

## Linux/Unix Access control commands

This handout describes some of the main Linux/Unix commands for access control.

### whoami

Gives the current users ID, e.g.:

```
sh-3.2# whoami
root
```

### ls -l

This command lists the files and the “-l” option displays the access control information, e.g.

```
wallace% ls -l
total 78580
drwxr-xr-x  7 tpc staff   8192 Oct  6 10:32 Desktop
-rwxr--r--  1 tpc staff    546 May  7 13:31 desktop.ini
drwxr-x---  2 tpc staff  4096 Aug 12 18:18 Teaching
drwxr-xr-x  4 tpc staff  4096 Aug  3 10:34 Papers
drwx-----  2 tpc staff  4096 Oct 13 13:46 private
drwxr-xr-x 16 tpc staff  4096 Jun  8 10:25 Projects
```

### chmod

Changes the access control settings for a file, e.g.

```
wallace% chmod o-rw private.txt
```

this particular command would remove all public read and write access to the file private.txt. Type `man chmod` to get full information on the syntax.

### chown

As root only you can change the ownership of a file. As the owner you can change its group:

```
wallace% chown tpc:webswwinternet Lecture3ICW.mov
wallace% ls -l
total 76728
-rw-rw-r-- 1 tpc webswwinternet 78409234 Oct 13 13:19 Lecture3ICW.mov
wallace%
```

## **su**

Switches user, if no user is named it switches you to root (as long as you know the password).

```
laptop:~ laptop$ whoami
laptop
laptop:~ laptop$ su
Password:
sh-3.2# whoami
root
```

## **sudo**

Executes a single command as root, e.g.

```
wallace%sudo rm /etc/shadow
```

The root password is required (running the above command would be a very bad idea).

## **Some interesting files on a Linux/Unix system:**

<code>/etc/passwd</code>	stores information on a system's users
<code>/etc/group</code>	stores the groups
<code>/etc/shadow</code>	stores the password hashes.
<code>/usr/bin/passwd</code>	a program to change passwords, i.e., edit the shadow file this program can be executed by anyone, but runs as root.