chmod and chown File permissions

CONCEPTS

> Mode digits are sums per class: r=4, w=2, x=1. 640 means owner rw-, group r--, others ---.

> On directories, r lists names, w creates or deletes entries, x lets you enter and access items by name.

OCTAL MODES

chmod 644 *<file>*
rw for owner, r for rest

chmod 755 *<file>*
rwx for owner, rx for rest

chmod 600 *<file>*
Private to owner

chmod 700 *<dir>*
Private directory

SYMBOLIC MODES

chmod +x *<file>*
Add execute (per umask)

chmod u+x *<file>*
Add execute for owner

chmod g+w *<file>*
Let group write

chmod o-rwx *<file>*
Remove all access for others

chmod u=rw,go=r *<file>*
Set classes exactly

RECURSIVE CHANGES

chmod -R *<mode>* *<dir>*
Recurse into a directory

chmod -R a+rX *<dir>*
Read all, x only on dirs

chmod -R u=rwX,go=rX *<dir>*
Sane tree permissions

find *<dir>* **-type d -exec chmod 755 {} +**
Dirs only

RECURSIVE CHANGES (cont.)

find *<dir>* **-type f -exec chmod 644 {} +**
Files only

SPECIAL BITS

chmod u+s *<file>*
Setuid: run as file owner

chmod g+s *<dir>*
Setgid dir: new files keep group

chmod +t *<dir>*
Sticky: only owners may delete

READING AND COPYING MODES

ls -l *<path>*
Long listing with mode

ls -ld *<dir>*
Mode of the dir itself

stat -c '%a %U:%G %n' *<file>*
Octal mode and owner

chmod --reference=*<ref>* *<file>*
Copy mode from another file

CHOWN

chown *<user>* *<file>*
Change owner

chown *<user>*:*<group>* *<file>*
Change owner and group

chown :*<group>* *<file>*
Change group only

chown -R *<user>*:*<group>* *<dir>*
Recursive ownership

sudo chown -R $USER: *<dir>*
Reclaim files made by root

UMASK

umask
Show current mask

umask 022
New files 644, dirs 755