CSI3660 – System Administration

Prof. Fredericks

Services Cont'd and Security

Outline

- Email
- Security
- Firewalls
- Monitoring Intrusions

Email Services

- Various email protocols exist,
  - SMTP, ESMTP, POP, and IMAP
- An email server looks up the name of target email server in domain's MX records
  - Stored on public DNS server
  - Resolves target email server name to IP address using public DNS server
- Daemons and system components rely on email to send important information to the root user

Types of Email Service (Sending)

- SMTP
  - Standard for sending email
  - Send messages to mail server
    - Relayed to destination

- ESMTP
  - SMTP, but adds additional features
    - Authentication
    - Security
    - Etc.

Types of Email Service (Receiving)

- **POP**
  - Downloads email locally
  - Advantages
    - Access offline
    - Faster for search
  - Disadvantages
    - Mail lost if workstation crashes
    - Only accessible on local machine
- **IMAP**
  - Leaves mail on mailserver
  - Advantages
    - Access anywhere
    - Easier to backup
  - Disadvantages
    - Requires server space
    - Slower read

Email Services

- Most common email daemon used on modern Linux systems is Postfix
  - Configured by default to accept email on TCP port 25 and route to appropriate user
  - To check email, use the `mail` command
- The /etc/aliases file contains other email names used to identify different users on the system
  - If that file is edited, run the `newaliases` command in order to rebuild the aliases database

Email Services

Line	Change
`mydomain = sample.com`	Sets the e-mail domain name; changes to desired name
`myorigin = $mydomain`	Sets local access to the domain name
`inet_interfaces = all`	Configures Postfix to listen for e-mail on all interfaces
`mydestination = $myhostname, localhost.$mydomain, localhost, $mydomain`	Configures destination domain for e-mail
`mynetworks_style=class`	Trusts e-mail from computers on the local network

Table 13-4: Sample lines in /etc/postfix/main.cf to modify or add when configuring Postfix

Demo

- Play around with configuring a server that can send mail
- Postfix already installed…

```
$ service postfix start

$ mail –s “<SUBJECT>” <email to send to>

<message body>

.
```

Change hostname

- FYI, this is just for pure fun…you can't receive email here
  - (Legitimate sending/receiving requires a fully-qualified domain name)
- Update /etc/postfix/main.cf
  - myhostname = csi3660.com
  - myorigin = $myhostname
- Restart Postfix service

BUT we caaaan use other services!

- We can…hook up our OU email accounts to the server!

```
# yum install postfix mailx cyrus-sasl
cyrus-sasl-plain
```

- Then configure your server to use the proper authentication
  - Create/Edit /etc/postfix/sasl_passwd
  - And add:

[smtp.gmail.com]:587 <username>@oakland.edu:<password>

Note that your password is stored here in **clear text – horribly insecure!!**

https://www.howtoforge.com/tutorial/configure-postfix-to-use-gmail-as-a-mail-relay/

Gmail cont'd. – Configure Postfix

- Edit /etc/postfix/main.cf

Add/modify lines to be like this:

```
relayhost = [smtp.gmail.com]:587

smtp_use_tls = yes

smtp_sasl_auth_enable = yes

smtp_sasl_security_options =

smtp_sasl_password_maps =
hash:/etc/postfix/sasl_passwd

smtp_tls_CAfile = /etc/ssl/certs/ca-bundle.crt
```

Gmail cont'd – Hash password

```
# postmap /etc/postfix/sasl_passwd
```

And restart postfix

```
# service postfix restart
```

And update Google to allow this to happen:

https://support.google.com/accounts/answer/6010255

Then, send a message!

$ mail –s “Subject” destination@email.com

- Hit enter again after blank line to ignore the CC field
- enter body text after that
- CTRL+d send email, CTRL+c cancels

Security

- https://www.youtube.com/watch?v=8oI_laHhGjE
- https://vimeo.com/25118844
- https://www.youtube.com/watch?v=bs0xswK0eZk

What are some steps you could take?

Write down **3** things that you could do

Security

- Systems typically made available across networks such as the Internet
  - More prone to security loopholes and attacks
- To protect Linux systems, you should
  - Improve local and network security
  - Understand how to detect intruders who breach the system

Securing the Local Computer

- Limit physical access
  - Prevent malicious users from accessing files by directly booting the computer with their own device
- Server closet
  - Secured room to store servers
- Remove CD and DVD drives from workstations in public areas
- Ensure BIOS prevents booting from USB ports
  - Set system BIOS password

Securing the Local Computer

- Limit access to graphical desktops and shells
  - Lock screen using GNOME or KDE while away

- Exit command-line shell before leaving computer
  - `nohup` command
    - Ignore hangup signal (SIGHUP)
    - Prevents background processes from being killed when parent shell is killed or exited
  - `screen` command
    - Keeps bash shell executing in background

Securing the Local Computer

- Minimize root user's time logged in
  - `su (switch user)` command
    - Switch current user account to another
- `sudo` command
  - Perform commands as another user if you have the rights to do that listed in /etc/sudoers file
  - Using `sudo` over `su` decreases the chance that an unattended computer is wide open for passersby

Protecting Against Network Attacks

- As long as networks services exist on a computer
  - Possibility that hackers can manipulate a network service by interacting with it in unusual ways
- Buffer overrun
  - Program information for a network service can be altered in memory
  - Altering how the network service operates

Network Security Essentials

- Minimize number of running network services
- `nmap (network mapper)` command
  - (Not installed locally)
  - Scans ports on network computers to determine what network services are running
  - $ sudo nmap –sV –O –v localhost
    - -sV : probe open ports
    - -O : OS detection
    - -v : verbose
- Ensure that services that are not needed are not automatically started when entering different runlevels

Network Security Essentials

- Ensure network service daemons for essential services are not run as root user when possible
  - Root has access to *everything*
- Ensure that the shell listed in /etc/passwd for daemons is set to an invalid shell
  - E.g., /sbin/nologin
  - Hacker will not be able to get BASH shell
- Keep services up to date
  - Generally include fixes for known network attacks

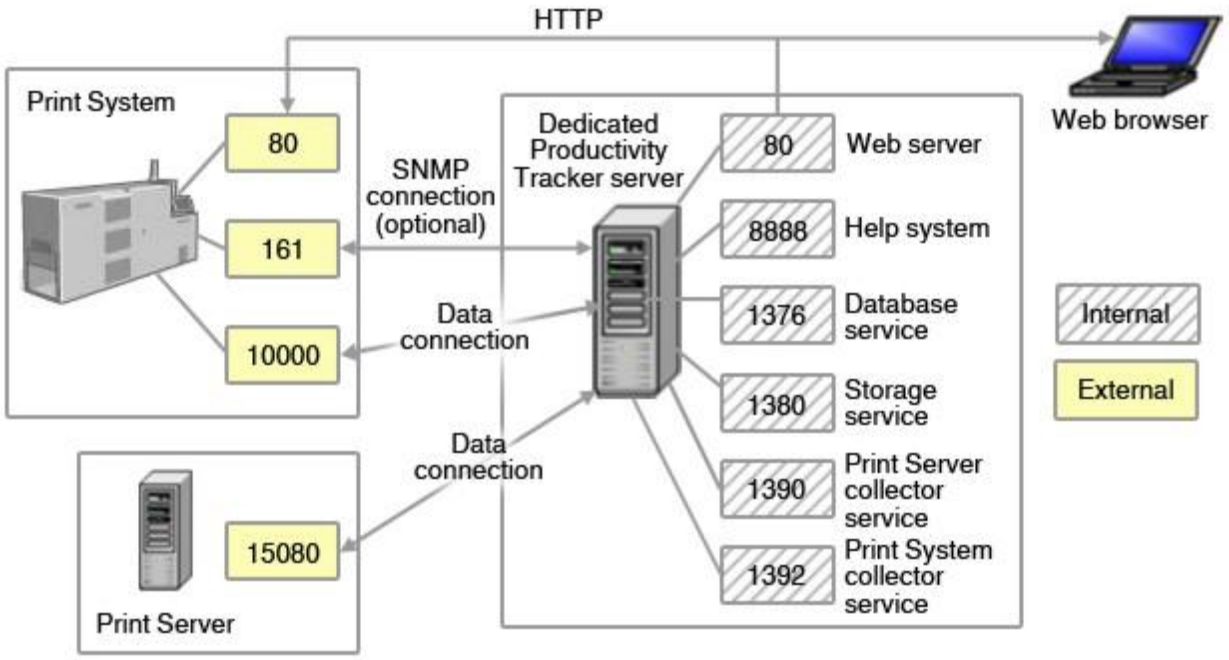
Network Security Essentials

- TCP wrapper
  - Program that can start a network daemon
  - Checks /etc/hosts.allow and /etc/hosts.deny files before starting a network daemon
  - Allows you to restrict which hosts can access the network service
    - Allow:
      - ALL: .csi3660.com
        - Only allow connections from csi3660.com
    - Deny:
      - ALL: ALL
        - Only allow connections from hosts.allow file
- Carefully examine permissions on files and directories associated with system and network services

Firewalls

- What is a port?
  - Point of network communication between devices
- Programs communicate via sockets
  - IP address + port
  - Apache: 141.210.25.98**:80**

Firewalls


HTTP
Web browser
Print System
80
161
10000
SNMP connection (optional)
Data connection
Data connection
Dedicated Productivity Tracker server
80
Web server
8888
Help system
1376
Database service
1380
Storage service
1390
Print Server collector service
1392
Print System collector service
Internal
External
15080
Print Server

Configuring a Firewall (Terms)

- Chains
  - Specify general type of network traffic to apply rules to
  - Three chain types:
    - INPUT
      - Incoming packets to computer
    - FORWARD
      - Packets passing through computer (routed)
    - OUTPUT
      - Outgoing packets from computer
- Rules
  - Match network traffic to be allowed or dropped
  - Added to each chain
  - E.g., drop all packets from specific IP address

Configuring a Firewall

- `iptables` command
  - Creates rules for each chain
  - Can also be entered manually to /etc/sysconfig/iptables
    - Restart iptables service
- Rules can be based on:
  - Source IP address
  - Destination IP address
  - Protocol used (TCP, UDP, ICMP)
  - Packet status

iptables Options

Option	Description
`-s address`	Specifies the source address of packets for a rule.
`-d address`	Specifies the destination address of packets for a rule.
`-sport port#`	Specifies the source port number for a rule.
`-dport port#`	Specifies the destination port number for a rule.
`-p protocol`	Specifies the protocol type for a rule.
`-i interface`	Specifies the input network interface.
`-o interface`	Specifies the output network interface.
`-j action`	Specifies the action that is taken for a rule.
`-m match`	Specifies a match parameter that should be used within the rule. The most common match used is `state`, which creates a stateful packet filtering firewall.
`-A chain`	Specifies the chain used.
`-L chain`	Lists rules for a certain chain. If no chain is given, all chains are listed.
`-P policy`	Specifies the default policy for a certain chain type.
`-D number`	Deletes a rule for a chain specified by additional arguments. Rules start at number 1.
`-R number`	Replaces a rule for a chain specified by additional arguments. Rules start at number 1.
`-F chain`	Removes all rules for a certain chain. If no chain is specified, it removes all rules for all chains.

iptables

- Firewall allows or blocks traffic through system ports
  - SSH: 22, MYSQL: 3306, etc.

$ iptables -A INPUT -s 192.168.0.4 -j ACCEPT

- -A: Specify INPUT chain
- -s: "Trusted" IP address
- -j: ACCEPT packets to local machine

$ service iptables save

- Save changes to /etc/sysconfig/iptables
- Overwrites file – removes any formatting!

Add manually to /etc/sysconfig/iptables

-A INPUT –s 192.168.0.4 –j ACCEPT

iptables

- Open a port

#MySQL

-I INPUT -p tcp --dport 3306 -m state --state NEW,ESTABLISHED -j ACCEPT

-I OUTPUT -p tcp --sport 3306 -m state --state ESTABLISHED -j ACCEPT

(NEW → new packets / ESTABLISHED → part of ongoing connection)

- Block a range of ports

-A INPUT -p tcp --match multiport --dports 1024:3000 -j DROP

firewalld (firewall-cmd)

Well that sure was fun.

Life is much easier now!

```
firewall-cmd
    --add-port=<port>/<tcp|udp>
    --add-service=<service>
    --permanent

firewall-cmd --reload
```

firewall-cmd

Useful commands (flags)

--list-services

--list-ports

https://www.techrepublic.com/article/how-to-manage-zones-on-centos-7-with-firewalld/

SELinux

Configuring SELinux

- SELinux: Security Enhanced Linux
  - Series of kernel patches and utilities created by NSA
    - Enforces role-based security
  - Enabled by default in several distributions
    - Including Scientific / CentOS
- To enable/disable:
  - Edit /etc/selinux/config file
    - SELINUX = enforcing
    - SELINUX = permissive
    - SELINUX = disabled

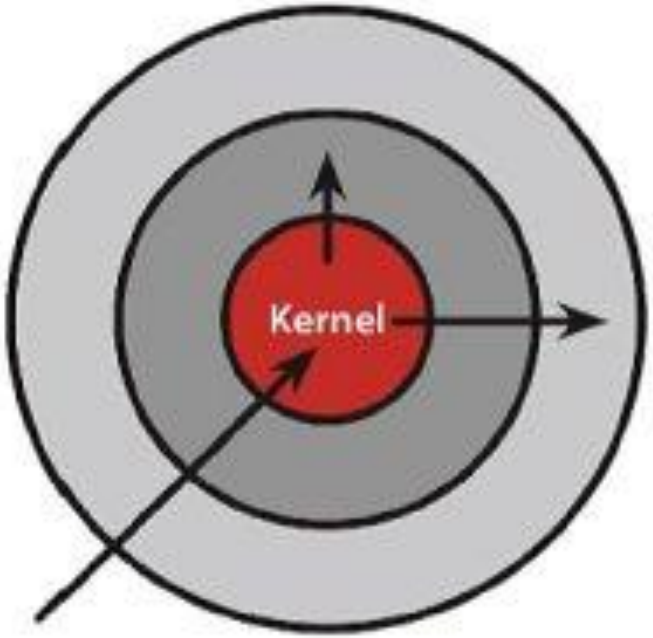
Configuring SELinux

- Select an SELINUX policy
  - /etc/selinux/ config:
    - SELINUXTYPE = targeted
      - Targeted processes protected
    - SELINUXTYPE = strict
      - Full protection on all processes
- After enabling SELinux,
  - **Reboot** to relabel the system for the changes to take effect
- `sestatus` command
  - Use to view current SELinux status

SELinux

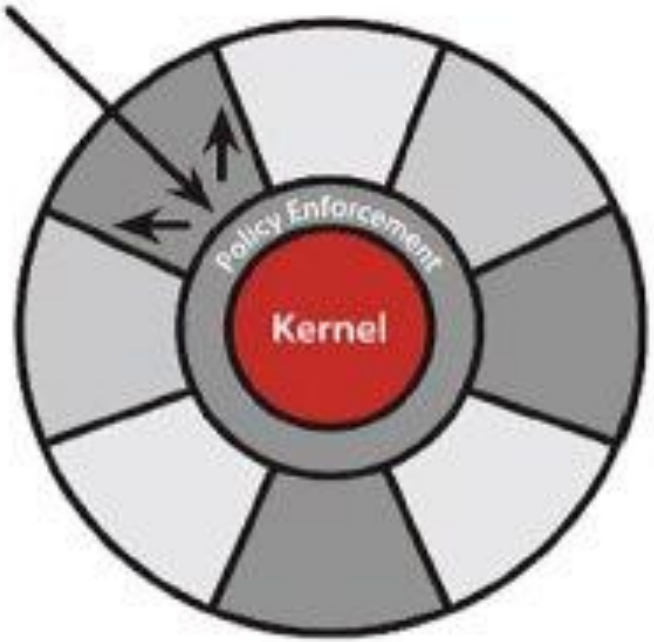
- Provides Mandatory Access Control (MAC)
  - Contrasts to standard Discretionary Access Control (DAC) used normally by Linux
- DAC
  - Executing process has user's permissions when running
    - E.g., root started a process, that process has root permissions…
- MAC
  - Defines access and rights of all users, processes, applications, and files on system
  - Helpful for servers, but a headache to update

DAC vs MAC



Discretionary Access Control

Once a security exploit gains access to priveleged system component, the entire system is compromised.



Mandatory Access Control

Kernel policy defines application rights, firewalling applications from compromising the entire system.

SELinux

- SELINUX = enforcing
  - System running with enhanced security
- SELINUX = permissive
  - System prints warnings but does not enforce them
- SELINUX = disabled
  - System essentially not running SELinux

- SELINUXTYPE=targeted
  - Enforces MAC on dhcpd, httpd, nptd, etc.
- SELINUXTYPE=strict
  - Enforces MAC on all daemons

Common SELinux Commands

- View process context

```
$ ps axZ | grep httpd
```

  - httpd_t type domain

- Show file context

```
$ ls -Z
```

- Access only allowed between same types
  - httpd_t cannot access user_home_t
  - Need to change context to allow

Common SELinux Commands

- Change context

  `$ chcon -t samba_share_t <file>`

  - Allow file to be shared by Samba

- Set SELinux Boolean

  `$ setsebool -P httpd_can_network_connect_db on`

  - Allow httpd to connect to network databases

  - allow_ftpd_anon_write
    - Allow ftpd to upload files to directories labeled public_content_rw_t
  - allow_user_mysql_connect
    - Allow user to connect to mysql socket

  - Full list: https://wiki.centos.org/TipsAndTricks/SelinuxBooleans

Detecting Intrusions

- UNIX security model (without SELinux) has a bit of a flaw…

Detecting Intrusions

- Reliance on superuser security model
  - Processes running with superuser privileges

- If you can coopt a SUID process….

Example

- Flaw in /etc/fingerd
  - Finger service over network
  - Displays information about users

- Possibility of buffer overrun
  - Read in text from standard input
  - No check on length of data read (512 bytes expected…more provided)
  - Overflowed buffer – caused fingerd to execute a shell
    - Shell has root privileges…

Example

- Issue was with C `gets` command
  - All distributions patched their programs to use `fgets`, which allows for size check

- However…
  - sprintf() and strcpy() commands became popular
  - New vulnerability found years later
    - sprint() and strcpy() can be called without boundary checking…
    - Patch again!

Detecting Intrusion

- Log files can contain information or irregularities indicating an intrusion
  - Review contents of log files in /var/log associated with network services
  - At minimum, review system log files associated with authentication
- Pluggable Authentication Module (PAM)
  - Handles authentication requests by network applications
  - PAM logs information to the journald database or to a log file in /var/log directory

Detecting Intrusion

- Check /var/log/wtmp log file
  - Lists users who receive BASH shells
  - Use `who` command to view the file
- `lsof` (list open files) command
  - Lists files that are currently being edited
- Periodically search for files that have SUID bit set
  - `$ sudo find / -user root -perm -4000 -print > SUIDfiles.txt`
- Tripwire
  - Monitors important files and directories
  - One of the more popular host-based IDSs

Intrusion Detection System

- IDS
  - Active process used to detect intruders on a Linux system
  - Knowledge-based
    - Database of common attacks
    - Warn sysadmin early
  - Behavioral-based
    - Monitor resource usage
  - Host-based
    - Most comprehensive
    - Detection on each host
  - Network-based
    - Sends network packets to single device
    - Less comprehensive – unreliable with many hosts

Detecting Intrusion

Name	Description
Advanced Intrusion Detection Environment (AIDE)	An alternative to tripwire that has added functionality for checking the integrity of files and directories.
Integrity Checking Utility (ICU)	A PERL-based program that is designed to work with AIDE to check the integrity of Linux computers remotely across a network.
PortSentry	An IDS that monitors traffic on ports and allows you to detect whether hackers are probing your ports using port scanning utilities such as nmap.
Snort / Airsnort	A complex IDS that can be used to capture and monitor network packets. It can be used to detect a wide range of network attacks and port probing.
Linux Intrusion Detection System (LIDS)	An IDS that involves modifying the Linux kernel to increase process and file security as well as detect security breaches.
Simple WATCHer (SWATCH)	An IDS that monitors log files and alerts administrators when an intrusion is detected.

Table 14-5: Common Linux Intrusion Detection Systems

Tripwire

- Install tripwire from yum
- (1) Update /etc/tripwire/twcfg.txt and /etc/tripwire/twpol.txt as needed
- (2) Run `tripwire-setup-keyfiles`
  - Create passphrases, generates keys, signs policies
- (3) Run `tripwire --init` to initialize tripwire
  - Update twpol.txt and comment (#) the errors generated by this init
    - twpol by default lists files for pretty much all Linux distros, not just Scientific
- Run integrity check to see if filesystem objects match database snapshot
  - Run `tripwire --check`

Email Report to Admin

- Let's make this useful...install a Python library

$ sudo yum install scipy

- And now let's send an email report to…myself

$ sudo tripwire --check | mail –s "Tripwire Report for `uname –n`" erik

More Useful…

- Setup a cron entry!
  - Actually, one exists in /etc/cron.daily
  - Modify to add a mail command

Time for groupwork!