A *routing protocol* is

- ✓ a mechansim for routers to communicate with each other and learn how packets should be routed.
- a mechanism where the source or destination address is rewritten in each packet header.
- a mechanism for configuring network devices with IP addresses and other information
- a list of rules for deciding how each outgoing packet should be sent.

2 0 / 1 point

Give a bash command to display the long listing of items in the working directory, but only showing the size (column 5) and the name (column 9, assuming no spaces in any names). Leave a space between the columns, but otherwise do not worry about formatting.

prompt# × ls -l

Correct Answer: **ls -l | awk '{print $5" "$9}'**

3 0 / 1 point

Suppose we are building a public computer lab where machines have optical drives, and we want to prevent users from being able to boot from an optical disc. This is an example of a × user authorization concern.

Correct Answer: **physical security**

4 0 / 1 point

You are setting up a home router that uses NAT. The router has IP address 10.222.33.4/24 on the private (home) network, is given IP address 10.138.12.52/20 by your ISP, and uses 10.138.0.1 as its gateway. Select all IP addresses below that a client machine on the private (home) network could use as its gateway.

- × ☑ 10.138.0.1

Selected Answer - Incorrect

- ☐ 10.138.0.77
- ☐ 10.138.12.52
- ☐ 10.138.12.77
- ✓ ☑ 10.222.33.4
- ☐ 10.222.33.77
- ☐ 10.222.33.222
- ☐ 10.222.34.77

Which strings (ignoring the quotes) match the regular expression

```
[A-Z][ A-Z]* [ 1-6][0-9][0-9]
```

- × ☑ "COM S 752"

Selected Answer - Incorrect

- × ☐ "V W XY Z 666"

Missed Option - Incorrect

- ☐ " E E 490"
- × ☐ "S 200"

Missed Option - Incorrect

- ☐ " 311"
- × ☐ "MATH 10"

Missed Option - Incorrect

0 / 1 point

Give a bash command to rename all files in the working directory whose name has the form *filename*.jpg, so that its new name is *filename*.jpg.jpeg (for any non-empty *filename*; you may assume they do not contain spaces or other troublesome characters)

prompt# × no

Correct Answer: **for f in *.jpg; do mv $f $f.jpeg; done**

0 / 1 point

Give a bash command to give the long listing for items in the working directory whose name ends in .jpeg

prompt# × ls -l *.jpg

Correct Answer: **ls -l *.jpeg**

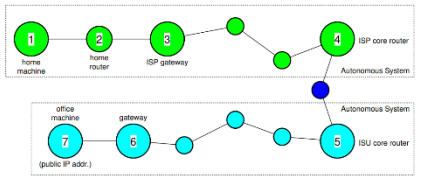
0 / 1 point

Give a bash command, using sed, to display the long listing of items in the working directory, but replacing the first 10 characters on each line with the text "10 mystery".

prompt# × sed

Correct Answer: **ls -l | sed 's/^.........../10 mystery/', ls -l | sed 's/^.{10}/10 mystery/', ls -l | sed 's/.........../10 mystery/', ls -l | sed 's/.{10}/10 mystery/'**

Example: connecting from home to campus



- Routes 1 → 2 and 2 → 3 are trivial
- Route 3 → 4 determined by ISP's Interior Gateway Protocol
- Route 4 → 5 determined by Border Gateway Protocol
- Route 5 → 6 determined by ISU's IGP
- Route 6 → 7 is trivial

On a Linux system that uses firewalld, give a bash command to immediately disallow http packets through the firewall.

prompt# × firewalld off

Correct Answer: **firewall-cmd --remove-service http, firewall-cmd --remove-service=http**

10 0 / 1 point

Systems that require a "special character" in your password, do so because

- × it prevents dictionary style attacks

Correct Answer: **it enlarges the password space, making brute-force password attacks less effective**

- it forces users to type their passwords more carefully
- attackers don't think to try those
- it enlarges the password space, making brute-force password attacks less effective

11 1 / 1 point

When an executable file has its setuid bit set, it causes the running process

- to have its real user set to the file owner of the executable file.
- to run with root privileges.
- to override file permission checks.
- ✓ to have its effective user set to the file owner of the executable file.

12 0 / 1 point

Give a bash command to copy all files in the working directory whose name ends in .jpeg, into a directory named /images

prompt# × no

Correct Answer: **cp *.jpeg /images**

13 0 / 1 point

On a Linux system using systemd, give a bash command that will cause service dhcpd to **not** start up at boot time.

prompt# × no

Correct Answer: **systemctl disable dhcpd**

```
prompt$ echo 'hi hi hi hi' | sed 's/hi/ho/g'
ho ho ho ho
prompt$ ls | sed '4s/txt/text/'
a.out
bar.txt
ch3.sed*
foo.text
prompt$ ls | sed -n 's/bar/pub/p'
```

Cryptic 1–line example program

```
$3 ~ /bob/ {print $9}
```

- For all records where field 3 matches the pattern /bob/, ...
- Print field 9

```
prompt$ ls -l /tmp
total 408
-rw------- 1 alice staff    135 Aug  9  13:30 bar.txt
-rw------- 1 chuck chuck    703 Feb 14   2009 bob
-rwxr-x--- 1 root  bob     1024 Oct  5   2007 congrats*
-rw------- 1 bob   staff   4386 Apr 11   2011 foo.txt
-rw------- 1 chuck staff 391275 Oct 26   2010 turboboost
prompt$ ls -l /tmp | awk '$3 ~ /bob/ {print $9}'
```

14 0 / 1 point

You are setting up a home router that uses NAT. The router has IP address 10.222.33.4/24 on the private (home) network, is given IP address 10.138.12.52/20 by your ISP, and uses 10.138.0.1 as its gateway. Select all IP addresses below that could be assigned to a client machine on the private (home) network.

- ☐ 10.138.0.1
- ☐ 10.138.0.77
- × ☑ 10.138.12.52

Selected Answer - Incorrect

- × ☑ 10.138.12.77

Selected Answer - Incorrect

- × ☑ 10.222.33.4

Selected Answer - Incorrect

- × ☐ 10.222.33.77

Missed Option - Incorrect

- × ☐ 10.222.33.222

Missed Option - Incorrect

- ☐ 10.222.34.77

NFS client

- NFS versions 2 and 3 use the portmapper
  - For file locking
- Make sure the portmapper is running on the client
- Server exports can be mounted using

```
mount -t nfs server:/export /mount/point/as/usual
```

  - The `-t nfs` is optional
  - Mount knows that `server:/export` means NFS
- Server exports may also be automounted
- See `man mount.nfs` and `man nfs` for mounting options
  - Will want the defaults for most options

Example: mount -t nfs 192.168.11.11:/export/ /mount/point

NFS server

- Need the portmapper running for versions 2 and 3
- Need the server daemon running
  - Use "`systemctl`" to start service "`nfs`"
  - This may start several daemons
- Server configuration: in `/etc/exports`
  - Lines starting with "`#`" are ignored
  - One rule per line
  - Rule format: `folder IPs(options) IPs(options) ...`
    - IP can be a complete address
    - IP can be a subnet of the form IP/mask or IP/prefix length
- See `man exports` for more information
  - Especially for the allowed options

Example `/etc/exports`

```
/home     192.168.42.0/24(rw,sync)
/special  192.168.42.3(ro) 192.168.42.4(ro)
```

How to find my user ID?

`id`

- Usage: `id [option]... [username]`
- Print user identity for specified user (default: current user)
- Also prints the current group (or primary group)
- Check your `man` pages for options

```
prompt$ id
uid=1235(alice) gid=152(staff) groups=152(staff),424(hackers)
prompt$ newgrp hackers
prompt$ id
uid=1235(alice) gid=424(hackers) groups=152(staff),424(hackers)
prompt$ id bob
uid=1239(bob) gid=152(staff) groups=152(staff),207(webadmin)
```

The truth about processes

A process has multiple userIDs and groupIDs:

- real user ID
  - User who started the process; its "owner"
  - C system call: `getuid()` to obtain this
- real group ID
  - Current group of user who started the process
  - C system call: `getgid()` to obtain this
- effective user ID
  - User ID to use for file permissions
  - *Usually* the same as the real user ID
  - C system call: `geteuid()` to obtain this
- effective group ID
  - Group ID to use for file permissions
  - *Usually* the same as the group user ID
  - C system call: `getegid()` to obtain this

`chsh` : change your login shell
`gpasswd` : group administration
`groupadd` : add a new group
`groupdel` : remove an existing group
`groupmod` : modify an existing group
`id` : show userID and groupID
`newgrp` : change the current group
`passwd` : change passwords
`sudo` : run a command as another user
`useradd` : add a new user account
`userdel` : remove an existing user account
`usermod` : modify an existing user account

`chmod 640 foo.txt`

6 : 4 + 2 + 0 means `rw-` for user
4 : 4 + 0 + 0 means `r--` for group
0 : 0 + 0 + 0 means `---` for other

`chmod 755 public/`

7 : 4 + 2 + 1 means `rwx` for user
5 : 4 + 0 + 1 means `r-x` for group
5 : 4 + 0 + 1 means `r-x` for other

setuid and setgid bits

- There are two permission bits we have not discussed
  - setuid bit : set user ID upon execution
    - When set, the process's effective user ID is set to the owner of the executable file
  - setgid bit : set group ID upon execution
    - When set, the process's effective group ID is set to the group of the executable file
- These can be changed with `chmod`
  - Use "`s`" where you would use "`r`", "`w`", or "`x`"
  - Use an extra octal digit before the usual three
    - 4 : setuid bit is on
    - 2 : setgid bit is on
    - 1 : sticky bit is on
- These can be seen in `ls -l`
  - Will have an "`s`" instead of "`x`" in the appropriate place

Fun setuid example

```
-r-------- 1 bob bob    27 Nov 2 13:34 file.txt
prompt$ logout

Fedora release 15 (Lovelock)
Kernel 2.6.43.8-1.fc15.i686.PAE on a i686 (tty1)

krankor login: alice
Password:
Last login: Thu Nov 1 17:12:23 on tty1
prompt$ cd /tmp
prompt$ cat file.txt
cat: file.txt: Permission denied
prompt$ ./bobcat file.txt
This is an unreadable file
```

`.` : current directory
- Useful for relative paths
- May appear in absolute paths

`..` : parent directory ("up one")
- Useful for relative paths
- May appear in absolute paths
- In root directory, acts like "`.`"

`~` : current user's home directory
- Only valid at the start of a path
- Expanded by the shell

`~user` : another user's home directory
- Only valid at the start of a path
- Expanded by the shell

`cd` : Change working directory.
`info` : Fancy browsing of the online manual.
`ls` : List the contents of a directory.
`man` : Browse the online manual.
`mkdir` : Create a directory.
`pwd` : Print working directory.
`rmdir` : Remove a directory.
`type` : Is a command built-in, or not?

Summary of today's commands

`cat` : Concatenate a file (to the display).
`chgrp` : Change file group.
`chmod` : Change permissions.
`chown` : Change file owner.
`cp` : Copy files or directories.
`hexdump` : Show hex contents of a file.
`mv` : Move files or directories.
`reset` : Reset a trashed terminal.
`rm` : Remove files or directories.
`df` : show disk space available on devices
`fdisk` : disk partitioning
`ln` : link files
`mkfs` : create a filesystem ("format a disk")
`mount` : mount a filesystem
`touch` : change file time
`umount` : un-mount a filesystem

AWK

- Small scripting language
  - POSIX now specifies a standard for the language
  - Programs are often very short (and cryptic), e.g.:
    `$3 ~ /bob/ {print $9}`
  - You will understand this program by the end of lecture
- Named for its inventors
  - Aho, Weinberger, Kernighan
  - The same Kernighan of "Kernighan and Ritchie" C
- Great for editing streams
- There are multiple implementations of the AWK language
  - This lecture uses a generic "`awk`"
- Often used in pipelines
  - E.g., `crazy | pipeline | awk ... | other | things`

`firewall-cmd` example

```
prompt$ firewall-cmd --add-service=ssh
success
prompt$ firewall-cmd --list-services
http ssh
prompt$ firewall-cmd --remove-service=http
success
prompt$ firewall-cmd --list-services
ssh
prompt$ firewall-cmd --list-services --permanent
mdns dhcpv6-client http
prompt$ firewall-cmd --add-service=ssh --permanent
success
prompt$
```

What to back up

General rule: back up any data that would be hard to reconstruct

- Definitely back up:
  - User files (`/home`)
  - Configuration files (`/etc` and `/var`)
- Definitely do not back up:
  - `/tmp`, `/proc`, `/dev`
- Grey area: other "system files"
  - E.g., kernel image, modules, libraries, applications?
  - Arguments for yes:
    - May be easier to rebuild the system
    - No danger of missing some critical file(s)
  - Arguments for no:
    - These things can be rebuilt "easily" by re-installing the OS
    - Backups will take more time and space if we include these

Single user mode

Must do all of the following to prevent crackers from booting in single user mode:

1. Use a GrUB password
   - Prevents editing of boot entries
2. Disable "boot from CD drive" in the BIOS
   - Prevents booting a live CD
3. Use a BIOS password
4. Lock machine(s) shut
   - Prevents crackers from resetting BIOS
   - Prevents crackers from changing drives

Simple steps to make crackers' work more difficult

- Use a different, memorized, strong password for each system
- Avoid logging in as `root` or administrator
  - Use `su` or `sudo` instead
- Remember the principle of least privilege
- Minimize the amount of software installed
- Minimize the number of running services
- Keep system software up to date
- Use security–enhanced tools whenever possible
  - SELinux, IPTables, TCP wrappers
- Encrypt network traffic whenever possible

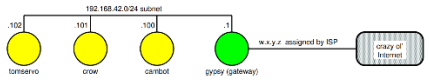
Detecting a rootkit — properly

Suppose you *suspect* that someone has compromised your machine (obtained `root` access). How do you check this?

- Boot using a live CD then search for the rootkit
  - Live CD gives the "proper" versions of utilities
  - The rootkit cannot hide itself
- `chkrootkit`: utility to search for known rootkits
  - `http://freecode.com/projects/chkrootkit`
  - Again, run this from a live CD

cp: copy command

Example Home LAN



Suppose `gypsy` is a typical router
- Packets from one subnet are forwarded to the other

But what happens when I run a browser on `cambot`?

Starting a service: Systemctl start http

Permission meaning

For files

read : necessary to view or copy a file
write : necessary to modify a file
execute : necessary to execute a file

For directories

read : necessary to examine entries ("`ls`" the directory)
write : necessary to modify the directory
- Create a file
- Rename a file
- Remove a file

execute : necessary to access a directory ("`cd`" it)

Principle of Least Privilege

This is such a fundamental philosophy in system security that it gets its own slide

The principle of least privilege

Every entity (user, process, or program) must be able to access only the resources necessary for its legitimate purpose.