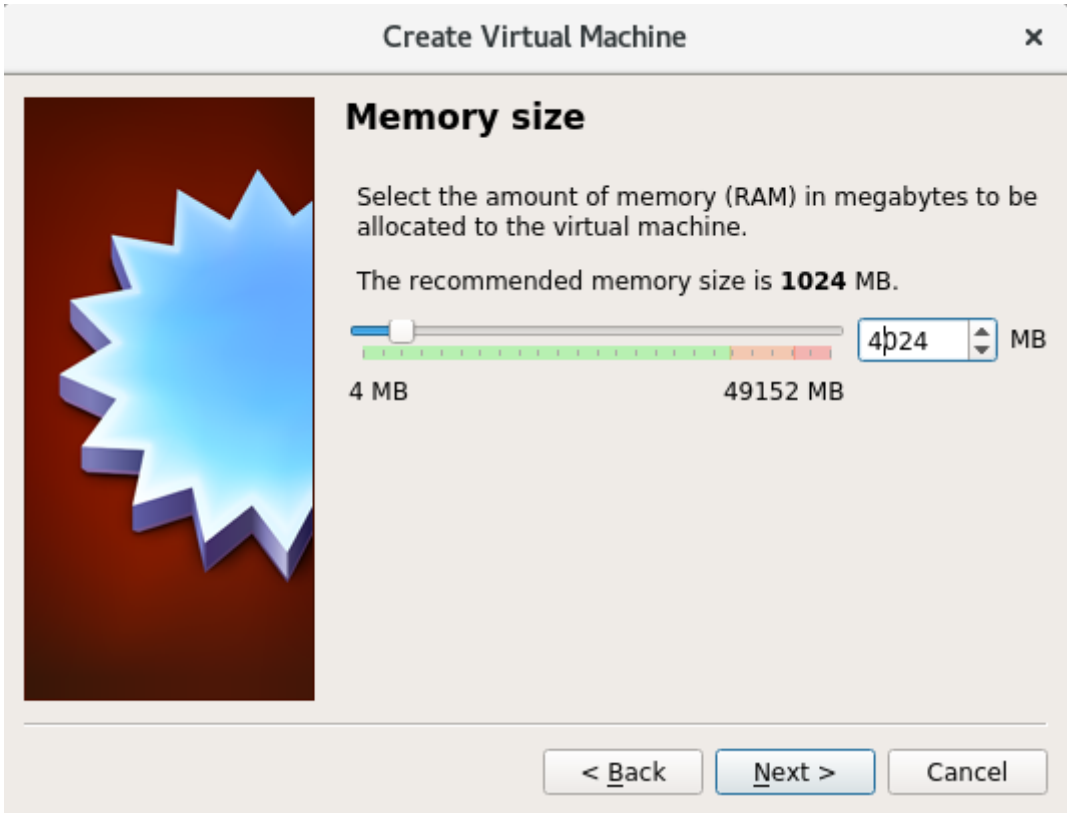
$ virtualbox



7.b. Create a new machine



7.c. Allocate RAM. Remember, this allocation will be assigned whenever the virtual machine is turned on so it will come from the hypervisor's total and not be available for other system processes. We choose 4gb per machine, but 2gb should be sufficient.



7.d. Create a virtual hard drive for the new machine. We chose VDI, 40gb, and dynamic allocation.

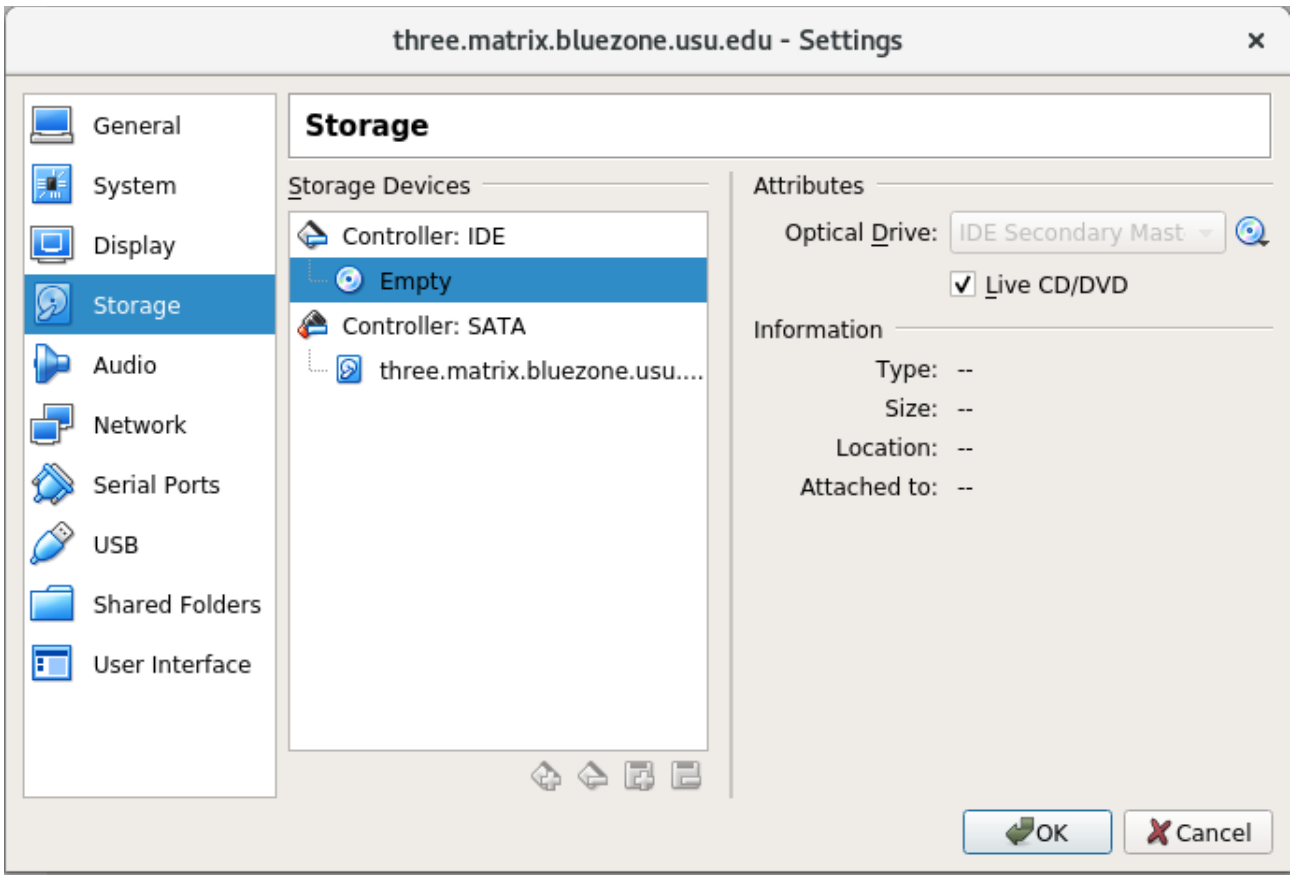

Create Virtual Hard Disk
Hard disk file type
Please choose the type of file that you would like to use for the new virtual hard disk. If you do not need to use it with other virtualization software you can leave this setting unchanged.
VDI (VirtualBox Disk Image)
VHD (Virtual Hard Disk)
VMDK (Virtual Machine Disk)
Expert Mode
< Back
Next >
Cancel


Create Virtual Hard Disk
Storage on physical hard disk
Please choose whether the new virtual hard disk file should grow as it is used (dynamically allocated) or if it should be created at its maximum size (fixed size).
A dynamically allocated hard disk file will only use space on your physical hard disk as it fills up (up to a maximum fixed size), although it will not shrink again automatically when space on it is freed.
A fixed size hard disk file may take longer to create on some systems but is often faster to use.
Dynamically allocated
Fixed size
< Back
Next >
Cancel

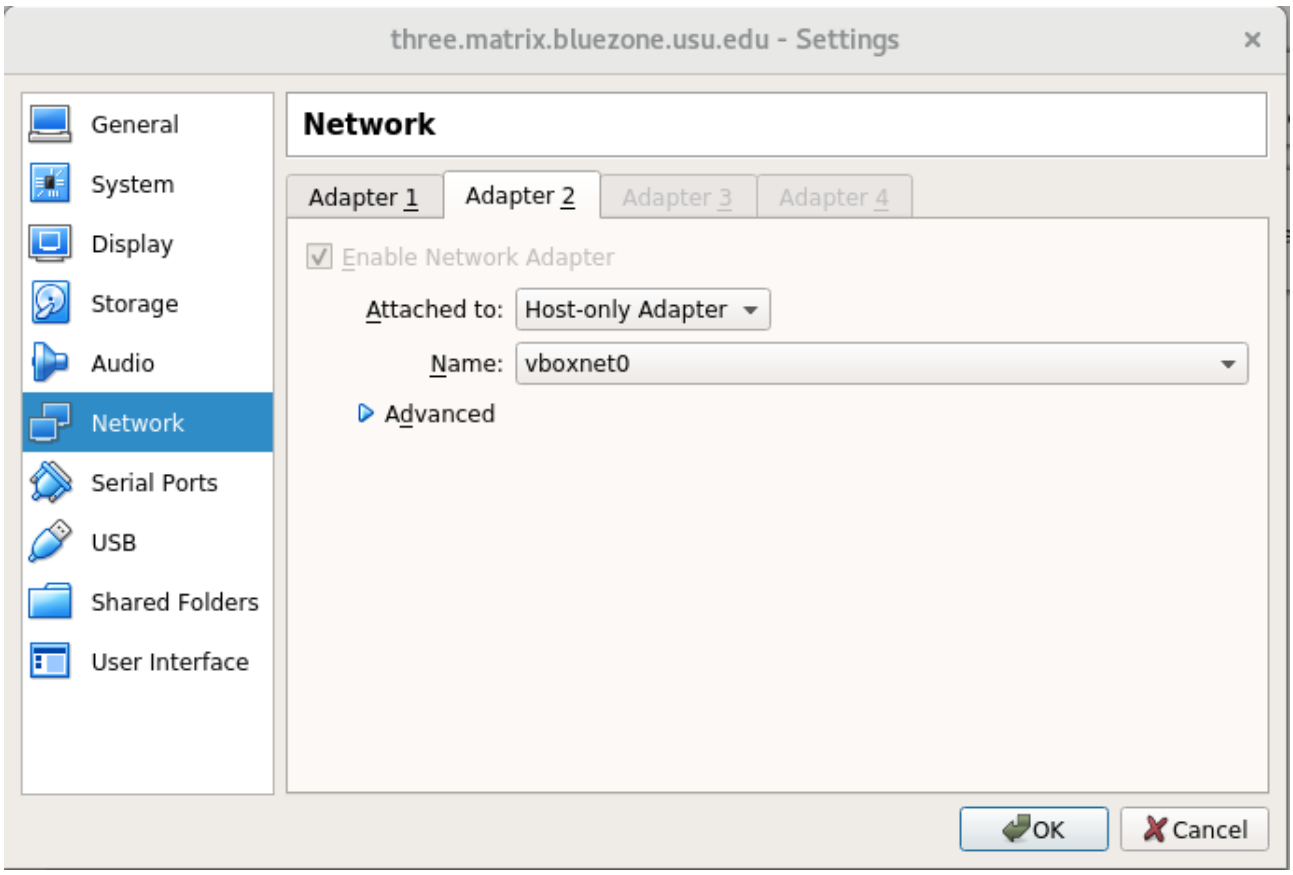
7.e. After the hard drive has been created, we will adjust the settings on our newly created VM. Remember to make all the configuration changes before clicking ‘ok’ or you will have to re open the settings tab to finish.

7.f. Under the system tab, you can increase the number of processors.

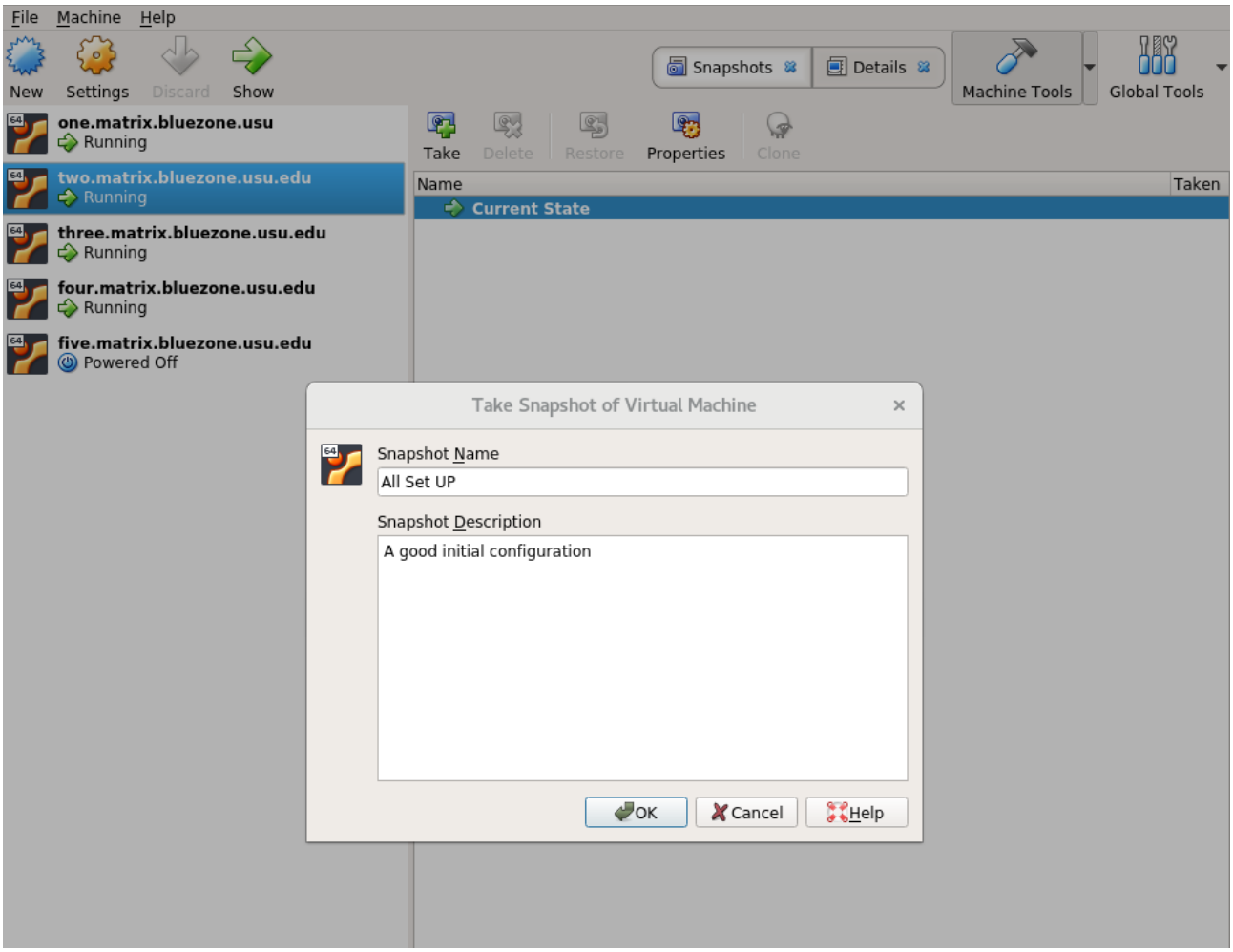
7. g Under Storage, click on the CD icon. Check the Live CD/DVD box and click on the 2nd CD icon crop down menu. Select the iso that was downloaded earlier.



7.h. Under the network tab, Adapter 1 will show NAT. We want to enable adapter 2. Click the tab for adapter 2, select the check mark to “Enable network adapter”. Select “host-only adapter” from the drop down menu.



Adding the host only adapter will give each virtual machine an address on the 192.168.56.0/24 subnet. Typically, the host will be 192.168.56.1 and the guests will start at 192.168.56.101.



8.a. Now we can click the start arrow on virtual box and install Ubuntu from the iso to our virtual machine's hard drive. For example, we will install openssh during installation, and apache2 from the command line. We set up an apache server on this machine with the default values. We also created a second machine following the same steps as before.

8.b On the new machine it may be nice to have some files automatically added when create new users to run the Vms. We will use our admin user to modify /etc/skel/. The contents of /etc/skel/ determine the initial contents of a user's home directory.

$ sudo su
# cd /etc/skel
# touch .Xauthority
# mkdir Documents
# mkdir Downloads

9.a On the virtual machine, view the networking information with the ip addr command.

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:3f:b2:43 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 84817sec preferred_lft 84817sec
    inet6 fe80::a00:27ff:fe3f:b243/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:c7:c2:cd brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.101/24 brd 192.168.56.255 scope global dynamic enp0s8
       valid_lft 900sec preferred_lft 900sec
    inet6 fe80::a00:27ff:fec7:c2cd/64 scope link
       valid_lft forever preferred_lft forever
```

9.b We can view the default route with

ip route show default

In this case, enp0s8 is the adapter to the 192.168.56.1 gateway, so we will make it the default route with this commnad:

ip route add default via 192.168.56.1 dev enp0s8

9.c. If we forward anything through our host, we want to be able to respond. We use iptables to allow a response from our internal network. [On the host]

iptables -t nat -A POSTROUTING -o eno2 -j MASQUERADE

Remember the -o flag is the outgoing interface. Entering ip addr on the hypervisor [host] show that eno2 is the outward facing network adaptor

```
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eno1: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether a4:ba:db:0b:c8:ae brd ff:ff:ff:ff:ff:ff
3: eno2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether a4:ba:db:0b:c8:b0 brd ff:ff:ff:ff:ff:ff
    inet 129.123.76.161/24 brd 129.123.76.255 scope global dynamic eno2
       valid_lft 27601sec preferred_lft 27601sec
    inet6 fe80::a6ba:dbff:fe0b:c8b0/64 scope link
       valid_lft forever preferred_lft forever
4: eno3: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether a4:ba:db:0b:c8:b2 brd ff:ff:ff:ff:ff:ff
5: eno4: <BROADCAST,MULTICAST> mtu 1500 qdisc noop state DOWN group default qlen 1000
    link/ether a4:ba:db:0b:c8:b4 brd ff:ff:ff:ff:ff:ff
6: vboxnet0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 0a:00:27:00:00:00 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.1/24 brd 192.168.56.255 scope global vboxnet0
       valid_lft forever preferred_lft forever
    inet6 fe80::800:27ff:fe00:0/64 scope link
       valid_lft forever preferred_lft forever
```

10. Now we can set up nginx as proxy server. Nginx lives in /etc/nginx. The file /etc/nginx/sites-available/default is link to the sites-enabled folder so changes made to sites-available/default automatically makes things enabled. We will use nginx as a reverse proxy to mangle http packets sent to the physical machine's ip address and forward them to the ip address of the virtual machine webservers. NOTE: These configurations work on a complete match principle. So while all virtual

servers are listening for [anything]:80[port] the match comes from the server name. Multiple names can be given. Proxy_pass gives the internal ip of the virtual machines (from setting up the host-only adapter)

```
nero@matrix:/etc/nginx$ cat sites-available/default
server {
        listen 80;
        listen [::]:80;

        server_name matrix.bluezone.usu.edu matrix.usu.edu;

}

server {
        listen 80;
        listen [::]:80;

        server_name one.matrix.bluezone.usu.edu one.matirx.usu.edu      one.usu.edu;

        location / {
                proxy_pass http://192.168.56.101:80;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;

        }
}

server {
        listen 80;
        listen [::]:80;

        server_name two.matrix.bluezone.usu.edu two.matrix.usu.edu      two.usu.edu;

        location / {
                proxy_pass http://192.168.56.102:80;
                proxy_http_version 1.1;
                proxy_set_header Upgrade $http_upgrade;
                proxy_set_header Connection 'upgrade';
                proxy_set_header Host $host;
                proxy_cache_bypass $http_upgrade;

        }
}
```

11. iptables can now be used to forward other services into our internal network. We have set up a php server running on server three.matrix.bluezone.usu.edu at port 8000. To forward traffic on port 8000 we add the command

iptables -t nat -A PREROUTING -p tcp --dport 8000 -j DNAT –to-destination 192.168.56.104:8000

For another example, we set up a samba server on the same host. Samba needs TCP 139, 445 and UDP 137, 138 to function. So we give the host the following iptables command:

iptables -t nat -A PREROUTING -p tcp --dport 139 -j DNAT --to-destination 192.168.56.104:139
iptables -t nat -A PREROUTING -p tcp --dport 445 -j DNAT --to-destination 192.168.56.104:445
iptables -t nat -A PREROUTING -p udp --dport 137 -j DNAT --to-destination 192.168.56.104:137
iptables -t nat -A PREROUTING -p udp --dport 138 -j DNAT --to-destination 192.168.56.104:138

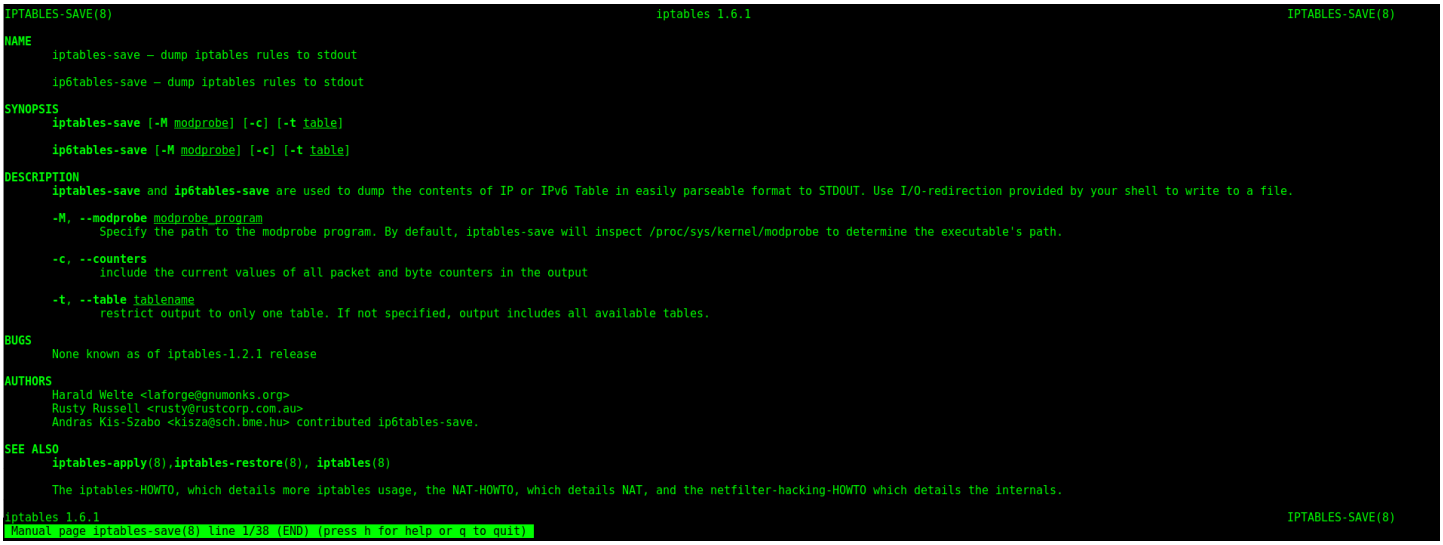
Remember, because we only have one outward facing address, we are limited to 64000 ports. So we must do some planning with the services we offer. Either use a reverse proxy (ideal for web traffic),

forward the common port to the server handling the traffic, or have a server listening for a non-traditional port and forward that port.

12. Unfortunately, how routing information is stored varies with Linux distributions we are not going to show how to script loading these configurations on startup. However, we will use this opportunity to demonstrate the linux man command. For example the command

man iptables-save

Displays a help page like this:

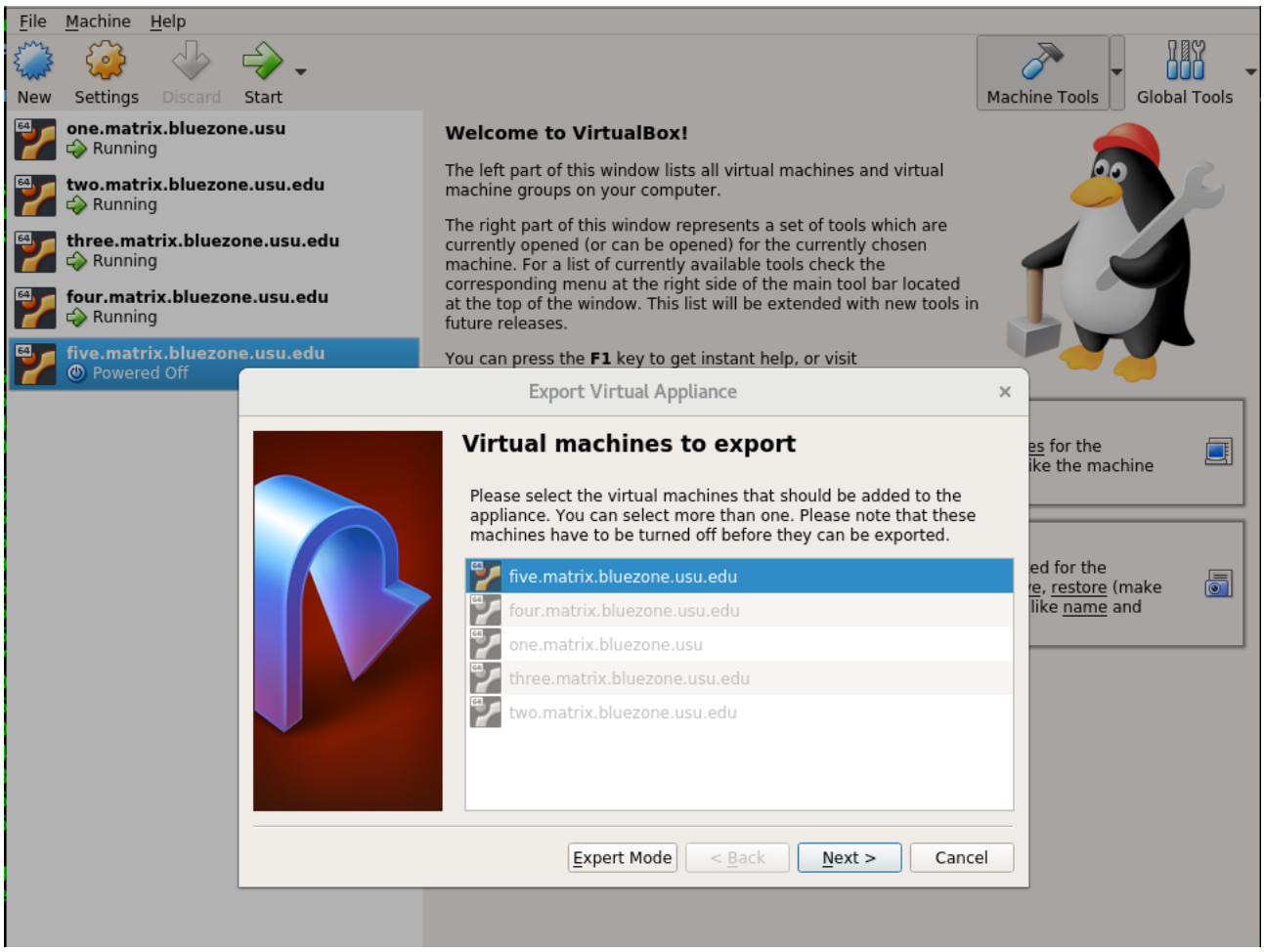


13. With the current setup, users will use the host as a jump box to administer to the individual vms. The syntax is:

$ ssh -J [user]@[hypervisor] [user]@[vm]

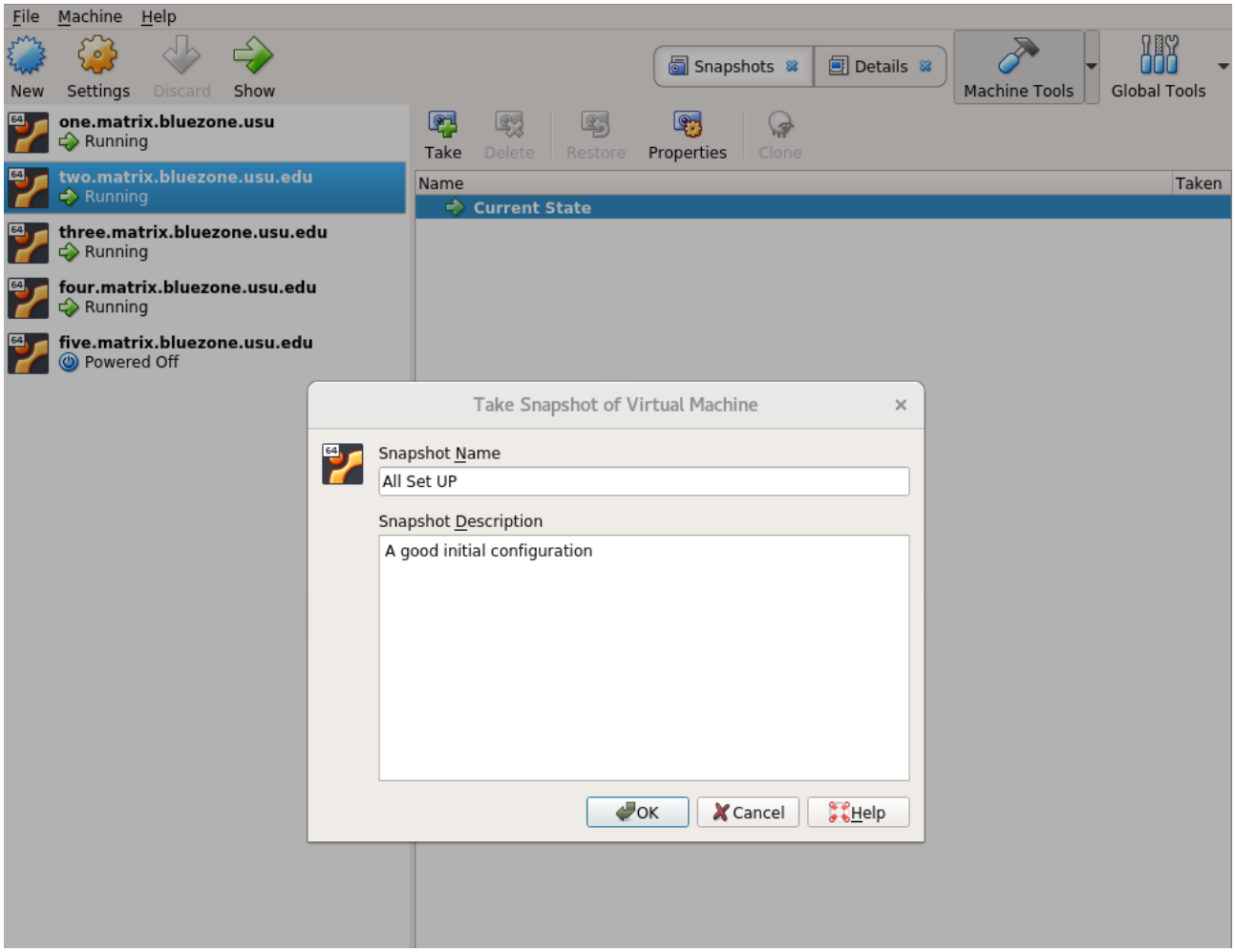
Alternatively, you can use iptables to send all port 22 requests to a vm and then jump from there. This shrinks the security footprint of the hyper visor. Just don’t forget to set up a way to administer to the hyper visor.

14. backups in Virtualbox use the export appliance function under the file tab. Unfortunately, the linux cp command will not work.



Follow the prompts and create a copy of your machine.

15. snapshots. Virtualbox gives the user the option of creating a snapshot which captures the machine state much like an incremental backup. It can be handy when you have the basic machine all set up and a new admin is just starting out. Snapshots can be taken even when the machine is running. The snapshot tab can be found under the machine tools drop down menu.


File
Machine
Help
New
Settings
Discard
Show
Snapshots
Details
Machine Tools
Global Tools
one.matrix.bluezone.usu
Running
two.matrix.bluezone.usu.edu
Running
three.matrix.bluezone.usu.edu
Running
four.matrix.bluezone.usu.edu
Running
five.matrix.bluezone.usu.edu
Powered Off
Take
Delete
Restore
Properties
Clone
Name
Taken
Current State
Take Snapshot of Virtual Machine
Snapshot Name
All Set UP
Snapshot Description
A good initial configuration
OK
Cancel
Help