Protecting Host from Net

Host Hardening, Default Services, Host Based Firewall, Patching, Backup

Fakrul (pappu) Alam
fakrul@bdhub.com

Acknowledgement

- Original slides prepared by Patrick Okui

Protecting Host from Net
Host / OS Hardening

OS Hardening: General Consideration

- Differs per operating system
  - OS X : make things work magically for users. Try to handle security issues in the background
  - Windows: users can not be trusted to make security related decisions in almost all cases.
  - Linux: varies by distribution:
    - Ubuntu: try like OS X to make things just work.
    - RedHat: include very useful tools but turned off by default
    - Slackware: users are experts – they'll figure it out.
- Changes with time

OS Hardening: General Consideration

- Define a personal usage profile and policy.
  - What hardware do you use?
  - What software tasks do you do on your computer?
  - Do the first two change when you travel?
  - What habits from the above two do you need to change to be more secure?
  - Decide if you *really* need VPN access to your network while travelling.

OS Hardening: General Consideration

- Install only the services and software you actually need.
- Uninstall or disable all software and services you do not use or need.
- Periodically actively scan your machine for vulnerabilities.
- Have as few user accounts on your systems as possible
- Protect your administrative account. Have a strong password, do not permit remote password based logins and do not log in as an administrator unless you need to do an administrative task.

Hardening: hardware and firmware

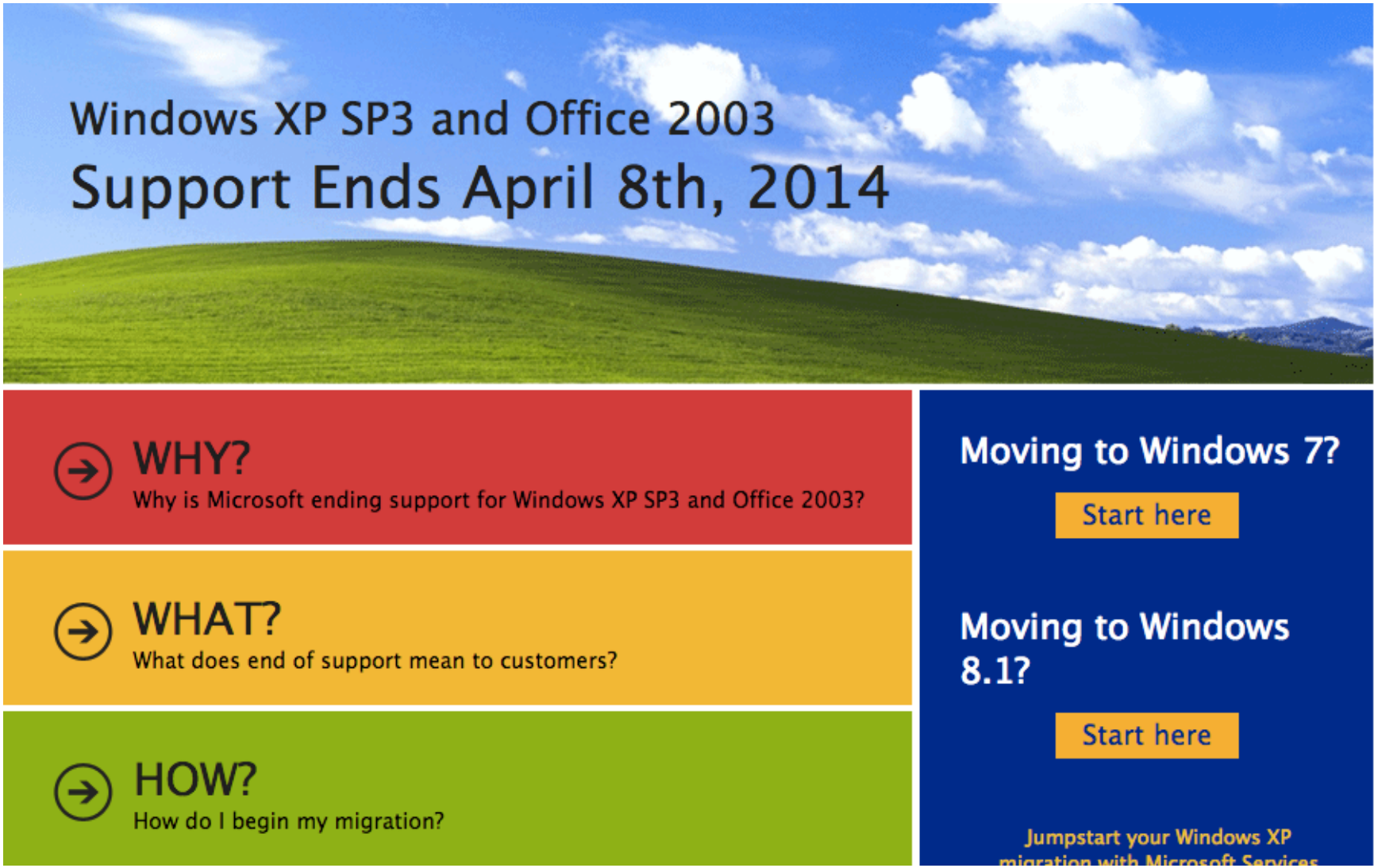
- Rule 1: all bets are off with physical access to your devices.
- Consider removing (from laptops mostly) hardware you never use – say bluetooth.
- Disable in BIOS or EFI or your operating system the hardware or features you can not remove physically.
  - wake-on-lan
  - Bluetooth discoverability
  - USB ports?

Hardening: hardware and firmware

- Cautiously turn on features that may be useful to you.
  - Fingerprint scanners
  - BIOS passwords not that useful
  - BIOS level encryption/locking of hard disks may not be portable
  - Privacy screens
  - Trusted aka Treacherous Computing?

Host Hardening
Windows

All Windows Lover



Host Hardening : Windows

- Which Windows operating system should I use
- What is a workgroup?
- What is Active Directory?
- What are local users and groups?
- What is Group Policy, and why is it so important?

Securing Windows: Best Practices

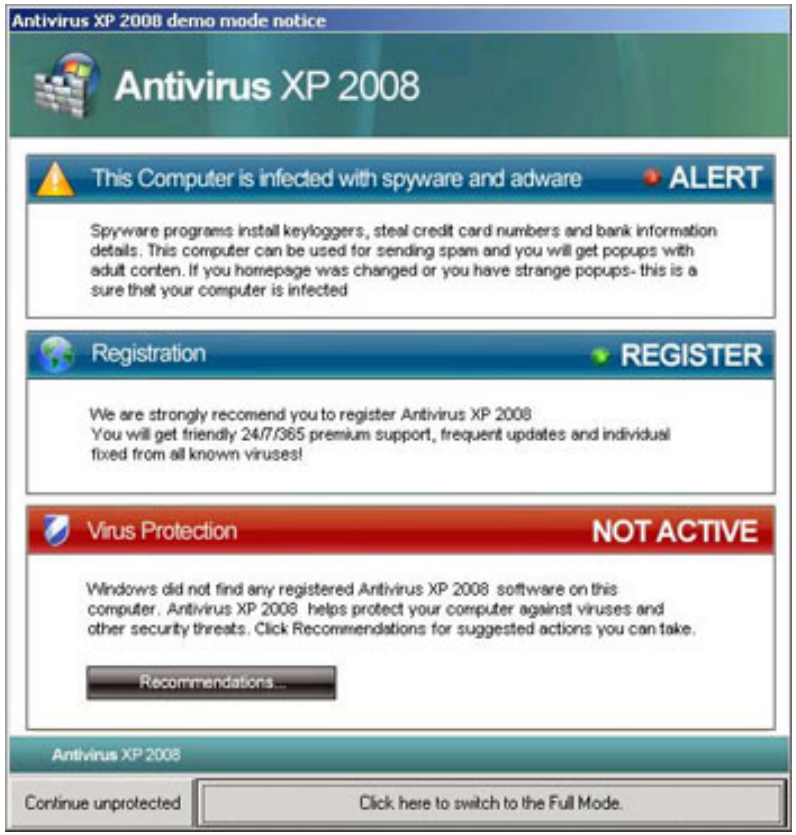
- Disable Guest Account
  - Start->Control Panel->UserAccounts
- Configure an update method to install patches
  - Start->Control Panel->System > Automatic Updates tab
- Disable unused system services
  - Start->Control Panel->Administrative Tools->Services

Securing Windows: Best Practices

- Verify the appropriate Local Security Settings
  - Start->Control Panel->Administrative Tools->Local Security Settings
- Check the Windows Firewall settings
  - Start->Control Panel->Network Connections and choose the network connection that corresponds to your Internet connection

Securing Windows: Best Practices

- Install an antivirus application
  - Lot of options. But watch out for fake one



Securing Windows: Best Practices

- Disable hiding of file extensions
  - Start->Control Panel->Appearance and Personalization->Folder Options->View->Advance Settings

Securing Windows: Best Practices

- Install more secure default applications:
  - Firefox or Chrome rather than Internet Explorer
  - Thunderbird or Claws instead of Outlook (unless you need exchange access)
  - A different MTA rather than Exchange for servers (unless you really need the groupware features)
  - Configure applications to at minimum warn when VBA extensions or macros want to run

Securing Windows: Best Practices

- For more
  - http://www.microsoft.com/security/default.aspx

Host Hardening
Linux

Securing Linux

- Available features differ by distribution.
  - Use as few different distributions in your environment as possible.
- Some distributions have optional scripts to "harden" your system e.g. SuSE, RedHat
- Using security distro like Kali Linux/Backtrack doesn't mean you are secure.

Securing Linux

- UNIX security model is based on permissions – ensure they are sane:
  - check for SUID/SGID applications
    - Some distros will include a nightly check for such applications.
  - Many distros have permission profiles that range from permissive to paranoid that essentially limit or permit people from/to run(ning) particular applications or accessing particular files like logs.
    - Run the periodic scripts that reset these permissions.

Securing Linux

- For servers, always pick the “minimal” installation to ensure as few packages as possible end up on the system by default.
- Use system tools to disable services that are required by other packages but do not need to be running. E.g some packages will not install if you do not have an SMTP server.

Securing Linux

- Periodic security checks:
  - Checksums of system files kept offline and checked against the running system
    - tripwire
    - fcheck
  - Periodic scans of the system.
    - nmap
    - Openvas
- Realtime checks
  - incrond /inotify – kernel feature to notify you as soon as a specified watchlist of inodes are changed.

Securing Linux: Best Practices

1. **Minimize Software to Minimize Vulnerability**
   - CentOS/Fedora

```
# yum list installed
# yum list packageName
# yum remove packageName
```

   - FreeBSD/Ubuntu

```
# dpkg --list
# dpkg --info packageName
# apt-get remove packageName
```

Securing Linux: Best Practices

2. **Keep Linux Kernel and Software Up to Date**
   - CentOS/Fedora

     ```
     # yum update
     ```

   - FreeBSD/Ubuntu

     ```
     # apt-get update && apt-get upgrade
     ```

Securing Linux: Best Practices

3. Use Linux Security Extensions

- SELinux
  - SELinux is an acronym for Security-enhanced Linux. It is a security feature of the Linux kernel.
  - It is designed to protect the server against misconfigurations and/or compromised daemons. It put limits and instructs server daemons or programs what files they can access and what actions they can take by defining a security policy.
  - Check http://selinuxproject.org/ for details.

Securing Linux: Best Practices

4. **User Accounts and Strong Password Policy**
   - A good password includes at least 8 characters long and mixture of alphabets, number, special character, upper & lower alphabets etc. Most important pick a password you can remember
   - Verify No Accounts Have Empty Passwords?

     ```
     # awk -F: '($2 == "") {print}' /etc/shadow
     ```

   - Lock all empty password account

     ```
     # passwd -l accountName
     ```

Securing Linux: Best Practices

5. **Disable root Login**
   - Never ever login as root user. You should use sudo to execute root level commands as and when required.

```
# vi /etc/ssh/sshd_config
-> PermitRootLogin no
# /etc/init.d/sshd restart
```

Securing Linux: Best Practices

6. **Physical Server Security**
   - Configure the BIOS and disable the booting from external devices such as DVDs / CDs / USB pen. Set BIOS and grub boot loader password to protect these settings.

Securing Linux: Best Practices

7. **Disable Unwanted Services**
   - Disable all unnecessary services and daemons (services that runs in the background).

     ```
     # chkconfig --list | grep '3:on'
     ```

   - To disable service, enter:

     ```
     # service serviceName stop
     # chkconfig serviceName off
     ```

Securing Linux: Best Practices

8. **Separate Disk Partitions**
   - Separation of the operating system files from user files may result into a better and secure system. Make sure the following filesystems are mounted on separate partitions:
     - `/usr`
     - `/home`
     - `/var and /var/tmp`
     - `/tmp`

Securing Linux: Best Practices

9. Secure OpenSSH Server

- Change SSH Port

```
# /etc/ssh/sshd_config
-> Port 420 #default port is 22
```

Securing Linux: Best Practices

10. TCP Wrappers

```
# vi /etc/hosts.allow
sshd:[2405:7600:0:6::250]/128,
202.4.96.250
# vi /etc/hosts.deny
ALL:ALL
```

Securing Linux: Best Practices

11. Check Listening Network Ports

```
# netstat –tulpn
```

```
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 0.0.0.0:27017           0.0.0.0:*               LISTEN      1073/mongod
tcp        0      0 127.0.0.1:3306          0.0.0.0:*               LISTEN      1094/mysqld
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      1770/nginx
tcp        0      0 0.0.0.0:28017           0.0.0.0:*               LISTEN      1073/mongod
tcp6       0      0 :::420                  :::*                    LISTEN      952/sshd
udp        0      0 0.0.0.0:1812            0.0.0.0:*                           18720/radiusd
udp        0      0 0.0.0.0:1813            0.0.0.0:*                           18721/radiusd
```

Securing Linux: Best Practices

12. SSH Banner

```
# vi /etc/issue.net
```

!!!! WARNING !!!!
You have accessed a restricted device.
All access is being logged and any unauthorized access
will be prosecuted to the full extent of the law.

vi /etc/ssh/sshd_config

Banner /etc/issue.net

Securing Linux: Best Practices

13. Review Log

/var/log/message – Where whole system logs or current activity logs are available

/var/log/auth.log – Authentication logs

/var/log/kern.log – Kernel logs

/var/log/cron.log – Crond logs (cron job)

/var/log/maillog – Mail server logs

/var/log/boot.log – System boot log

/var/log/mysqld.log – MySQL database server log file

/var/log/secure – Authentication log

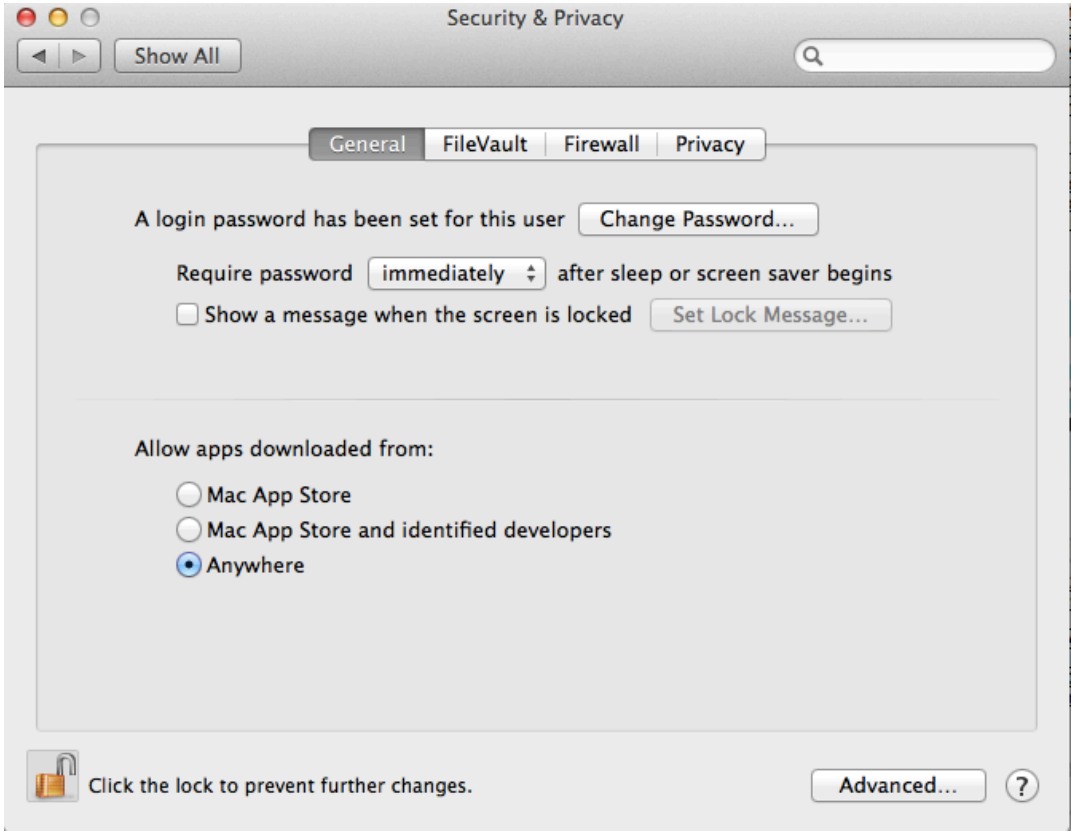
/var/log/utmp or /var/log/wtmp : Login records file

/var/log/yum.log: Yum log files

Host Hardening
Mac OSX

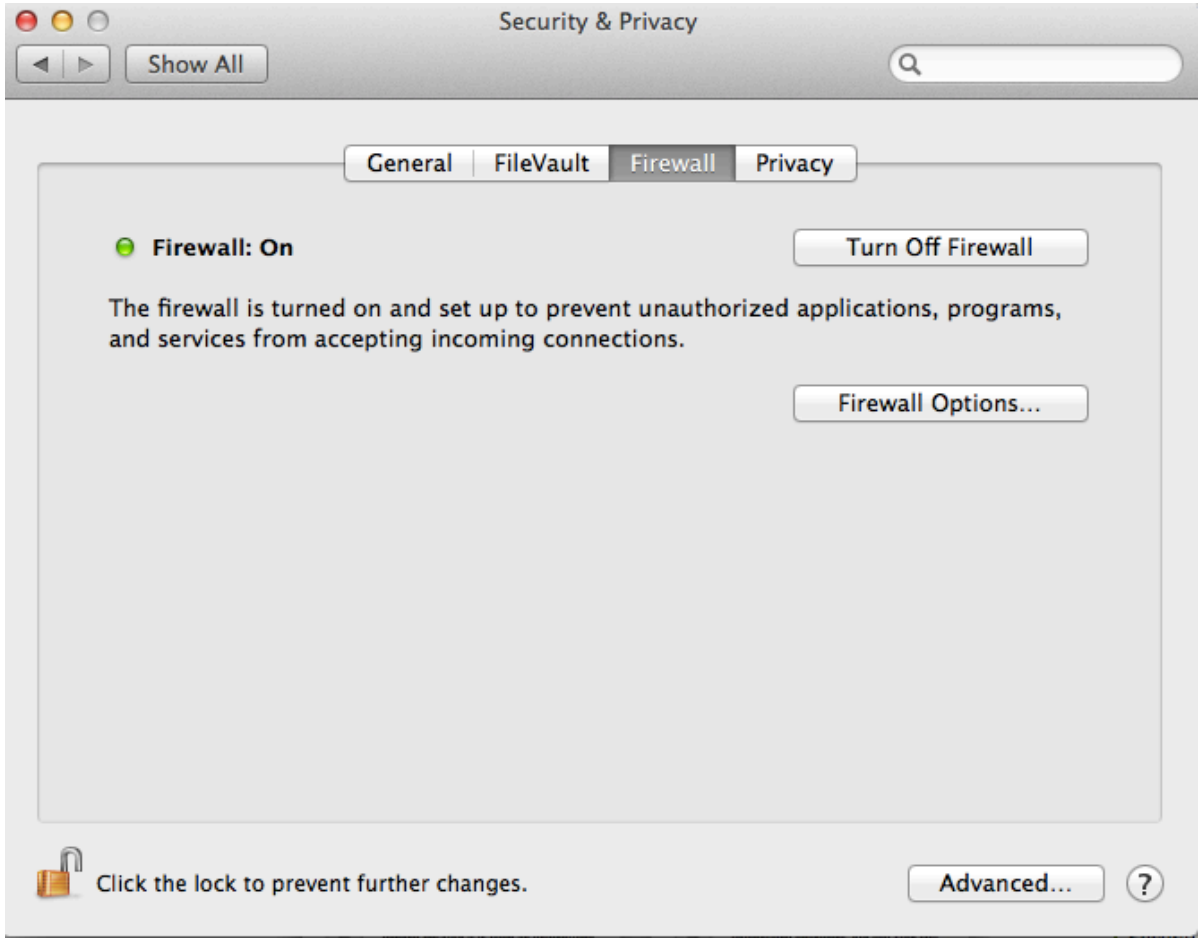
Securing Mac OSX: Best Practices

Allow apps installation from “Mac App Store and identified developers”



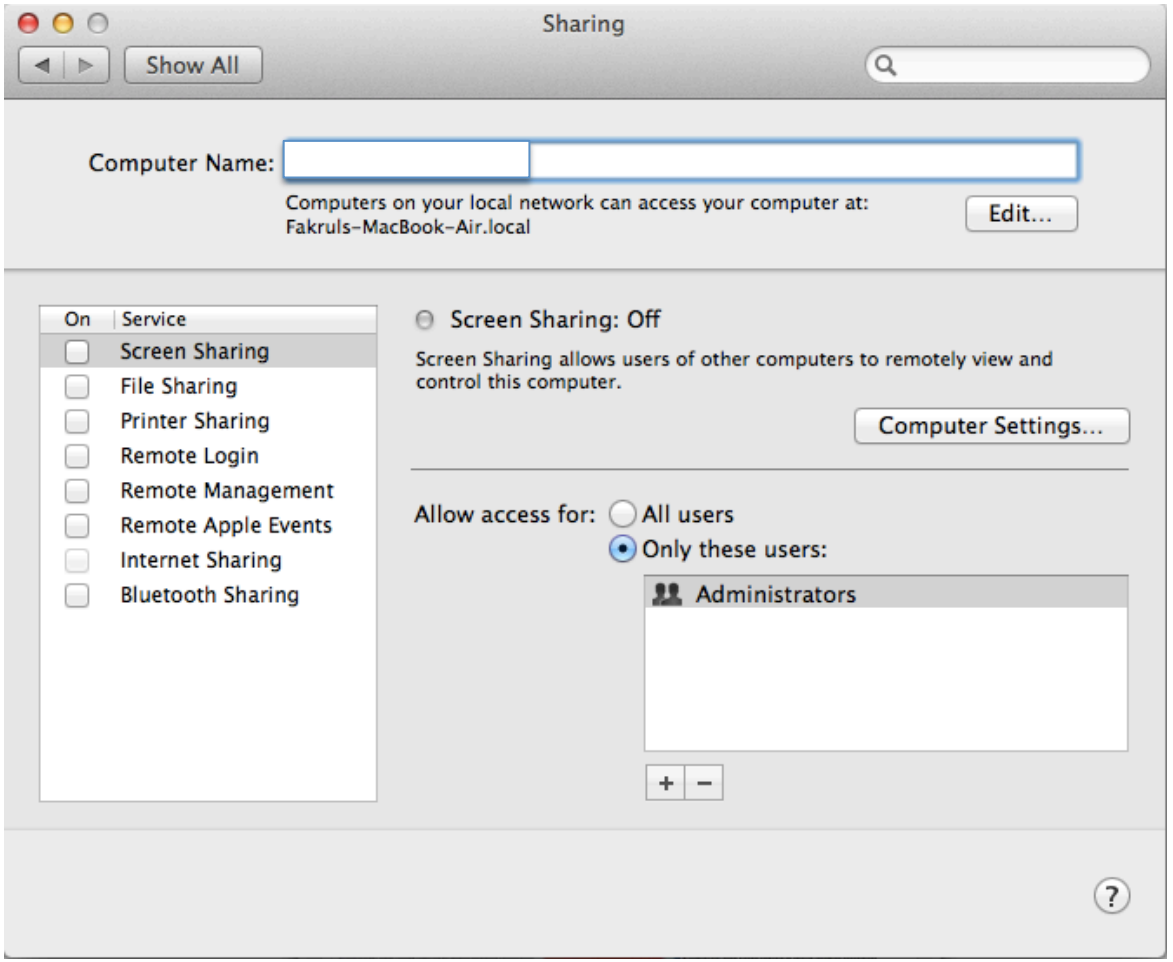
Securing Mac OSX: Best Practices

Enable Firewall



Securing Mac OSX: Best Practices

Disable unused services



Securing Mac OSX: Best Practices

Read and understand security guides at http://www.apple.com/support/security/guides/ (not updated for Lion or Mountain Lion)

Mac OS X v10.6 (Snow Leopard)

- Mac OS X Security Configuration Guide
- Mac OS X Server Security Configuration Guide

Mac OS X v10.5 (Leopard)

- Mac OS X Security Configuration Guide
- Mac OS X Server Security Configuration Guide

Mac OS X v10.4 (Tiger)

- Mac OS X Security Configuration Guide
- Mac OS X Server Security Configuration Guide

Mac OS X v10.3 (Panther)

- Client Security Configuration Guide
- Server Security Configuration Guide

Protecting Host from Net

Patch & Updates /Security Updates

Patch

“A patch is a small piece of software that is used to correct a problem with a software program or an operating system. Patches are often called "fixes." Service packs usually contain many different patches.”

Security Updates

Microsoft Security Updates

http://www.microsoft.com/security/default.aspx

Apple Security Updates

http://support.apple.com/kb/ht1222

Getting Security Updates

- Subscribe to application/host specific security updates.
- RSS Feed!!!
- Twitter Feedback!!!!
- Subscribe US-CERT Cyber Awareness System.
  - http://www.us-cert.gov/ncas

Getting Security Updates

NCCIC
NATIONAL CYBERSECURITY AND COMMUNICATIONS INTEGRATION CENTER
US-CERT
UNITED STATES COMPUTER EMERGENCY READINESS TEAM

National Cyber Awareness System:

Google Releases Google Chrome Update

01/28/2014 11:57 AM EST

Original release date: January 28, 2014

Google has released Google Chrome 32.0.1700.10
Frame to address multiple vulnerabilities. These vu
cause a denial of service or bypass intended securit

US-CERT encourages users and administrators to r
and follow best-practice security policies to determ

This product is provided subject to this Notificatior

NCCIC
NATIONAL CYBERSECURITY AND COMMUNICATIONS INTEGRATION CENTER
US-CERT
UNITED STATES COMPUTER EMERGENCY READINESS TEAM

National Cyber Awareness System:

TA14-017A: UDP-based Amplification Attacks

01/17/2014 03:22 PM EST

Original release date: January 17, 2014 | Last revised: February 09, 2014

Systems Affected

Certain UDP protocols have been identified as potential attack vectors:

- DNS
- NTP
- SNMPv2
- NetBIOS
- SSDP
- CharGEN
- QOTD
- BitTorrent
- Kad
- Quake Network Protocol
- Steam Protocol

Overview

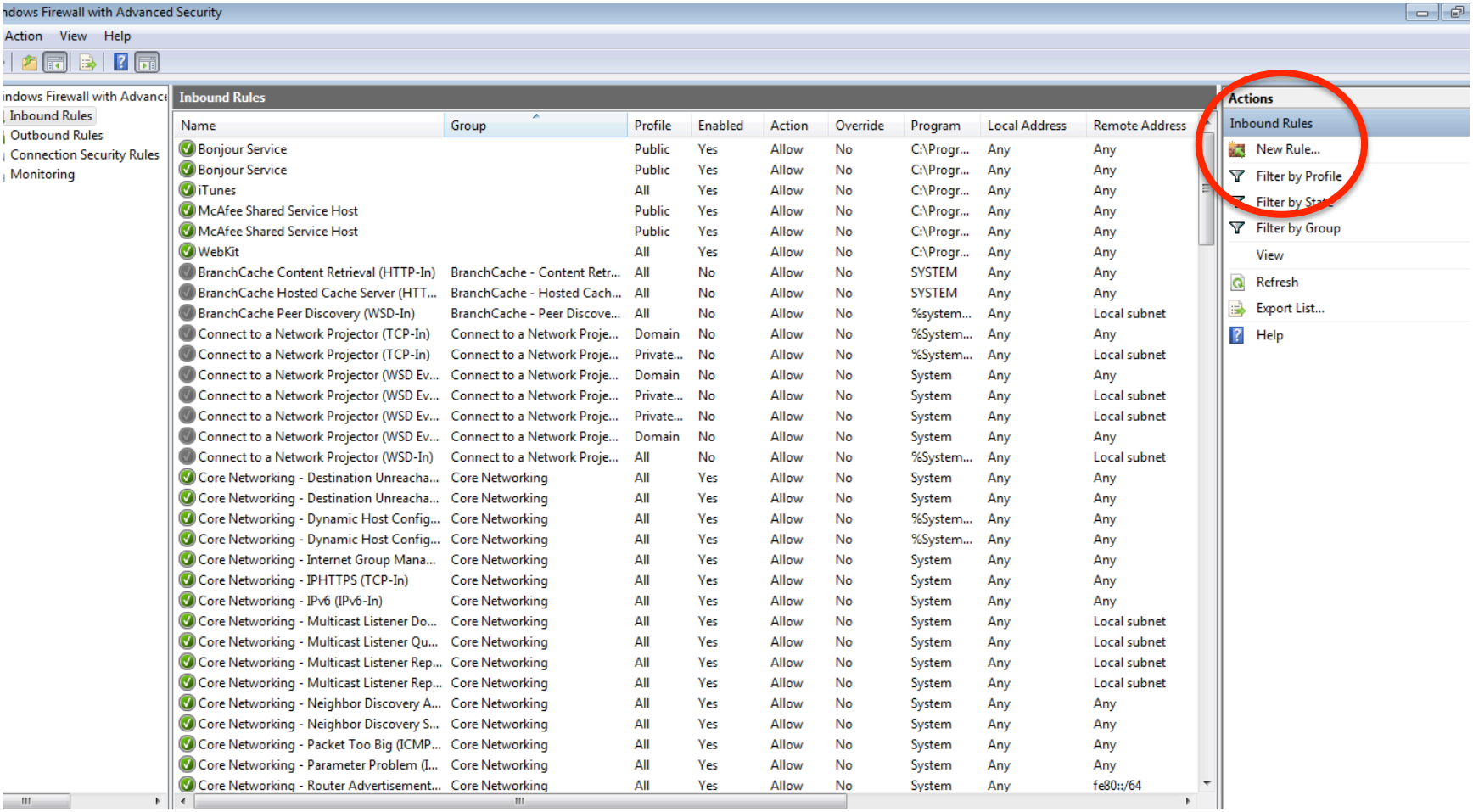
A Distributed Reflective Denial of Service (DRDoS) attack is an emerging form of Distributed Denial of Service (DDoS) that relies on the use of publicly accessible UDP servers, as well as

Protecting Host from Net
Host Based Firewall

Protecting Host from Net
Host Based Firewall
Windows

Host Based Firewall: Windows

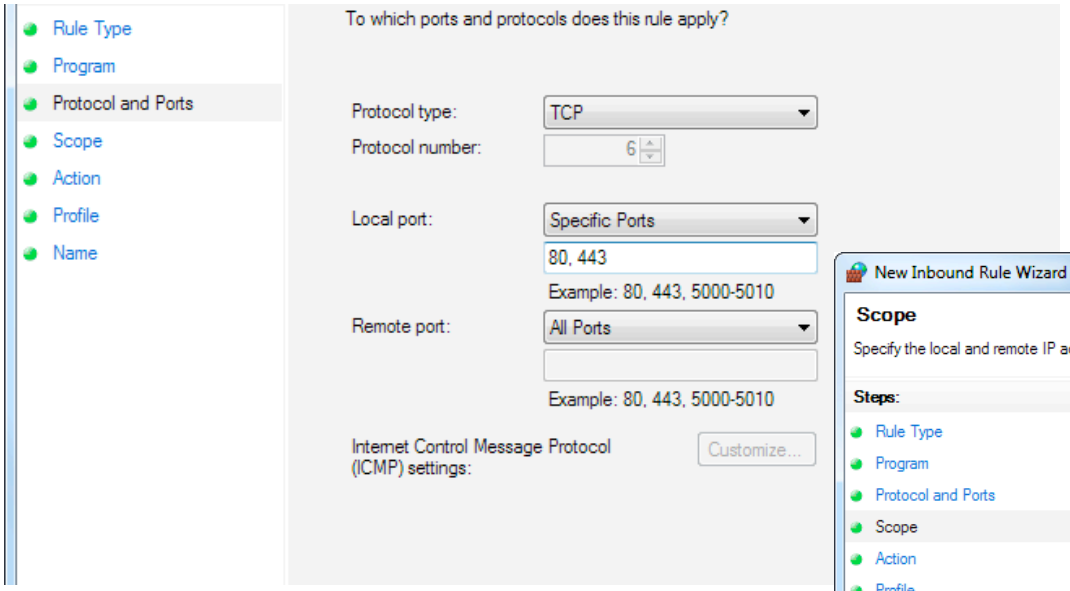
Host Based Firewall: Windows

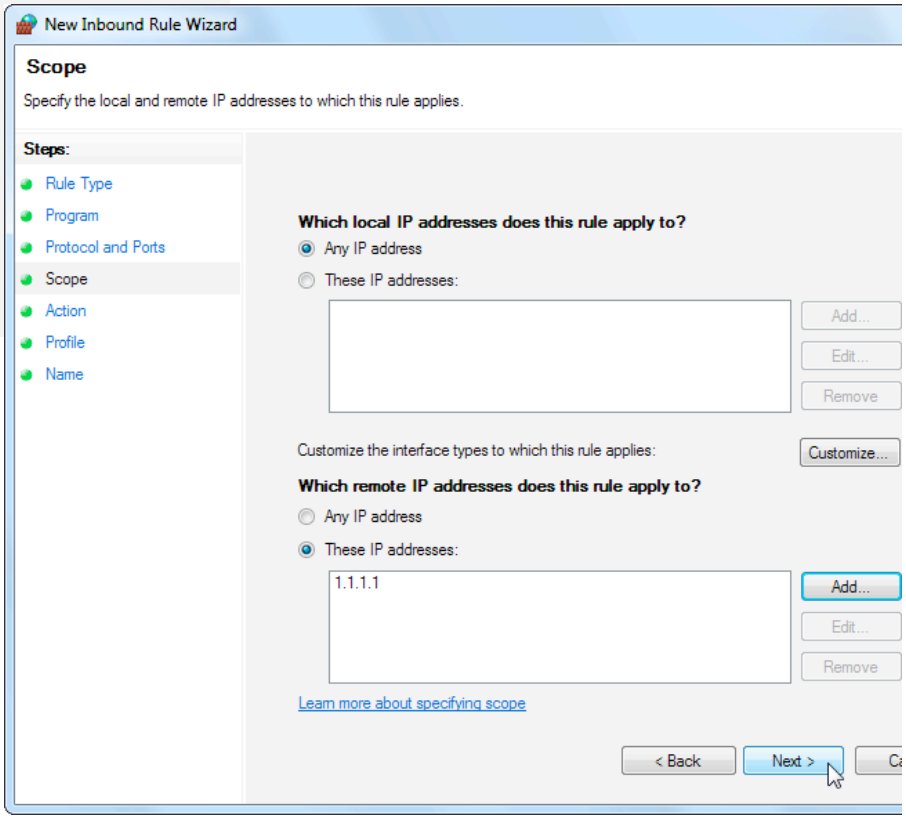


Host Based Firewall: Windows

- The Windows firewall offers four types of rules:
  - Program – Block or allow a program.
  - Port – Block or a allow a port, port range, or protocol.
  - Predefined – Use a predefined firewall rule included with Windows.
  - Custom – Specify a combination of program, port, and IP address to block or allow.

Host Based Firewall: Windows





Protecting Host from Net
Host Based Firewall
Linux

Host Based Firewall: IPTABLES

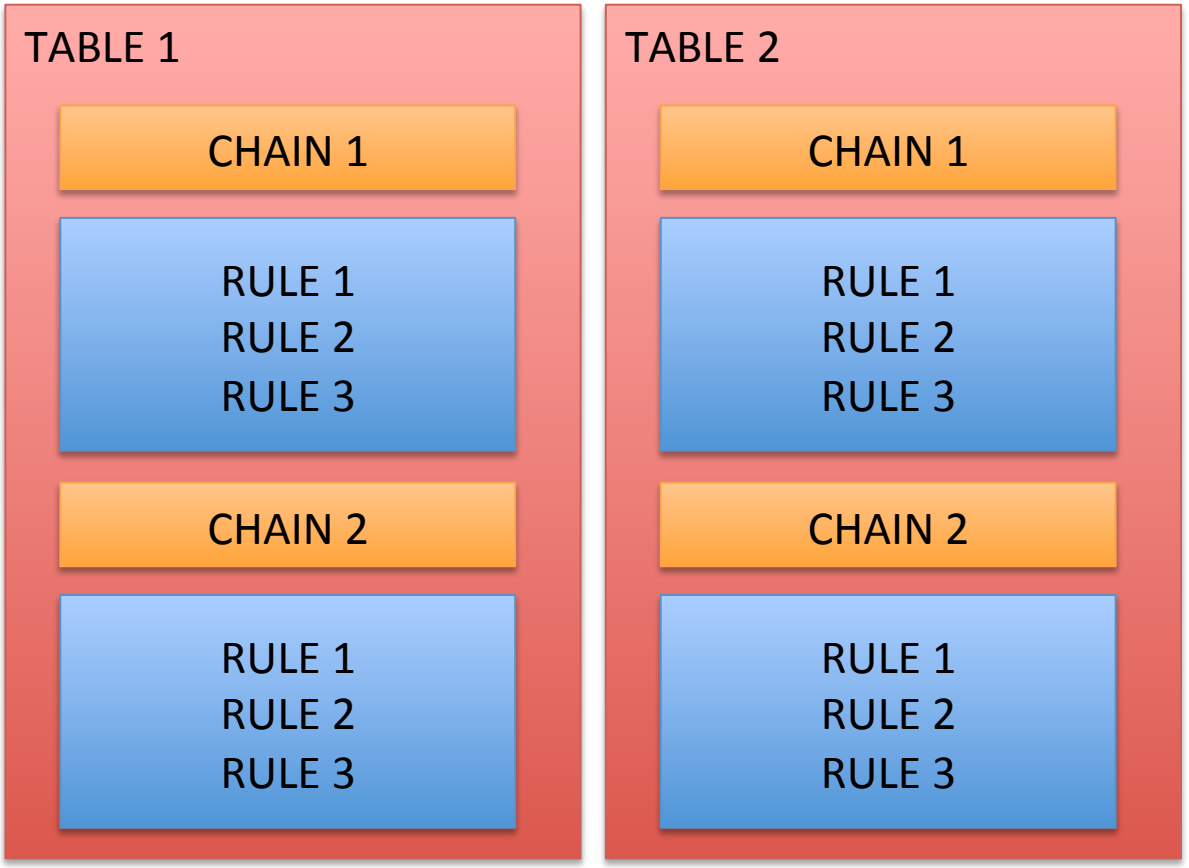
IPTABLES Structure

IPTABLES

>TABLES

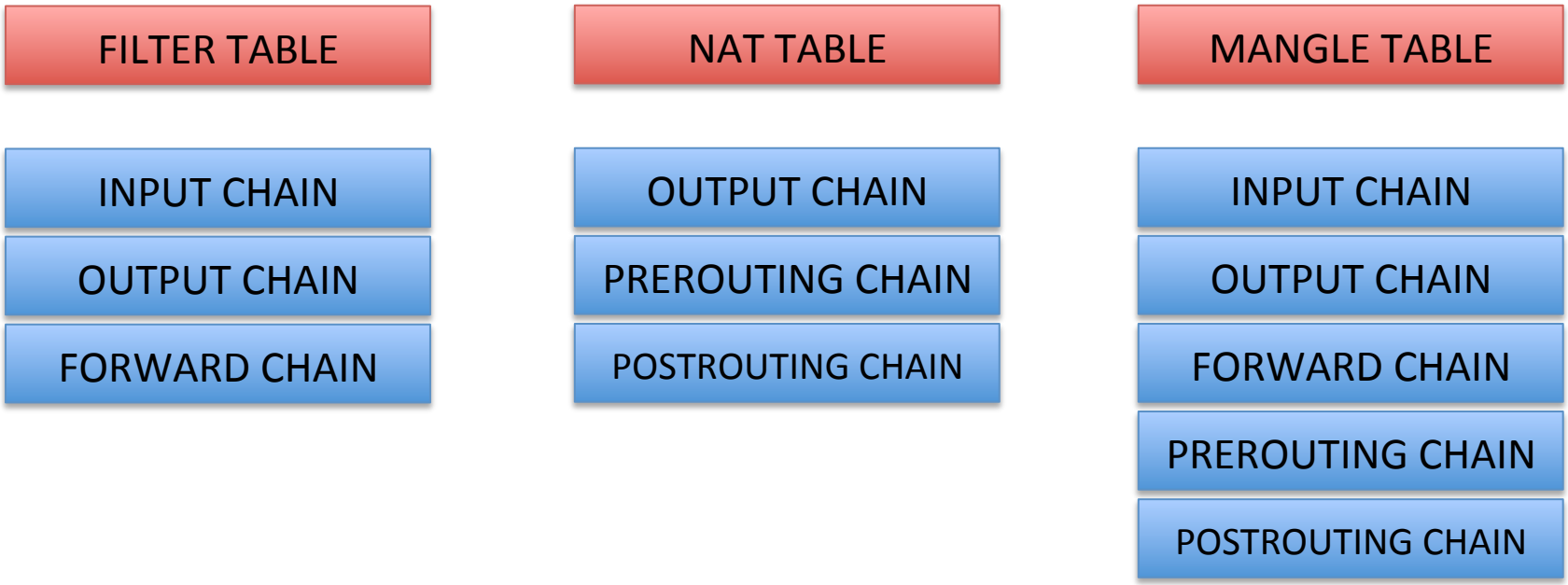
>CHAINS

>RULES



Host Based Firewall: IPTABLES

IPTABLES table & chains



Host Based Firewall: IPTABLES

IPTABLES Rules

- Rules contain a criteria and a target.
- If the criteria is matched, it goes to the rules specified in the target (or) executes the special values mentioned in the target.
- If the criteria is not matched, it moves on to the next rule.

Host Based Firewall: IPTABLES

Displaying the Status of Your Firewall

root@access /h/fakrul# iptables -L -n -v --line-numbers

```
Chain INPUT (policy ACCEPT 1885K packets, 417M bytes)
num   pkts bytes target     prot opt in     out     source               destination
1        0     0 ACCEPT     udp  --  virbr0 *       0.0.0.0/0            0.0.0.0/0           udp dpt:53
2        0     0 ACCEPT     tcp  --  virbr0 *       0.0.0.0/0            0.0.0.0/0           tcp dpt:53
3        0     0 ACCEPT     udp  --  virbr0 *       0.0.0.0/0            0.0.0.0/0           udp dpt:67
4        0     0 ACCEPT     tcp  --  virbr0 *       0.0.0.0/0            0.0.0.0/0           tcp dpt:67

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
num   pkts bytes target     prot opt in     out     source               destination
1          0      0 ACCEPT      all  --  *       virbr0  0.0.0.0/0            192.168.122.0/24      state
RELATED,ESTABLISHED
2        0     0 ACCEPT     all  --  virbr0 *       192.168.122.0/24     0.0.0.0/0
3        0     0 ACCEPT     all  --  virbr0 virbr0  0.0.0.0/0            0.0.0.0/0
4        0     0 REJECT     all  --  *      virbr0  0.0.0.0/0            0.0.0.0/0           reject-with
icmp-port-unreachable
5        0     0 REJECT     all  --  virbr0 *       0.0.0.0/0            0.0.0.0/0           reject-with
icmp-port-unreachable

Chain OUTPUT (policy ACCEPT 1843K packets, 147M bytes)
num   pkts bytes target     prot opt in     out     source               destination
```

Host Based Firewall: IPTABLES

Displaying the Status of NAT Table

root@access /h/fakrul# iptables -t nat -L -n -v

```
Chain PREROUTING (policy ACCEPT 867 packets, 146K bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 DROP       all  --  vlan2  *       0.0.0.0/0            192.168.1.0/24

Chain POSTROUTING (policy ACCEPT 99 packets, 6875 bytes)
 pkts bytes target     prot opt in     out     source               destination
    0     0 MASQUERADE  all  --  *      vlan2   0.0.0.0/0             0.0.0.0/0

Chain OUTPUT (policy ACCEPT 99 packets, 6875 bytes)
 pkts bytes target     prot opt in     out     source               destination
```

Host Based Firewall: IPTABLES

Stop / Start / Restart Firewall

- CentOS / RHEL / Fedora Linux:

```
# service iptables stop
# service iptables start
# service iptables restart
```

- Delete Firewall Rules

```
# iptables -L INPUT -n --line-numbers
# iptables -L OUTPUT -n --line-numbers
# iptables -L OUTPUT -n --line-numbers | less
# iptables -L OUTPUT -n --line-numbers | grep 192.168.1.1
```

Host Based Firewall: IPTABLES

Save Firewall Rules

- CentOS / RHEL / Fedora Linux:

```
# service iptables save
```

- For other distro:

```
# iptables-save > /root/iptables_rules
```

- Restore Firewall Rules:

```
# iptables-restore < /root/iptables_rules
```

Host Based Firewall: IPTABLES

Sample Rules

```
# iptables -A INPUT -i eth1 -s 192.168.0.0/24 -j DROP
# iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j DROP

# iptables -A INPUT -i eth1 -p tcp --dport 80 -j ACCEPT

# iptables -A OUTPUT -d 75.126.153.206 -j DROP

# iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j LOG --log-prefix "IP_SPOOF A: »
# iptables -A INPUT -i eth1 -s 10.0.0.0/8 -j DROP
```

Host Based Firewall: IPTABLES

Block or Allow ICMP Ping Request

```
# iptables -A INPUT -p icmp --icmp-type echo-request -j DROP
# iptables -A INPUT -i eth1 -p icmp --icmp-type echo-request -j DROP
```

Host Based Firewall: IPTABLES

```
# iptables -L
   – Chain INPUT (policy ACCEPT)
   – Chain FORWARD (policy ACCEPT)
   – Chain OUTPUT (policy ACCEPT)

# iptables -P INPUT DROP
# iptables -P OUTPUT DROP
# iptables -P FORWARD DROP
```

Host Based Firewall: IPTABLES (Bogon Filter)

```
# iptables -A INPUT -i eth0 -s 10.0.0.0/8 -j LOG --log-prefix "IP DROP SPOOF "
# iptables -A INPUT -i eth0 -s 172.16.0.0/12 -j LOG --log-prefix "IP DROP SPOOF "
# iptables -A INPUT -i eth0 -s 192.168.0.0/16 -j LOG --log-prefix "IP DROP SPOOF "
# iptables -A INPUT -i eth0 -s 224.0.0.0/4 -j LOG --log-prefix "IP DROP MULTICAST "
# iptables -A INPUT -i eth0 -s 240.0.0.0/5 -j LOG --log-prefix "IP DROP SPOOF "
# iptables -A INPUT -i eth0 -d 127.0.0.0/8 -j LOG --log-prefix "IP DROP LOOPBACK "
# iptables -A INPUT -i eth0 -s 169.254.0.0/16  -j LOG --log-prefix "IP DROP MULTICAST "

# iptables -A INPUT -m tcp -p tcp --dport 80 -j ACCEPT
# iptables -A INPUT -s 103.12.179.0/24 -m state --state NEW -p tcp --dport 22 -j ACCEPT
# iptables -D INPUT -s 103.12.176.0/22 -m state --state NEW -p udp --dport 123 -j ACCEPT
```

Protecting Host from Net

Backups

Backups General

- Know what you are going to backup.
- For servers, this needs to be centralized where as for end user devices this will usually need to be individual systems.
- Test recovery procedure.
- Incremental VS full backups.
- For personal backups, consider what happens when you travel.

Backups General

- Automate the backups as much as possible to avoid people forgetting.
- Consider encrypted contents: your backups must have at minimum the same encryption level as the source data.
- For personal backups, the backup disks will need to be kept separate from the machine – or you may lose both.

Backups: Windows

- Inbuilt windows backup.
- Needs to be setup – does not seem to exist on starter or home editions.
  - Control Panel->System and Security ->Backup and Restore
- Recovery is non trivial.
- NTFS has the ability to create snapshots on an external drive.
- Other option is things like dropbox/Google Drive!!!
- Software on some external drives comes with “one touch” backups.
  - Some with hardware encryption on the external drive.

Backups: Linux

- Host based backups are more or less same software options as server based network backups.
- Some distributions automate backups to external drives
  - https://wiki.ubuntu.com/HomeUserBackup
- You can also use home made scripts to use SSH to copy critical data to a remote server

Backups: OS X

- System events make it easy to automatically launch backups as soon as a particular disk is plugged in.
- Options for remote backup are the same as for Linux/Unix
- From 10.5 all systems come with “time machine”
  - Mixes full and incremental backups.

Backups: OS X

- Dead easy to setup and comes with the OS.
- Issues with encrypted home directories et al.

- Carbon Copy Cloner
  - Can create a bootable copy.
  - Can also do both incremental and full backups