Working with Files and Directories

REDIRECTION(OUTPUT REDIRECTION)

- Your keyboard is your standard input (stdin) device, and the screen or a particular terminal window is the standard output (stdout) device.
- **The redirection operators**
- Output redirection with > and |

REDIRECTION(OUTPUT REDIRECTION)

- Sometimes you will want to put output of a command in a file, or you may want to issue another command on the output of one command. This is known as redirecting output.
- Redirection is done using either the ">" (greater-than symbol), or using the "|" (pipe) operator which sends the standard output of one command to another command as standard input.

REDIRECTION(OUTPUT REDIRECTION)

- cat test1

  some words

- cat test2

  some other words

- **cat test1 test2 > test3**
- cat test3

some words

some other words

REDIRECTION(INPUT REDIRECTION)

- In another case, you may want a file to be the input for a command that normally wouldn't accept a file as an option. This redirecting of input is done using the "<" (less-than symbol) operator.

REDIRECTION(THE >> OPERATOR)

- Instead of overwriting file data, you can also append text to an existing file using two subsequent greater-than signs:
- Example:

```
cat wishlist
more money
less work
```

REDIRECTION(THE >> OPERATOR)

- date >> wishlist
- cat wishlist

```
more money
less work
Thu Feb 28 20:23:07 CET 2002
```

The date command would normally put the last line on the screen; now it is appended to the file wishlist.

FILES AND DIRECTORY PERMISSIONS (CHMOD)

- Change the file permissions.
- Specifications = u user, g group, o other, + add permission, - remove, r read, w write , x execute.

- **chmod <specification> filename**
- **chmod (u,g,o)(+,-,=)(r,w,x)**

FILES AND DIRECTORY PERMISSIONS (CHMOD)

- Add read permission for the owner and the group.
  chmod go+r myfile
- Allow all users to read, write or execute myfile.
  chmod a +rwx myfile

FILES AND DIRECTORY PERMISSIONS (CHMOD)

- Remove read permission from the group and others.

chmod go -r myfile

- Assigns multiple permissions.
- **chmod u=rw,og=x myfile**

FILES AND DIRECTORY PERMISSIONS (CHMOD)

- Change the permissions of a directory recursively.
- To change permission of a directory and everything within that directory, use this command.
- **chmod -R <specification> dirname**

FILES AND DIRECTORY PERMISSIONS (CHMOD)

Examples:

First, we grant the group execute permission and then we take write permission away from the owner:

- ls -l file1.txt
- chmod g+x file1.txt
- ls -l file1.txt

FILES AND DIRECTORY PERMISSIONS (CHMOD)

- chmod u-w file1.txt

- ls -l file.txt

FILE OWNERSHIP COMMANDS (CHOWN, CHGRP)

- Change ownership of a file to user owner1.

chown owner1 filename

- Change primary group ownership of file filename to group grp_owner.

chgrp grp_owner filename

FILE OWNERSHIP COMMANDS (CHOWN, CHGRP)

- Change primary group ownership of directory dir-name to group grp_owner recursively.
- To change group ownership of a directory and everything within that directory, use this command.

```
chgrp -R grp_owner dir-name
```