Week 06

IT-230

- Manage SELinux Security
- Troubleshooting SELinux
- SELinux File Policies
- SELinux Booleans
- SELinux Practice
- Archives and Compression

IT-230 · Linux Administration

Start `workstation` and `servera` on your VDI

IT-230 · Linux Administration

Manage SELinux Security

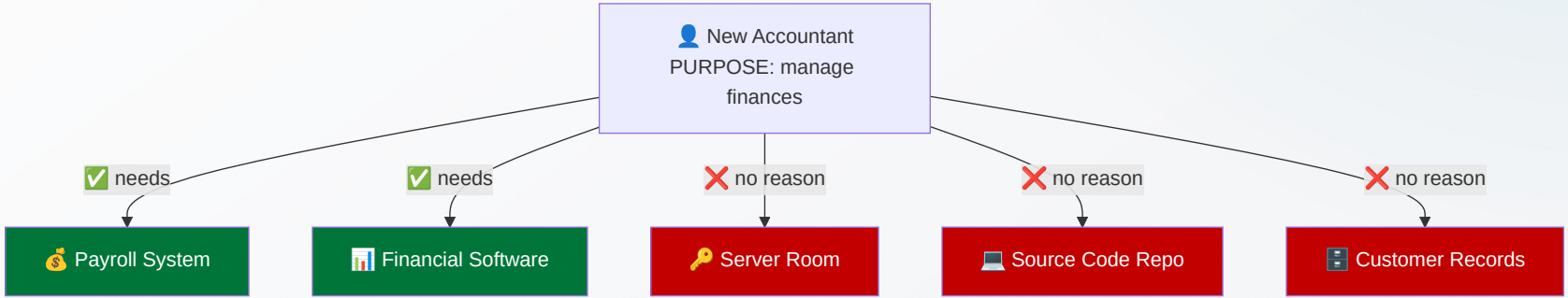
Security-Enhanced Linux

Principle of Least Privilege

Applied to people

- New accountant is hired

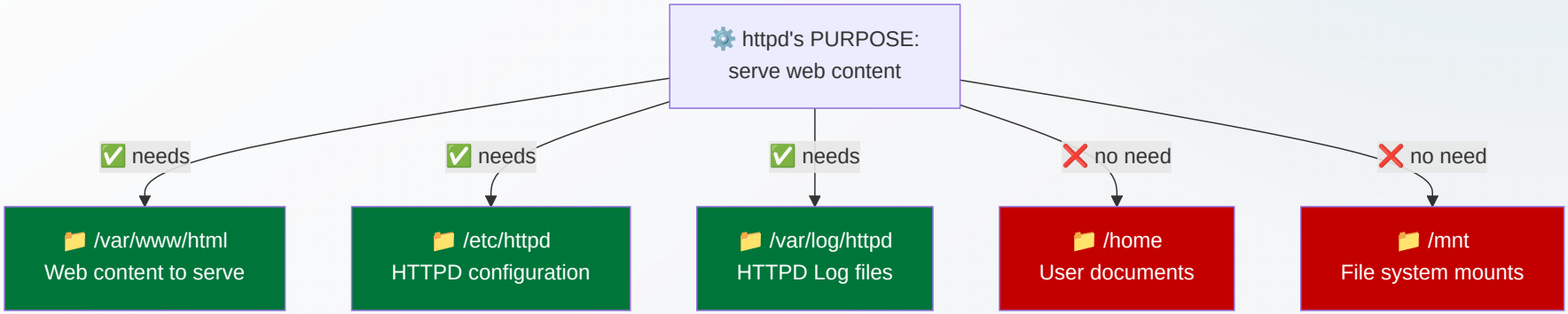
- Ask: What access do they need to do their job?
- Grant **only** that access, nothing more



Principle of Least Privilege

SELinux applies Principle of Least Privilege to processes

- Apache web server is installed
- What access does the `httpd` process need to do its job?
- Grant **only** that access, nothing more



Principle of Least Privilege: DAC vs MAC

Don’t File Permissions Already Do This?

- Yes, but they can be set by the file owner
- Discretionary Access Control (DAC)
- Up to the discretion of the file owner

Only System Administrators Can Change SELinux Policy

- Mandatory Access Control (MAC)
- Mandated by the system administrator

Every Process Has a Label

- Viewed using `ps -AZ`

View Process Label

student@servera:~

```
1 student@servera:~$ ps -AZ | grep httpd
2 system_u:system_r:httpd_t:s0       2205 ?        00:00:00 httpd
3 system_u:system_r:httpd_t:s0       2206 ?        00:00:00 httpd
4 system_u:system_r:httpd_t:s0       2207 ?        00:00:00 httpd
5 system_u:system_r:httpd_t:s0       2209 ?        00:00:00 httpd
6 system_u:system_r:httpd_t:s0       2210 ?        00:00:00 httpd
```

Every File Has a Label

- Viewed using `ls -lZ`

View File Label

student@servera:~

```
student@servera:~$ ls -lZ /var/www/html/
total 4
drwxr-xr-x. 2 root root unconfined_u:object_r:httpd_sys_content_t:s0  6 Sep 17 10:56 about
-rw-r--r--. 1 root root unconfined_u:object_r:httpd_sys_content_t:s0 28 Sep 17 10:56 index.html
```

SELinux Context Labels

```
system_u: object_r: httpd_sys_content_t: s0
```

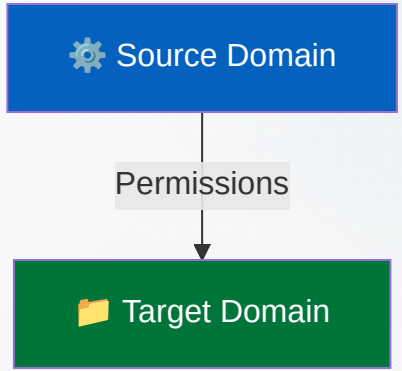
Level

SELinux Policy

List of Rules

Rule

- Source Domain
  - Usually a process
- Target Domain
  - Usually a file or directory
- List of permissions
  - read, write, etc.



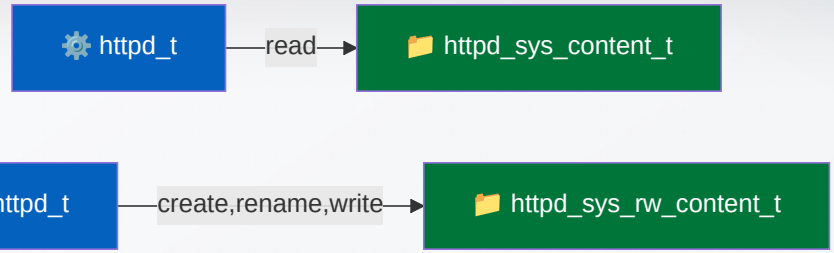
View SELinux Rule

- Viewed using `sesearch -A`

View SELinux Rule

student@servera:~

```
1 student@servera:~$ sesearch -A
2 ...output omitted...
3 allow httpd_t httpd_sys_content_t:dir { ioctl lock read };
4 allow httpd_t httpd_sys_rw_content_t:dir { add_name remove_name write }; [ httpd_builtin_scripting ]:True
5 ...output omitted...
```



Source Domain: httpd process

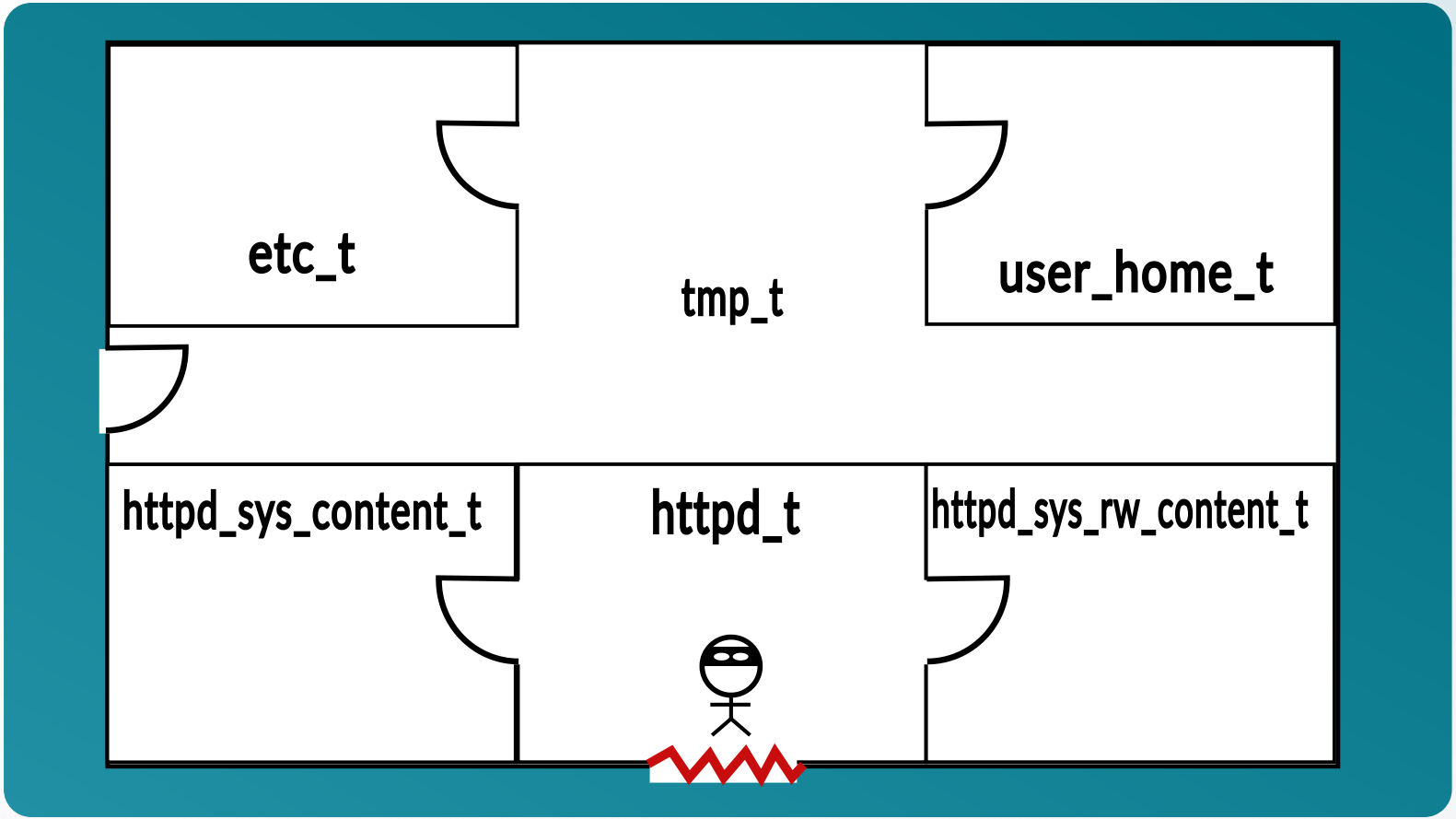
```
student@servera:~$ ps -AZ | grep httpd
system_u:system_r:httpd_t:s0       2205 ?        00:00:00 httpd
...output omitted...
```

Target Domain: html directory

```
student@servera:~$ ls -lZ /var/www
...output omitted...
drwxr-xr-x. 3 root root system_u:object_r:httpd_sys_content_t:s0     37 Sep 17 10:56 html
```

SELinux Rule: sesearch -A output

```
#     Source  Target              Access
allow httpd_t httpd_sys_content_t:dir { ioctl lock read };
```


etc_t
tmp_t
user_home_t
httpd_sys_content_t
httpd_t
httpd_sys_rw_content_t

HANDS-ON EXERCISE

Exercise: Set Up a Basic Web Server

GOAL

Install a basic web server

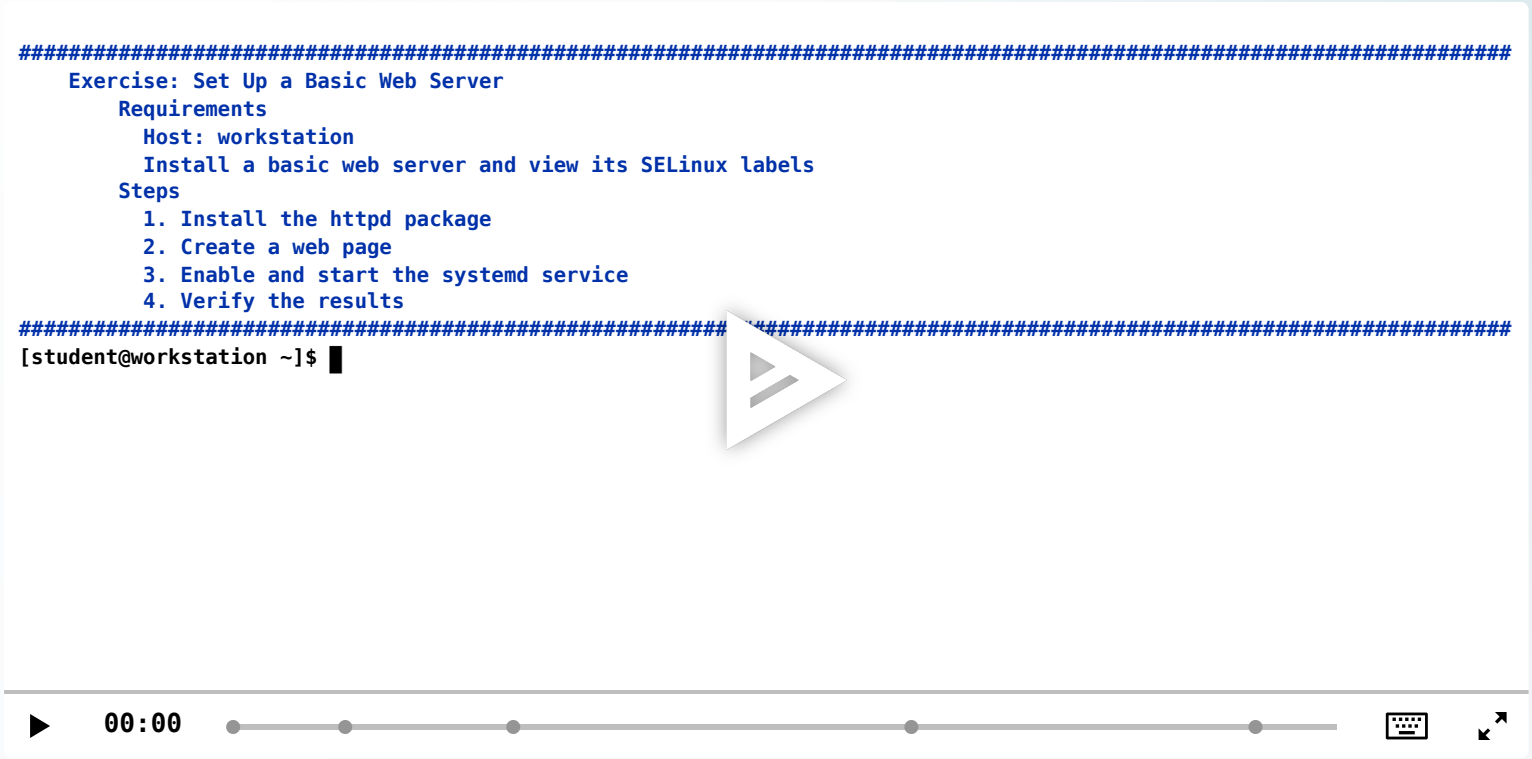
ENVIRONMENT

Host: `workstation`

WORKFLOW

1. Install the `httpd` package
2. Create a web page
3. Enable and start the systemd service
4. Verify the results

Exercise: Set Up a Basic Web Server



Written exercise

IT-230 · Linux Administration

Troubleshooting SELinux

Why is this not working? SELinux?

Troubleshoot SELinux Issues

Manage SELinux mode (temporarily)

Use it **for troubleshooting only**, since these changes do **not** persist through a reboot

- `setenforce 0` sets SELinux to `Permissive` mode
  - Logging only
- `setenforce 1` sets SELinux to `Enforcing` mode
  - Blocking and logging
- `getenforce` shows the current mode (`Enforcing` or `Permissive`)

Setting the Mode at Boot

`/etc/selinux/config` sets the mode at boot

`sestatus` shows detailed information

/etc/selinux/config (abridged)

```
# SELINUX= can take one of these three values:
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
SELINUX=enforcing

# SELINUXTYPE= can take one of these three values:
#     targeted - Targeted processes are protected,
#     minimum - Modification of targeted policy. Only selected...
#     mls - Multi Level Security protection.
SELINUXTYPE=targeted
```

student@servera:~

```
student@servera:~$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /sys/fs/selinux
SELinux root directory:         /etc/selinux
Loaded policy name:             targeted
Current mode:                   enforcing
Mode from config file:          enforcing
Policy MLS status:              enabled
Policy deny_unknown status:     allowed
Memory protection checking:     actual (secure)
Max kernel policy version:      33
```

Troubleshoot SELinux Issues

Log file: `/var/log/audit/audit.log`

Very densely packed information

/var/log/audit/audit.log

```
1 type=AVC msg=audit(1789642643.357:700): avc:  denied  { open } for  pid=2210 comm="httpd" path="/var/www/html/about.html"
2 dev="dm-0" ino=4204014 scontext=system_u:system_r:httpd_t:s0 tcontext=unconfined_u:object_r:user_tmp_t:s0 tclass=file
3 permissive=0
```

Reading an AVC Denial

```
type=AVC msg=audit(1789642643.357:700): avc:  denied  { open } for
pid=2210 comm="httpd" path="/var/www/html/about.html" dev="dm-0"
ino=4204014 scontext=system_u:system_r:httpd_t:s0
tcontext=unconfined_u:object_r:user_tmp_t:s0 tclass=file permissive=0
```

Enforcing mode, so the action really was blocked

Troubleshooting SELinux Issues

There is an easier way: install the `setroubleshoot-server` package and run `sealert`

sealert -a /var/log/audit/audit.log (abridged)

```
SELinux is preventing /usr/sbin/httpd from open access on the file /var/www/html/about.html.

*****  Plugin restorecon (92.2 confidence) suggests   ************************

If you want to fix the label.
/var/www/html/about.html default label should be httpd_sys_content_t.
Then you can run restorecon. The access attempt may have been stopped due to
insufficient permissions to access a parent directory in which case try to
change the following command accordingly.
Do
# /sbin/restorecon -v /var/www/html/about.html
...output omitted...
```

HANDS-ON EXERCISE

Exercise: Add New Web Page

GOAL

The developer wants you to add `about.html` to the web site

ENVIRONMENT

Host: `workstation`

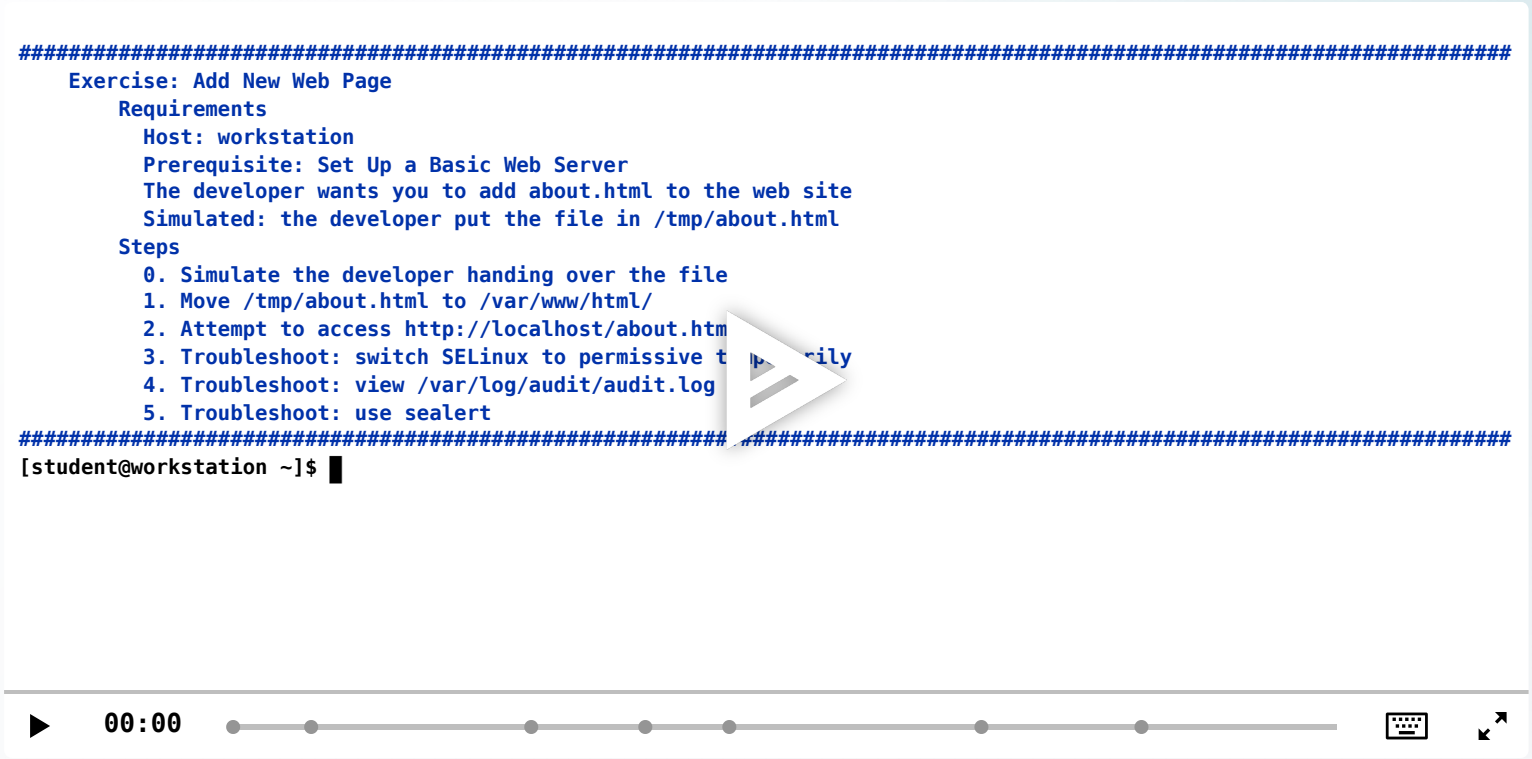
Prerequisite exercise: Set Up a Basic Web Server

Simulated: the developer put the file in `/tmp/about.html`

WORKFLOW

1. Move `/tmp/about.html` to `/var/www/html/`
2. Attempt to access `http://localhost/about.html`
3. Troubleshoot: switch SELinux to permissive temporarily
4. Troubleshoot: view `/var/log/audit/audit.log`
5. Troubleshoot: use `sealert`

Exercise: Add New Web Page



Written exercise

IT-230 · Linux Administration

SELinux File Policies

What decides a file’s label

View SELinux File Policies

What determines the context that is applied to a file?

`semanage fcontext -l` shows all SELinux file policies (there are a **LOT**)

semanage fcontext -l (abridged)

```
...output omitted...
/etc/httpd(/.*)?                    all files          system_u:object_r:httpd_config_t:s0
/tmp                                all files          system_u:object_r:tmp_t:s0
/var/log/httpd(/.*)?                all files          system_u:object_r:httpd_log_t:s0
/var/www(/.*)?                      all files          system_u:object_r:httpd_sys_content_t:s0
/var/www/cgi-bin(/.*)?              all files          system_u:object_r:httpd_sys_script_exec_t:s0
/var/www/git(/.*)?                  all files          system_u:object_r:git_content_t:s0
/var/www/html(/.*)?/uploads(/.*)?   all files          system_u:object_r:httpd_sys_rw_content_t:s0
/var/www/html/[^/]*/cgi-bin(/.*)?   all files          system_u:object_r:httpd_sys_script_exec_t:s0
...output omitted...
```

/var/www/html's Policy

```
/var/www(/.*)?                      all files          system_u:object_r:httpd_sys_content_t:s0
```

Regex Breakdown: `/var/www(/.*)?`

Component	Meaning	Explanation
`/var/www`	**Literal Characters**	Fixed text that matches only `/var/www`
`(...)`	**A Group**	Groups what is inside so a quantifier applies to all of it
`/`	**Literal Character**	The group must start with a forward slash
`.`	**Wildcard Character**	Any single character except a newline
`*`	**Quantifier**	The preceding `.` zero or more times, so `.*` is any sequence of characters
`?`	**Optional Quantifier**	The whole group `(/.*)` zero or one time, so the group is optional

`/var/www` and any files or directories beneath it

`restorecon`: Apply the Policy to Files

`semanage fcontext` only updates the **policy**

`restorecon` reads the database and **stamps the label onto the files**

student@servera:~

```
student@servera:~$ sudo restorecon -v /var/www/html/about.html
Relabeled /var/www/html/about.html from unconfined_u:object_r:user_tmp_t:s0 to
unconfined_u:object_r:httpd_sys_content_t:s0
```

Flag	Meaning
`-v`	**Verbose**: print each file that is relabeled
`-R`	**Recursive**: apply to all files under the given path

`chcon` Skips the Policy

`chcon -t` sets a label directly, but the next `restorecon` puts the policy's label back

student@servera:~

```
student@servera:~$ sudo mkdir /site
student@servera:~$ ls -Zd /site
unconfined_u:object_r:default_t:s0 /site
student@servera:~$ sudo chcon -t httpd_sys_content_t /site
student@servera:~$ ls -Zd /site
unconfined_u:object_r:httpd_sys_content_t:s0 /site
student@servera:~$ sudo restorecon -v /site
Relabeled /site from unconfined_u:object_r:httpd_sys_content_t:s0 to unconfined_u:object_r:default_t:s0
student@servera:~$ ls -Zd /site
unconfined_u:object_r:default_t:s0 /site
```

Caution

Avoid using `chcon` except during a temporary test

HANDS-ON EXERCISE

Exercise: Fix SELinux Label on about.html File

GOAL

Apply the correct label to `about.html`

ENVIRONMENT

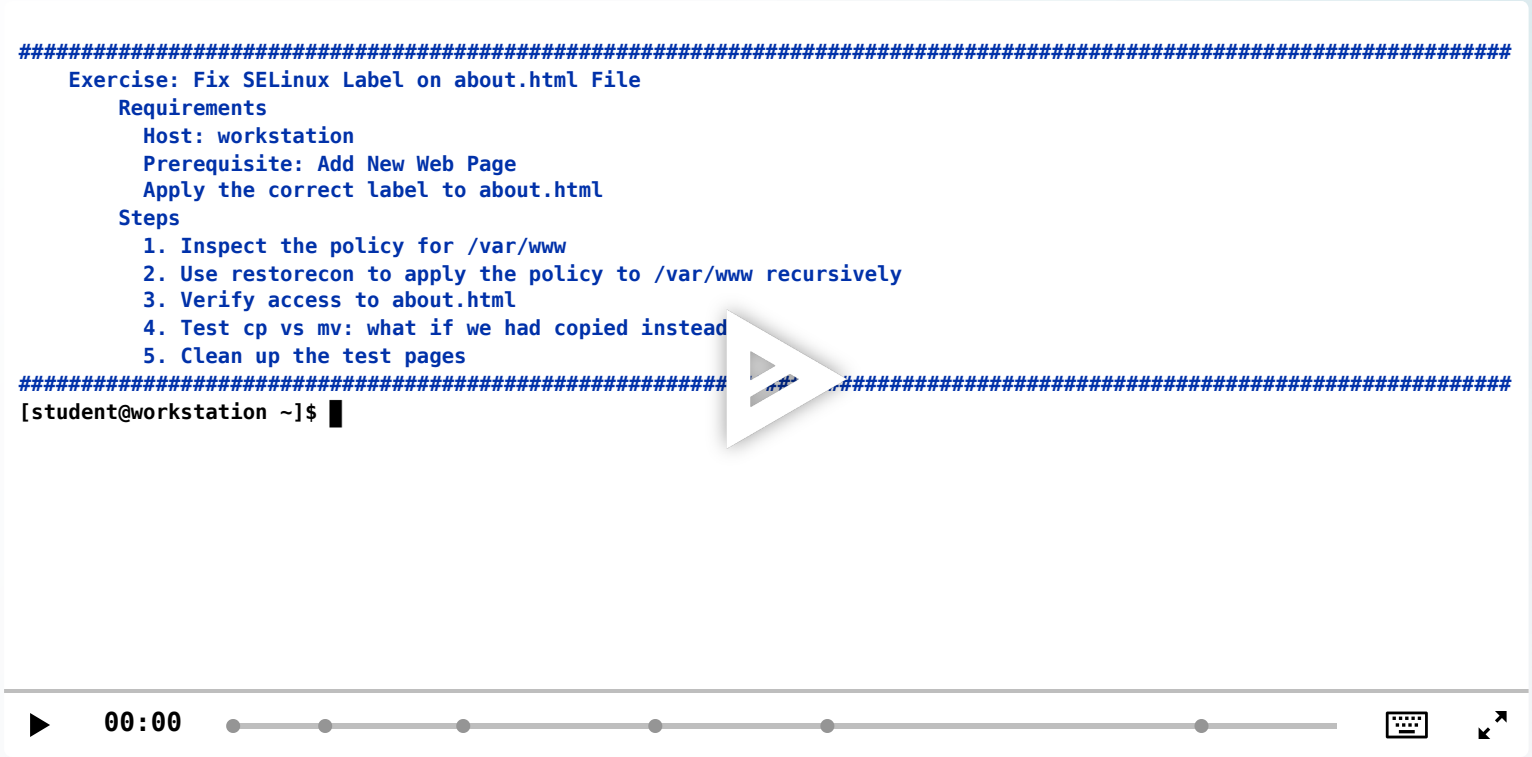
Host: `workstation`

Prerequisite exercise: Add New Web Page

WORKFLOW

1. Inspect the policy for `/var/www`
2. Use `restorecon` to apply the policy to `/var/www` recursively
3. Verify access to `about.html`
4. Test `cp` vs `mv` : what if we had copied instead?

Exercise: Fix SELinux Label on about.html File



Written exercise

IT-230 · Linux Administration

Custom File Policies

Teach the policy about a new location

Files Created in the html Directory Have the Correct Context. How?

- **Policies** are applied when the file is **CREATED**
- Most are defined for you
- You can make custom policies
- `semanage` is the command to manage SELinux policies

Add a Custom Directory

```
semanage fcontext -a -t httpd_sys_content_t '/web(/.*)?'
```

/web and everything beneath it

Other `semanage fcontext` options

`-l`	`-l -C`	`-d`
List rules	List only local customizations	Delete a rule

Policy First, Then `restorecon`

The rule changes nothing on disk until `restorecon` applies it

student@servera:~

```
#^ 1. Add the rule, then list your local rules
student@servera:~$ sudo semanage fcontext -a -t httpd_sys_content_t '/web(/.*)?'
student@servera:~$ sudo semanage fcontext -l -C
SELinux fcontext                                    type               Context

/web(/.*)?                                          all files          system_u:object_r:httpd_sys_content_t:s0
#^ 2. The directory still has its old label
student@servera:~$ ls -Zd /web
unconfined_u:object_r:default_t:s0 /web
#^ 3. restorecon applies the rule to the files
student@servera:~$ sudo restorecon -Rv /web
Relabeled /web from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
Relabeled /web/index.html from unconfined_u:object_r:default_t:s0 to unconfined_u:object_r:httpd_sys_content_t:s0
```

Pointing Apache at the Directory

`/etc/httpd/conf/httpd.conf`

Change `DocumentRoot` and the `<Directory>` block that grants access to it

```
...output omitted...
DocumentRoot "/web"
...output omitted...
<Directory "/web">
    Options Indexes FollowSymLinks
    AllowOverride None
    Require all granted
</Directory>
```

> **Warning**
>
> If you only change the `DocumentRoot` Apache will block the request, before SELinux is ever asked

HANDS-ON EXERCISE

Exercise: Configure a New DocumentRoot for Apache

GOAL

Configure Apache to serve content from the `/web` directory

ENVIRONMENT

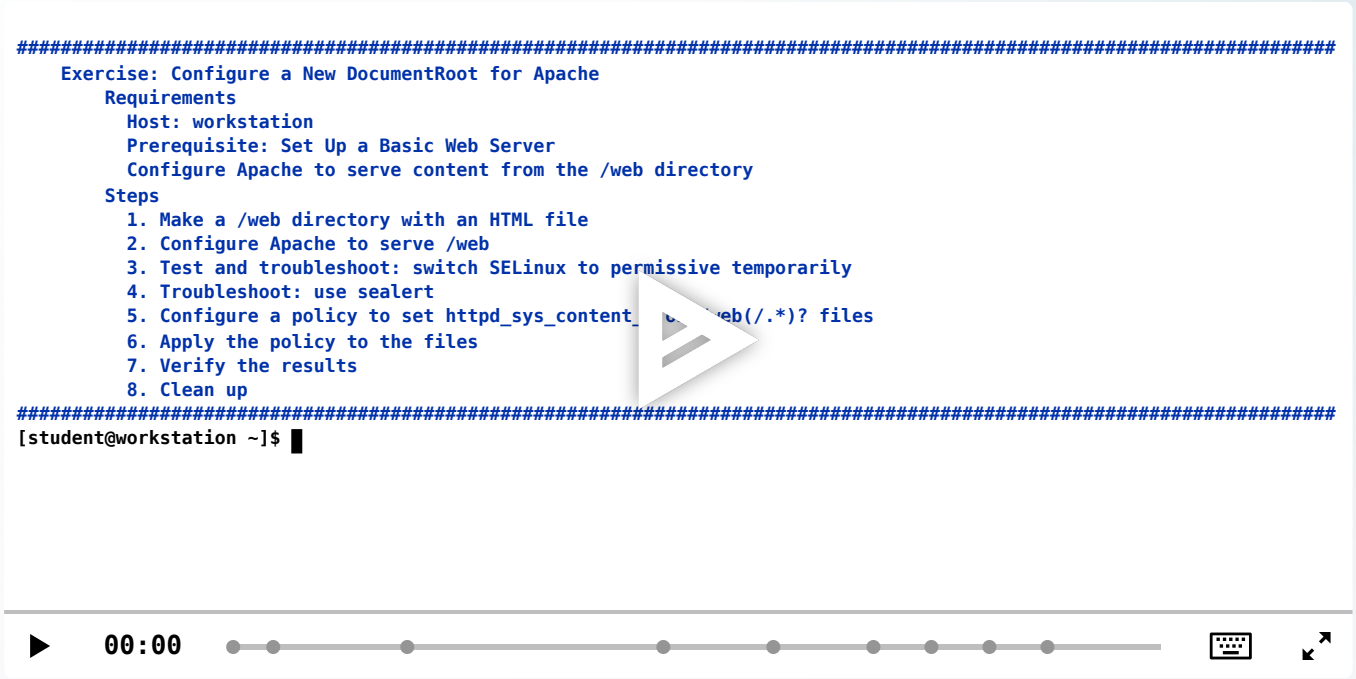
Host: `workstation`

Prerequisite exercise: Set Up a Basic Web Server

WORKFLOW

1. Make a `/web` directory with an HTML file
2. Configure Apache to serve `/web`
3. Test and troubleshoot: switch SELinux to permissive temporarily
4. Troubleshoot: use `sealert`
5. Configure a policy to set `httpd_sys_content_t` on `/web(/.*)?` files
6. Apply the policy to the files
7. Verify the results

Exercise: Configure a New DocumentRoot for Apache



Written exercise

IT-230 · Linux Administration

SELinux Booleans

Switches for common scenarios

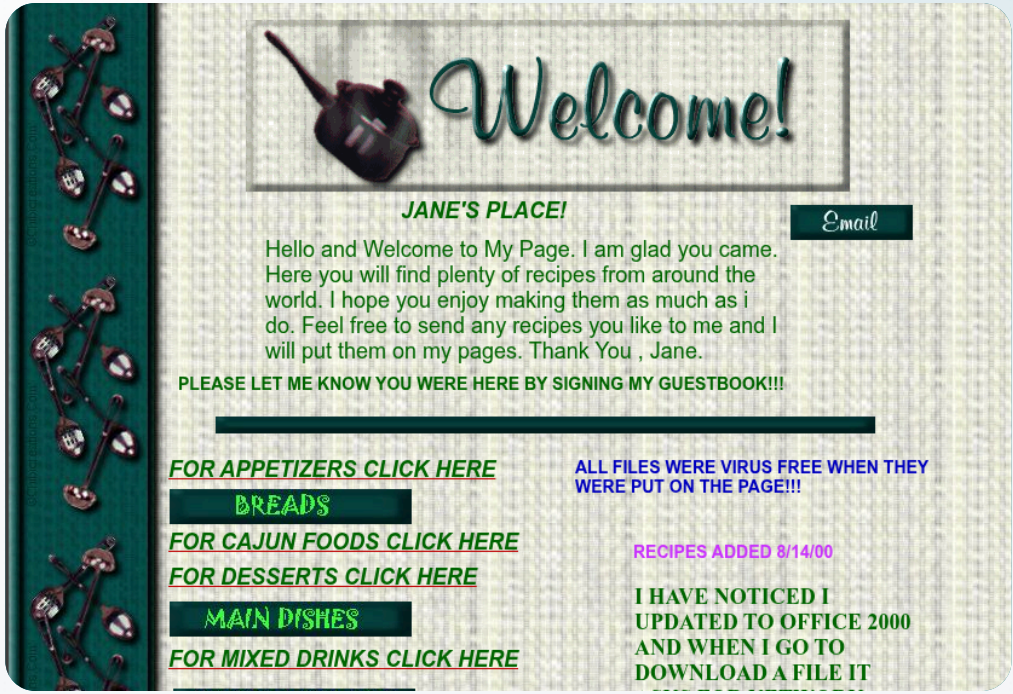
Geocities | Late 1990s-Early 2000s

Allowed anyone to create their own web page

Before social media, a page you built yourself was how you put yourself on the web

Universities ran the same idea on their own servers, where every student and faculty member got space for a personal page

Apache UserDir enabled this explosion of creativity on the web



Apache UserDir

- Lets every user on the server publish their own page
- User writes HTML in `~/public_html/`
- Viewable at `http://server/~username/`

Why Is This Useful?

- Academic environments, where students and faculty publish their own pages, such as `http://cs.university.edu/~jsmith/`
- Multiple developers share one server, each with their own test environment
- Limit access to `/var/www/html`

Configuring Apache UserDir

Edit `/etc/httpd/conf.d/userdir.conf`

```
#UserDir disabled
#
# To enable requests to /~user/ to serve the user's public_html
# directory, remove the "UserDir disabled" line above, and uncomment
# the following line instead:
#
UserDir public_html
```

Restart Apache and create the directory

student@workstation:~

```
student@workstation:~$ sudo systemctl restart httpd.service
student@workstation:~$ mkdir ~/public_html
student@workstation:~$ echo "<h1>Hello from student's Space</h1>" > ~/public_html/index.html
student@workstation:~$ echo "<p>My Categorization of boogers by shape, color, and taste.</p>" >> ~/public_html/index.html
student@workstation:~$
```

Allowing UserDir

In addition to being configured for UserDir, Apache must be given access

1. Discretionary Access Control
   - Permissions to traverse home directories
2. Mandatory Access Control
   1. SELinux policy to allow `httpd_t` to access `httpd_user_content_t`
   2. SELinux context on `~/public_html` and everything below it

1. DAC: Permissions

Home directories normally have `700` permissions

Only the user is allowed to traverse (execute bit on directory)

Need to give the home directory `711` permissions

Allows Apache to navigate to `/home/student/public_html`

student@workstation:~

```
1 student@workstation:~$ chmod 711 /home/student
```

Notice: no sudo required!

The `student` has discretion to control the permissions on their home directory

2.1 MAC: SELinux Policy

- What kind of complex policy modifications will I need to make?
  - **None! Thanks to SELinux Booleans**

Red Hat developers have already written hundreds of policies for common scenarios

You just need to **flip the switch** to turn them on

> **Set the boolean to allow `httpd_t` to read `httpd_user_content_t` directories and files**

SELinux Booleans

There are hundreds of pre-written boolean policies, and many are specifically for Apache

student@servera:~

```
student@servera:~$ sudo semanage boolean -l | grep httpd
...output omitted...
httpd_can_sendmail             (off  ,  off)  Allow httpd to can sendmail
httpd_enable_cgi               (on   ,   on)  Allow httpd to enable cgi
httpd_enable_homedirs          (off  ,  off)  Allow httpd to enable homedirs
httpd_use_nfs                  (off  ,  off)  Allow httpd to use nfs
...output omitted...
```

Boolean	Description
`httpd_enable_homedirs`	Allow httpd to read user home directories
`httpd_can_sendmail`	Allow httpd to send email
`httpd_use_nfs`	Allow httpd to serve files from NFS shares

Adjust SELinux Policy with Booleans

Easier configuration of commonly changed SELinux rules

- `semanage boolean -l` to list all booleans
  - `-C` to list only the changed booleans
- `getsebool {NAME}`
  - Get the value of a specific boolean
- `setsebool {NAME} {ON|OFF}`
  - Temporarily write boolean values
  - `-P` permanently write boolean values to policy

Interpreting SELinux Booleans

```
httpd_enable_homedirs   (off   ,   off)   Allow httpd to enable homedirs
```

The description

> **Note**
>
> `setsebool httpd_enable_homedirs on` to change the current value
>
> `setsebool -P httpd_enable_homedirs on` to change the default **and** current value

2 MAC: SELinux Context

Set the boolean to allow `httpd_t` to read `httpd_user_content_t` directories and files

student@workstation:~

```
1 student@workstation:~$ sudo setsebool -P httpd_enable_homedirs on
2 student@workstation:~$ getsebool httpd_enable_homedirs
3 httpd_enable_homedirs --> on
4 student@workstation:~$ sudo semanage boolean -l -C
5 httpd_enable_homedirs             (on   ,   on)  Allow httpd to enable homedirs
```

HANDS-ON EXERCISE

Exercise: Allow Users to Have Personal Web Pages

GOAL

Configure Apache's UserDir feature

ENVIRONMENT

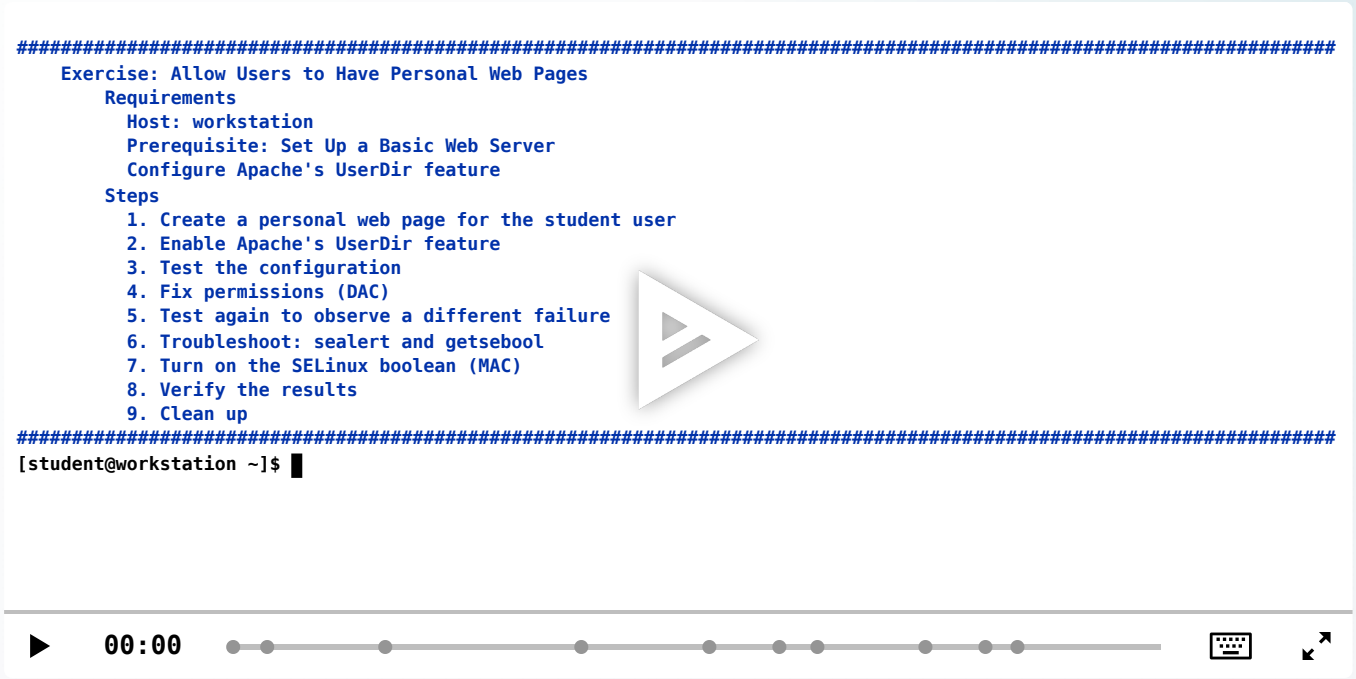
Host: `workstation`

Prerequisite exercise: Set Up a Basic Web Server

WORKFLOW

1. Create a personal web page for the `student` user
2. Enable Apache's UserDir feature
3. Test the configuration
4. Fix permissions (DAC)
5. Test again to observe a different failure
6. Troubleshoot: `sealert` and `getsebool`
7. Turn on the SELinux boolean (MAC)
8. Verify the results

Exercise: Allow Users to Have Personal Web Pages



Written exercise

IT-230 · Linux Administration

SELinux Practice

The same formulas on `servera`

HANDS-ON EXERCISE

Exercise: Configure a New DocumentRoot for Apache on servera

GOAL

Configure Apache to serve content from the `/website` directory

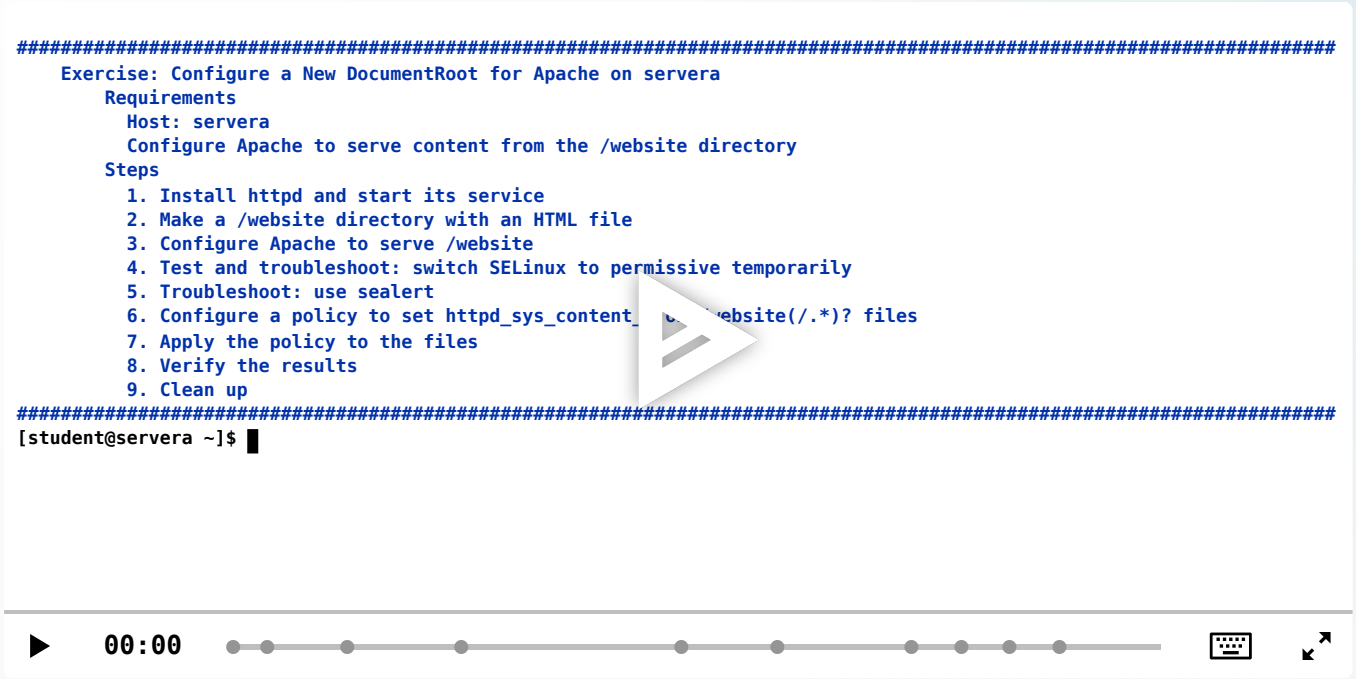
ENVIRONMENT

Host: `servera`

WORKFLOW

1. Install `httpd` and start its service
2. Make a `/website` directory with an HTML file
3. Configure Apache to serve `/website`
4. Test and troubleshoot: switch SELinux to permissive temporarily
5. Troubleshoot: use `sealert`
6. Configure a policy to set `httpd_sys_content_t` on `/website(/.*)?` files
7. Apply the policy to the files
8. Verify the results

Exercise: Configure a New DocumentRoot for Apache on servera



Written exercise

HANDS-ON EXERCISE

Exercise: Allow Apache to Send Email

GOAL

Allow Apache to send email

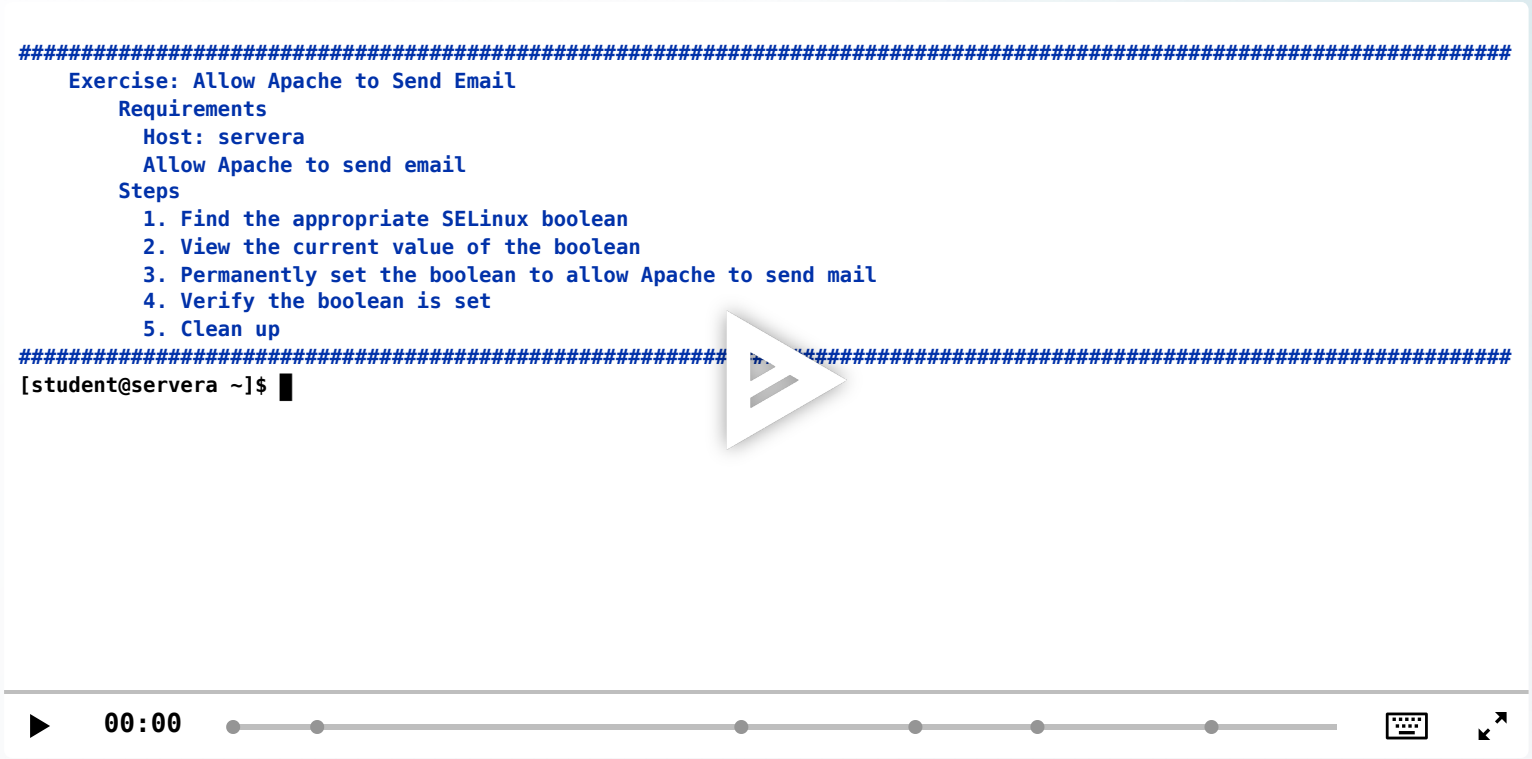
ENVIRONMENT

Host: `servera`

WORKFLOW

1. Find the appropriate SELinux boolean
2. View the current value of the boolean
3. Permanently set the boolean to allow Apache to send mail
4. Verify the boolean is set

Exercise: Allow Apache to Send Email



Written exercise

IT-230 · Linux Administration

Archives

Many files in, one file out

Why Bundle Files at All?

Simpler handling

One file is easier to back up, and manage

Better compression

A single stream of data compresses better than each file individually

> **Note**
>
> Bundling and compressing are two separate jobs.
>
> `tar` does the first but calls a compressor for the second

Preserved characteristics

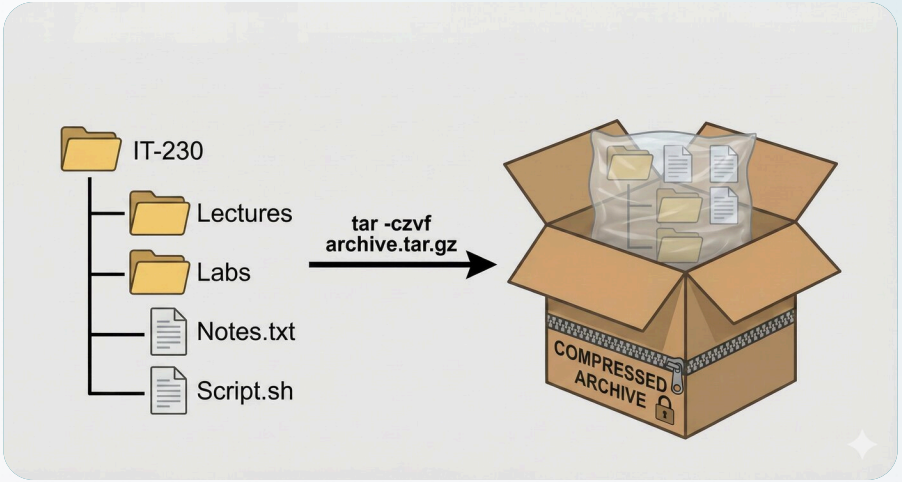
An archive can keep the:

- directory structure
- ownership
- permissions
- timestamps
- etc.

`tar`: Tape ARchive

What is an archive?

- One file that contains many files
- The Unix ancestor of the `.zip` file you know from Windows



How?

- The `tar` command
- **T**ape **AR**chive

Yes, They Really Used to Put Data on Cassette Tapes!

`tar` was written to send data to magnetic **tape**, one file after another in a single continuous stream

Photograph by Robert Jacek Tomczak, CC BY-SA 3.0, via Wikimedia Commons

Creating an Archive

```
tar -c v f /tmp/log-backup.tar /var/log
```

Source Directory

Creating an Archive

root@servera:~

```
root@servera:~# tar -cvf /tmp/log-backup.tar /var/log
tar: Removing leading `/' from member names
/var/log/
/var/log/private/
/var/log/wtmp
/var/log/lastlog
...output omitted...
root@servera:~# ls -lh /tmp/log-backup.tar
-rw-r--r--. 1 root root 7.1M Sep 17 11:01 /tmp/log-backup.tar
root@servera:~#
```

Warning

`tar` strips the leading `/` so the archive cannot overwrite `/var/log` when someone extracts it somewhere else

Listing What Is Inside

Read the contents without unpacking anything

```
tar -tf /tmp/log-backup.tar
```

File: the archive to inspect

root@servera:~

```
root@servera:~# tar -tf /tmp/log-backup.tar | head -n4
var/log/
var/log/private/
var/log/wtmp
var/log/lastlog
root@servera:~#
```

Extracting an Archive

`tar` unpacks into the **current working directory**, so change into an empty one first

root@servera:~

```
root@servera:~# mkdir -p /tmp/log-extract
root@servera:~# cd /tmp/log-extract
root@servera:/tmp/log-extract# tar -xf /tmp/log-backup.tar
root@servera:/tmp/log-extract# ls
var
root@servera:/tmp/log-extract#
```

`-x` is extract

`-f` names the archive

Extracting One File

Same extract command

Just add the file you want to extract at the end

Name the file exactly as `-t` printed it

root@servera:~

```
root@servera:~# mkdir -p /tmp/log-one
root@servera:~# cd /tmp/log-one
root@servera:/tmp/log-one# tar -xf /tmp/log-backup.tar var/log/secure
root@servera:/tmp/log-one# ls var/log
secure
root@servera:/tmp/log-one#
```

> **Note**
>
> Use `var/log/secure`, not `/var/log/secure`, because the archive stores the path without its leading slash

The Three Operations

Option	Name	What it does
`-c`	create	build a new archive from files
`-t`	list	show what is inside an archive
`-x`	extract	unpack an archive into the current directory

`-f` always names the archive file,

`-v` always makes the operation verbose

Caution

`-c` overwrites its target without asking, exactly like the `>` operator

Archives and File Attributes

An archive leaves out extended attributes unless you ask for them

Option	Keeps
`--selinux`	SELinux contexts
`--acls`	POSIX access control lists
`--xattrs`	Other extended attributes
`-p`	Original permissions on extract, already on for `root`

Without `--selinux`, extracted files are labeled like any new file

HANDS-ON EXERCISE

Exercise: Creating and Extracting Archives

GOAL

Back up `/etc` and restore it somewhere safe

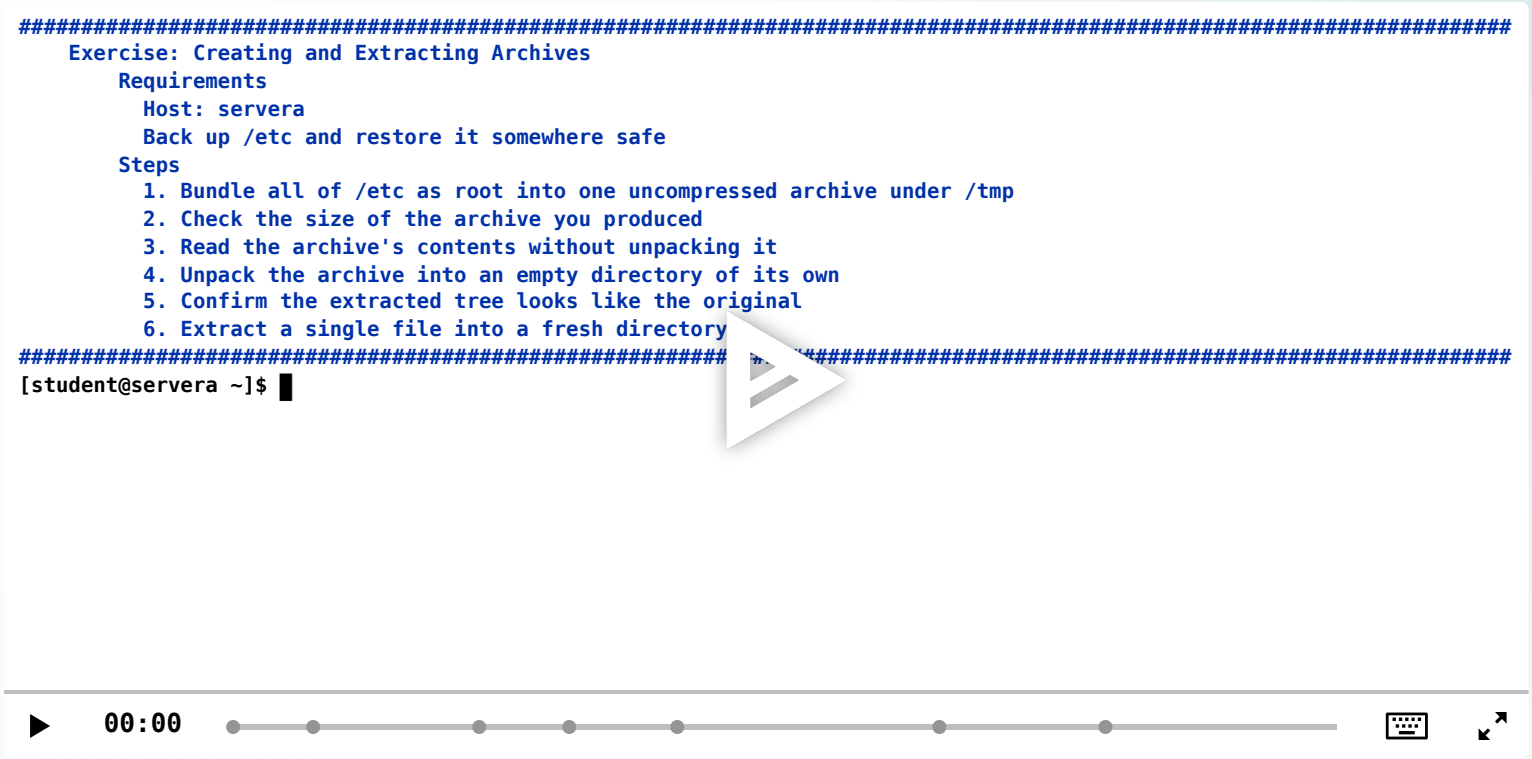
ENVIRONMENT

Host: `servera`

WORKFLOW

1. Bundle all of `/etc` as `root` into one uncompressed archive under `/tmp`
2. Check the size of the archive you produced
3. Read the archive's contents without unpacking it
4. Unpack the archive into an empty directory of its own
5. Confirm the extracted tree looks like the original
6. Extract a single file into a fresh directory

Exercise: Creating and Extracting Archives



Written exercise

IT-230 · Linux Administration

Compression

Trading CPU time for disk space

Compression gives you samller files

At the cost of CPU time

The Trade

The same `/var/log` archive, kept two ways

Uncompressed: `log-backup.tar`

7.1 MB

Compressed: `log-backup.tar.gz`

677 KB

Disk space saved: **6.4 MB**

Additional CPU Use is the cost:**once to compress, and on every read**

Is Compression Worth It?

It depends

How often will the data be read?

What kind of data is it?

Which algorithm are you using?

How much storage do you have?

How busy is the CPU?

Three Compressors, One Option Each

Add one letter to the `tar` options you already know

Option	Algorithm	Extension	Character
`-z`	gzip	`.gz`	Old faithful, installed nearly everywhere
`-j`	bzip2	`.bz2`	Smaller than gzip, more CPU; often missing on minimal installs
`-J`	xz	`.xz`	Smallest of the three, and the most CPU

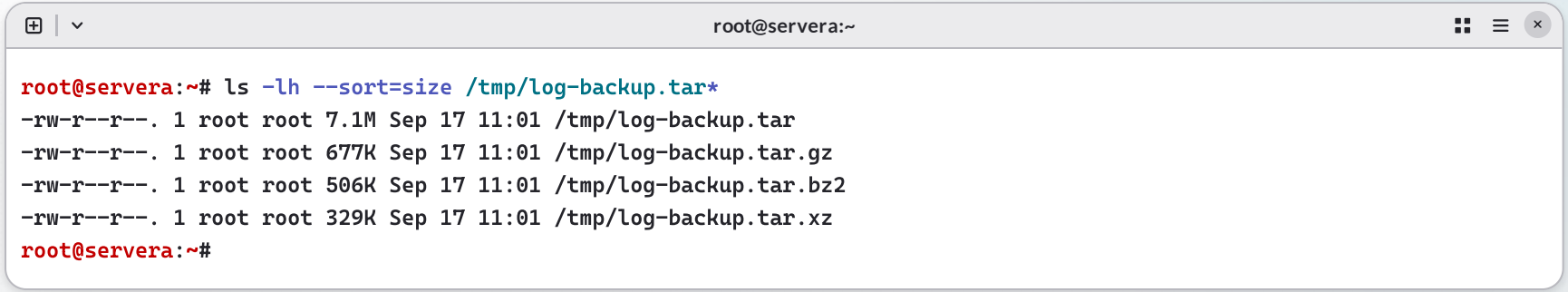
> **Warning**
>
> `-j` and `-J` are different options: lowercase is bzip2, uppercase is xz

Creating a Compressed Archive

```
tar -c z v f /tmp/log-backup.tar .gz /var/log
```

files to compress

Comparing the Results



Extracting Is Simpler Than Creating

`tar` inspects the file and picks the right decompressor itself

root@servera:/tmp/log-xz

```
root@servera:/tmp/log-xz# tar -xf /tmp/log-backup.tar.xz
root@servera:/tmp/log-xz#
```

> **Note**
>
> No `-J` needed

Checking a Compressed Archive

`gzip` and `xz` report the ratio with `-l`

root@servera:~

```
root@servera:~# gzip -l /tmp/log-backup.tar.gz
         compressed        uncompressed  ratio uncompressed_name
             692689             7352320  90.6% /tmp/log-backup.tar
root@servera:~# xz -l /tmp/log-backup.tar.xz
Strms  Blocks   Compressed Uncompressed  Ratio  Check   Filename
    1       1    328.6 KiB  7,180.0 KiB  0.046  CRC64   /tmp/log-backup.tar.xz
```

See how much disk space will be used by decompressing

HANDS-ON EXERCISE

Exercise: Comparing Compression Algorithms

GOAL

Find which compressor makes the smallest `/etc` backup, and at what cost

ENVIRONMENT

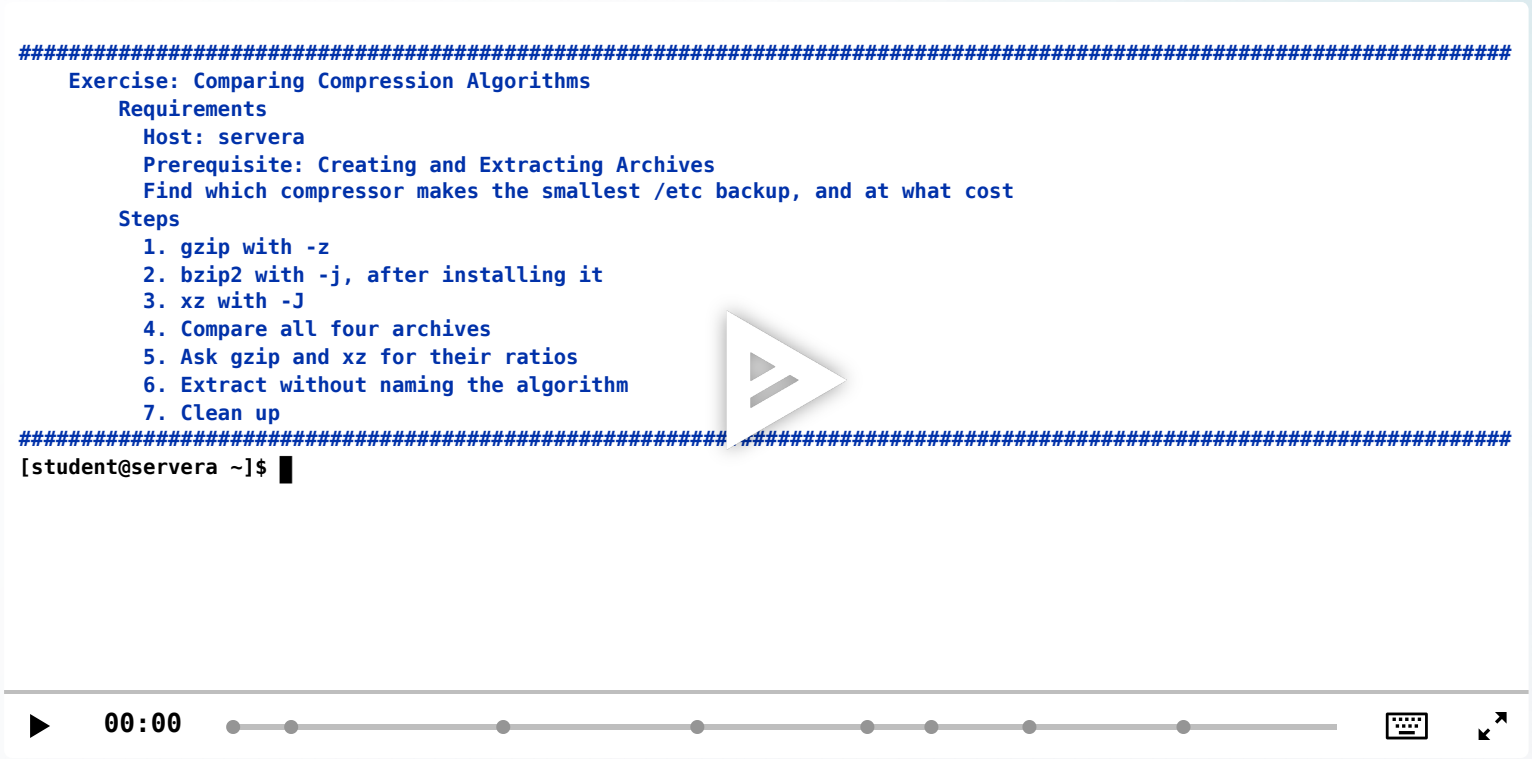
Host: `servera`

Prerequisite exercise: Creating and Extracting Archives

WORKFLOW

1. Bundle `/etc` three more times, once with each of the three compressors
2. Install whichever compression package is missing before you reach for it
3. List all four archives together, sorted so the sizes are easy to compare
4. Ask `gzip` and `xz` to report the compression ratio of their own archives
5. Extract one of the compressed archives without naming its algorithm
6. Note which archive was smallest and which took longest to create

Exercise: Comparing Compression Algorithms



Written exercise