Network Security: Practical Work (Not used in 2016–17)

i.g.batten@batten.eu.org

Solaris NIC Setup

```
igb@solaris:~$ sudo dladm create-etherstub ether0
Password:
igb@solaris:~$ sudo dladm create-vnic -l ether0 test0
igb@solaris:~$ sudo dladm create-vnic -l ether0 test1

     LINK                CLASS     MTU    STATE    OVER
net0                 phys      1500   up       --
ether0               etherstub 9000   unknown  --
test0                vnic      9000   up       ether0
test1                vnic      9000   up       ether0
```

Result: a stub ethernet, with two interfaces plugged into it.

Solaris Zone Setup

```
igb@solaris:~$ sudo zonecfg -z secondzone
Use 'create' to begin configuring a new zone.
zonecfg:secondzone> create
create: Using system default template 'SYSdefault'
zonecfg:secondzone> add net
zonecfg:secondzone:net> set physical=test1
zonecfg:secondzone:net> end
zonecfg:secondzone> verify
zonecfg:secondzone> commit
zonecfg:secondzone>
igb@solaris:~$ sudo zoneadm -z secondzone install
The following ZFS file system(s) have been created:
    rpool/VARSHARE/zones/secondzone
Progress being logged to /var/log/zones/zoneadm.20150212T130046Z.secondzone.install
       Image: Preparing at /system/zones/secondzone/root.

 Install Log: /system/volatile/install.8249/install_log
 AI Manifest: /tmp/manifest.xml.aRaOfq
  SC Profile: /usr/share/auto_install/sc_profiles/enable_sci.xml
    Zonename: secondzone
Installation: Starting ...

        Creating IPS image
Startup linked: 1/1 done
```

Zone Setup

- Will now run for some time dragging over the packages needed to install the zone
- They’re cached, so if you build more zones it’ll be a lot quicker

Zone Finishing

```
Updating package cache                           0/0
Updating image state                            Done
Creating fast lookup database                   Done
Updating package cache                           1/1
Installation: Succeeded

        Note: Man pages can be obtained by installing pkg:/system/manual

 done.

        Done: Installation completed in 852.406 seconds.


  Next Steps: Boot the zone, then log into the zone console (zlogin -C)

              to complete the configuration process.

Log saved in non-global zone as /system/zones/secondzone/root/var/log/zones/
zoneadm.20150212T130046Z.secondzone.install
igb@solaris:~$
```

Access the Zone

```
igb@solaris:~$ sudo zoneadm -z secondzone boot
Password:  // because it's been more than five minutes
igb@solaris:~$ sudo zlogin -C secondzone
[Connected to zone 'secondzone' console]
```

Counts down setting some things up, and then drops you into a configuration screen

Follow the obvious prompts

System Configuration Summary

Review the settings below before continuing. Go back (F3) to make changes.

Computer name: secondzone

Network:
Manual Configuration: test1/v4
IP Address: 192.168.141.100/24
Netmask: 255.255.255.0

Time Zone: UTC
Locale:
Default Language: English
Language Support: English (United States)
Username: igb

Support configuration:
Not generating a Support profile as OCM and ASR services are not installed.

What do you have?

- secondzone is a running Solaris instance (let's not worry about the precise details of what "instance" means here)

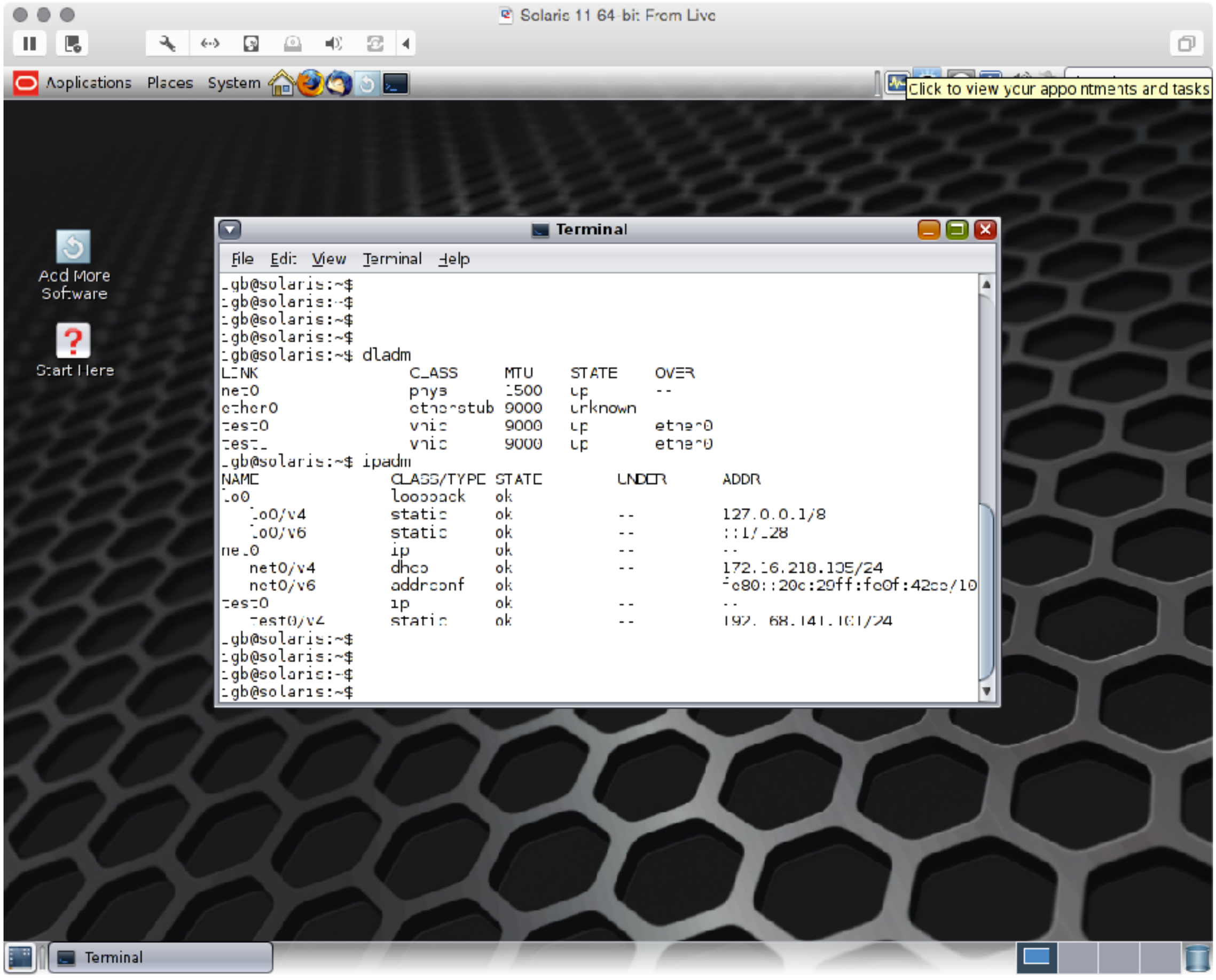
- IP address 192.168.141.100, purely internal to the Solaris VM

Now set up the other interface

```
igb@solaris:~$ sudo ipadm create-ip test0
igb@solaris:~$ sudo ipadm create-addr -T static -a 192.168.141.101/24 test0
test0/v4
igb@solaris:~$ ifconfig -a
lo0: flags=2001000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv4,VIRTUAL> mtu 8232 index 1
        inet 127.0.0.1 netmask ff000000
net0: flags=100001004843<UP,BROADCAST,RUNNING,MULTICAST,DHCP,IPv4,PHYSRUNNING> mtu 1500 index 5
        inet 172.16.218.135 netmask ffffff00 broadcast 172.16.218.255
test0: flags=100001000843<UP,BROADCAST,RUNNING,MULTICAST,IPv4,PHYSRUNNING> mtu 9000 index 6
        inet 192.168.141.101 netmask ffffff00 broadcast 192.168.141.255
lo0: flags=2002000849<UP,LOOPBACK,RUNNING,MULTICAST,IPv6,VIRTUAL> mtu 8252 index 1
        inet6 ::1/128
net0: flags=120002004841<UP,RUNNING,MULTICAST,DHCP,IPv6,PHYSRUNNING> mtu 1500 index 5
        inet6 fe80::20c:29ff:fe0f:42ce/10
test0: flags=120002000840<RUNNING,MULTICAST,IPv6,PHYSRUNNING> mtu 9000 index 6
        inet6 ::/0
igb@solaris:~$
```

So it’s a network!

```
igb@solaris:~$ ssh 192.168.141.100
The authenticity of host '192.168.141.100 (192.168.141.100)' can't be established.
RSA key fingerprint is de:32:e9:88:9e:2e:61:66:2e:24:a7:a7:4c:6c:c8:b7.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.141.100' (RSA) to the list of known hosts.
Password:
Last login: Thu Feb 12 13:20:03 2015
Oracle Corporation      SunOS 5.11      11.2    June 2014
igb@secondzone:~$
```


Solaris 11 64-bit: From Live
Applications
Places
System
Click to view your appointments and tasks
Add More Software
Start Here
Terminal
File Edit View Terminal Help
-gb@solaris:~$
-gb@solaris:~$
-gb@solaris:~$
-gb@solaris:~$
-gb@solaris:~$ dladm
LINK CLASS MTU STATE OVER
net0 phys 1500 up --
other0 etherstub 9000 unknown
test0 vnic 9000 up ether0
test1 vnic 9000 up ether0
-gb@solaris:~$ ipadm
NAME CLASS/TYPE STATE UNDER ADDR
lo0 loopback ok
lo0/v4 static ok -- 127.0.0.1/8
lo0/v6 static ok -- ::1/128
net0 ip ok --
net0/v4 dhcp ok -- 172.16.210.135/24
net0/v6 addrconf ok fe80::20c:29ff:fe0f:42ce/10
test0 ip ok --
test0/v4 static ok -- 192.168.141.101/24
-gb@solaris:~$
-gb@solaris:~$
-gb@solaris:~$
-gb@solaris:~$
Terminal

Exercises

- Read the manual pages for “ipf”, “ipfstat” and the configuration files (“man ipf”, “man ipfstat” and “man -s4 ipf”)
  - Not even Unix obsessives defend the manual page “section” nonsense
- You can then edit a firewall into /etc/ipf/ipf.conf
- Load it with ipf -Fa -f /etc/ipf/ipf.conf
- Flush it with ipf -Fa

For example

- Set up a firewall on secondzone to block access to the ssh server
  - You might want to use “flags S/SA”

Default Deny

```
block return-rst in log first level local1.info quick \
    on test0 proto tcp from any to any \
    flags S/SA head 102
pass in quick from any to any port = 22 keep state group 102
```

You can add more similar “pass” lines to permit more tcp protocols

How would you test that this is working correctly?

Tools

- Install nmap into the global zone
  - “sudo pkg install nmap”
- Read the manual page for it, and use it to see which ports are open on secondzone (you’ll probably have to quote secondzone’s IP number directly, or you can edit /etc/hosts)

Tools

- Also look back at previous lectures and see use of netstat to look for ports in state LISTEN, and read the manual page of pfiles to see how to look at individual processes

Exercise

- Write a default-deny firewall for Solaris which permits:
  - outbound DNS and the responses
  - inbound ssh
  - outbound TCP connection
  - Default Deny