



## Διάλεξη 3

# Διαχείριση Συστημάτων UNIX I

Δημήτρης Ζεϊναλιπούρ



# Περιεχόμενο Διάλεξης

- **Δομή καταλόγων** (*ls, cd, pwd, pathnames, pushd, popd*), χώρος δίσκου (*du, df*), Σύνδεσμοι (συμβολικοί, σκληροί - *ln*).
- **Συμπίεση-αποσυμπίεση** (*zip/unzip, gzip/gunzip, bzip2/bunzip2, tar*)
  - Παράδειγμα *Backup* (με *public/private keys*)
- **Ιδιοκτησία και Δικαιώματα Πρόσβασης** (*chmod, chgrp, chown, umask, suid, sgid, sticky bit*),
- **Προσθήκη Χρηστών** (*useradd/ldap*)



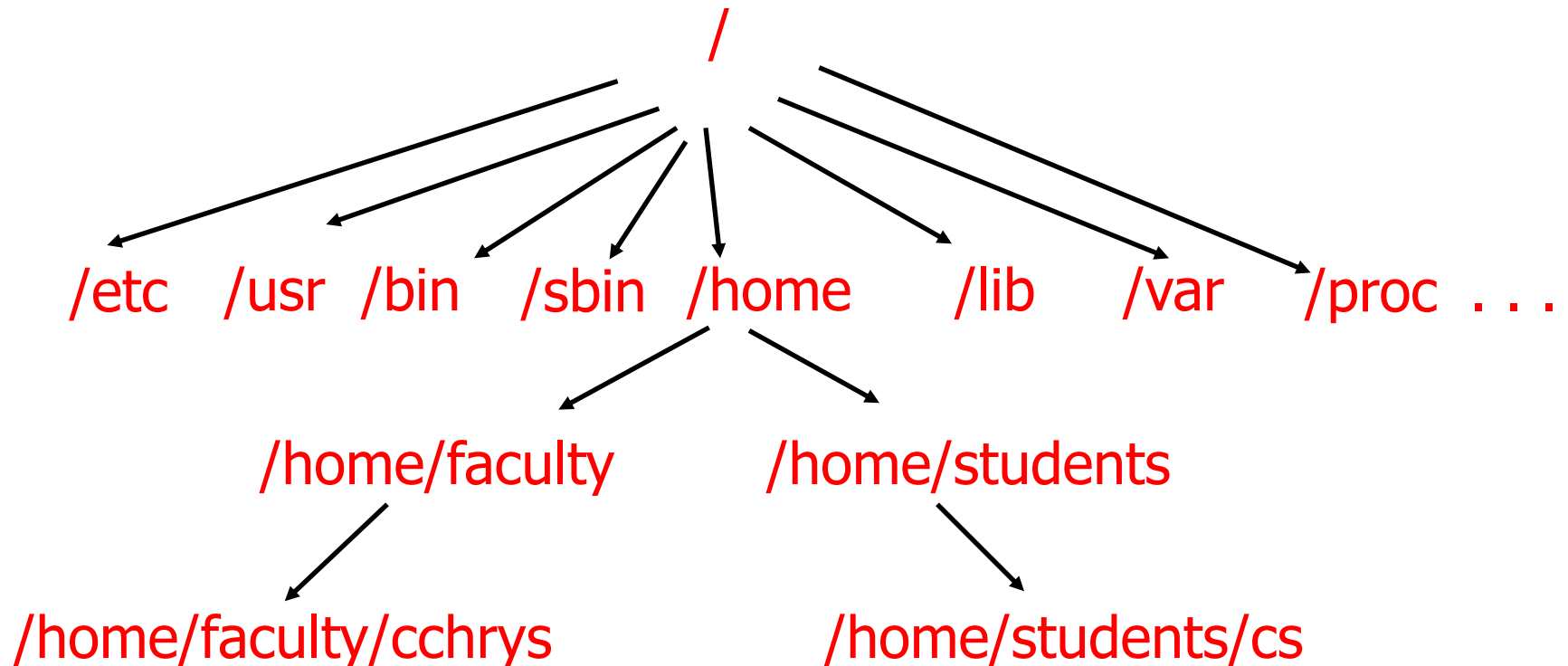
# Δομή Καταλόγων UNIX

- Το **UNIX** μεταχειρίζεται τα πάντα ως **αρχεία**
  - Standard text files and binaries
  - Κατάλογοι (Directories)
  - Σύνδεσμοι (Links)
  - Συσκευές
    - τερματικό (terminal), πληκτρολόγιο, σκληρός δίσκος, ..
- Αυτό πάει πίσω στο σχεδιασμό του UNIX **προσπαθώντας** να κρατήσει τα **πάντα απλά**.
  - Μεταχειρίζοντας τα πάντα το ίδιο επιτρέπει σε μια απλή διεπαφή να αλληλεπιδρά με όλα με τον ίδιο τρόπο.



# Δομή Καταλόγων UNIX

- Δομή Αντεστραμμένου Δέντρου





# Δομή Καταλόγων UNIX

- **/**
  - Ριζικός κατάλογος (Root) ολόκληρου του συστήματος
- **/boot**
  - Αρχεία που χρησιμοποιούνται κατά την εκκίνηση.
  - Το “/boot/vmlinuz...” μονολιθικός πυρήνας του Linux ή /mach\_kernel πυρήνας του MACOSX;
- **/bin**
  - Binaries (τα σημαντικότερα, όπως shells, ls, grep, ...)
- **/usr/bin**
  - Binaries (λιγότερο σημαντικά, εργαλεία που χρησιμοποιούνται από τον χρήστη όπως mysql, gnome, python, zip, κτλ ...)



# Δομή Καταλόγων UNIX

- **/sbin και /usr/sbin**

- System Binaries (Εκτελέσιμα που χρησιμοποιούνται για συντήρηση
  - Χαμηλού Επιπέδου (/sbin): reboot, grub (bootloader), mount, tcpdump, iptables, nmap, κτλ.
  - Υψηλού Επίπεδου (/usr/bin): π.χ., **httpd, squid, sshd, vsftpd, dovecot** κτλ.

- **/usr**

- Περιέχει δεδομένα που σχετίζονται με τους χρήστες, π.χ.,
  - user binaries, τα έγγραφά τους, βιβλιοθήκες, επικεφαλίδες αρχείων, κλπ...



# Δομή Καταλόγων UNIX

- **/lib (MacOSX => usr/lib & /Library)**
  - Διαμοιραζόμενες Βιβλιοθήκες (Shared libraries) για προγράμματα που γίνονται linked δυναμικά (όπως τα DLL's στα Windows)
  - Π.χ., /lib/libc-2.5.so είναι η βιβλιοθήκη της C στο UNIX η οποία δένεται δυναμικά με το εκτελέσιμο. Στα windows από την άλλη προγράμματα C είναι statically linked με τις βιβλιοθήκες (παρεχεται από τον compiler)
- **/dev**
  - Συσκευές (π.χ. disk, cdrom, dvd, port, audio, κλπ). Θυμηθείτε ότι στο Unix τα πάντα είναι αρχεία!
- **/home (MacOSX => /Users)**
  - Αποθηκεύονται οι λογαριασμοί χρηστών
  - Π.χ., /home/faculty/dzeina/



# Δομή Καταλόγων UNIX

- **/var**
  - Χώρος όπου διατηρούνται δεδομένα τα οποία αλλάζουν συχνά (π.χ., logs, emails, εργασίες που γίνονται queued, π.χ., printer jobs)
- **/proc (not available on MacOSX ☹)**
  - Χώρος όπου «φυλάγονται» (με νοητό τρόπο όχι πραγματικά) πληροφορίες για τις υπό εκτέλεση διεργασίες (ανοικτά αρχεία, μνήμη, κτλ)
    - Π.χ., `cat /proc/$$/status` δείχνει πληροφορίες για την διεργασία του κελύφους που εκτελείτε.
- **/etc:** Configuration files (inittab: processes started at system bootup, fstab: file systems and their mount points)
- **/lost+found:** files that it restores after a system crash



# Δομή Καταλόγων στο Android (Linux Kernel 2.6)



SH0APPL00803:

app-cache

config

cache

**sdcard**

acct

**mnt**

d

**etc**

system

sys

shutdown.bravo.rc

**sbin**

**proc**

init.rc

init.goldfish.rc

init.bravo.rc

**init**

default.prop

**data**

bootcomplete.bravo.rc

**root**

**dev**



# Πλοήγηση στη Δομή Καταλόγων

- Εντολή **ls** (επιλογές **-a**, **-l**, **-r**, **-R**, **-t**, **-i**)
  - Λίστα αρχείων και καταλόγων στο υφιστάμενο κατάλογο
- a** : *(all) do not ignore files starting with . (hidden files)*
- l** : *use a long listing format*  
`-rw-r--r-- 1 dzeina faculty 950784 Sep 14 13:29 01.Introduction.ppt`
- r** : *reverse order while sorting (αντίστροφη ταξινόμηση)*
- R** : *list subdirectories recursively (αναδρομικά)*
- t** : *sort by modification time (ταξινόμηση βάσει του χρόνου τροποποίησης)*
- i** : *print the index (inode) number of each file*



# Πλοήγηση στη Δομή Καταλόγων

- Ιδιότητες Αρχείων
  - Δικαιώματα
  - Αριθμό Σκληρών Συνδέσμων
  - Ιδιοκτήτης
  - Ομάδα Ιδιοκτήτη
  - Μέγεθος
  - Χρόνος Τροποποίησης
  - Όνομα

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ ls -lR test
test:
total 8
drwxr-xr-x 2 cchrys tspecial 4096 Jan 23 23:56 test1
drwxr-xr-x 2 cchrys tspecial 4096 Jan 23 23:56 test2

test/test1:
total 0
-rw-r--r-- 1 cchrys tspecial 0 Jan 23 23:56 test1.txt

test/test2:
total 0
-rw-r--r-- 1 cchrys tspecial 0 Jan 23 23:56 test2.txt
bash-3.1$
```

*drwxr-x--- 2 dzeina faculty*

*40 Oct 24 2006 zei*



# Πλοήγηση στη Δομή Καταλόγων

- Εντολή *cd*
  - Αλλαγή τρέχοντος καταλόγου
- Ειδικοί Κατάλογοι-Συμβολισμοί
  - Κατάλογος-ρίζα (/)
  - Τρέχων κατάλογος (.)
  - Γονικός κατάλογος (..)
  - Κατάλογος Αφετηρίας (\$HOME) (~)
  - Αυτοί οι κατάλογοι μπορούν να «στοιβαχθούν»
    - Π.χ. *../..* → δυο κατάλογοι πάνω από τον τρέχων κατάλογο



# Πλοήγηση στη Δομή Καταλόγων

- Εντολή ***pwd (ή echo \$PWD)***
  - Εμφάνιση **Απόλυτου** ονόματος-μονοπατιού τρέχοντος καταλόγου
- Ονόματα-Μονοπάτια
  - **Απόλυτα**
    - Πάντοτε ξεκινά από τον κατάλογο-ρίζα (/) και περιλαμβάνει όλη τη διαδρομή
      - Π.χ. */home/faculty/dzeina/public*
  - **Σχετικά**
    - Σχετικά με τον τρέχων κατάλογο
      - Π.χ. *~/public*

# Προχωρημένη Πλοήγηση στη Δομή Καταλόγων



- Εντολή *pushd* <dir>
  - «Σπρώχνει» ένα κατάλογο σε μια στοίβα
    - Επίσης μετακινούμαστε στο κατάλογο <dir>
- Εντολή *popd*
  - «Βγάζει» την κεφαλή X από τη στοίβα
    - Επίσης μετακινούμαστε στο κατάλογο <X>
- Μια στοίβα είναι γνωστή ως *LIFO*
  - *Last In, First Out*
- Εκτελώντας: *pushd*
  - παραθέτει το περιεχόμενο της στοίβας.

# Χώρος Δίσκου και Όριο Χρήσης



- Έλεγχος χρήσης χώρου δίσκου
  - Εντολή **du** (disk usage)
    - Δείχνει πόσο χώρο (σε kilobytes) δεσμεύεται για κάθε αρχείο ή κατάλογο (αναδρομικά)
      - **du -c**: συνοψίζει το τελικό άθροισμα σε bytes.
      - **du -s**: εκτελεί την ανάδρομη αλλά παρουσιάζει μόνο το άθροισμα του μεγέθους του καταλόγου (όχι το μέγεθος κάθε επί μέρους καταλόγου)

## Παραδείγματα

**du -c answers.txt input.txt**

28    answers.txt

4     input.txt

**32    total**

**du ze1**

0     ze1/s1

20    ze1

**du -s ze1**

20    ze1

# Χώρος Δίσκου και Όριο Χρήσης



- Έλεγχος διαθέσιμου χώρου δίσκου
  - Εντολή **df -h** (*human readable output*)
    - Δείχνει πόσος χώρος (σε kilobytes) είναι διαθέσιμος στο file system

Ο λογαριασμός του χρήστη έχει περιορισμένο χώρο.

Χρησιμοποιείται ένα όριο χρήσης (*quota*) για να ελεγχθεί ο διαθέσιμος χώρος.

`quota -v` # display disk usage and limits

Filesystem	Size	Used	Avail	Use%	Mounted on
/dev/mapper/VGSystem-LVroot	2.0G	881M	1002M	47%	/
/dev/mapper/VGSystem-LVtmp	992M	34M	908M	4%	/tmp
/dev/mapper/VGSystem-LVvar	3.2G	1.3G	1.8G	41%	/var
/dev/mapper/VGSystem-LVopt	496M	113M	359M	24%	/opt
/dev/mapper/VGSystem-LVusr	4.4G	2.8G	1.4G	67%	/usr
/dev/mapper/VGSystem-LVusrLocal	496M	64M	407M	14%	/usr/local
/dev/sda1	99M	40M	55M	43%	/boot
tmpfs	2.0G	0	2.0G	0%	/dev/shm
/dev/mapper/VGData-LVdata	56G	33G	21G	61%	/sys-data
csfs5.cs.ucy.ac.cy:/home/projects	25G	21G	3.4G	87%	/home/projects
csfs3.cs.ucy.ac.cy:/home/support	100G	68G	33G	68%	/home/support
csfs1.cs.ucy.ac.cy:/home/faculty	432G	410G	23G	95%	/home/faculty
csfs7.cs.ucy.ac.cy:/home/students	170G	130G	41G	77%	/home/students
csfs4.cs.ucy.ac.cy:/home/research	342G	338G	4.3G	99%	/home/research

NFS Mounts





# Φύλαξη Χώρου Δίσκου

- Συμπίεση αρχείων
  - Εντολή **zip**
    - Κάθε αρχείο που συμπιέζεται αντικαθίσταται με ένα αρχείο με προέκταση **.zip**
    - Παράδειγμα: **zip file.zip \***
  - Εντολή **gzip**
    - **GNU εκδοχή του zip**
    - Κάθε αρχείο που συμπιέζεται αντικαθίσταται με ένα αρχείο με προέκταση **.gz**
  - Εντολή **bzip2**
    - Διαφορετικός αλγόριθμος (LZW)
    - Κάθε αρχείο που συμπιέζεται αντικαθίσταται με ένα αρχείο με προέκταση **.bz2**

## Επιλογή -v

- Verbose mode – δείχνει την αναλογία συμπίεσης για κάθε αρχείο που επεξεργάζεται
- *Τυπώνει:* total bytes=55, compressed=44 -> 20% savings



# Φύλαξη Χώρου Δίσκου

- Αποσυμπίεση αρχείων
  - Εντολή *unzip*
  - Εντολή *gzip -d* (decompress) ή *gunzip*
  - Εντολή *bzip2 -d* ή *bunzip2*

# Φύλαξη Χώρου Δίσκου

Verbosely (δηλ.,  
δείχνει τι  
συμπιέζεται στο  
tar file)



create file contents list extract

- Εντολή **tar** (επιλογές **-c**, **-f**, **-t**, **-v**, **-x**)
  - Σώζει πολλά αρχεία μαζί σ' ένα **ενιαίο αρχείο** και **διατηρεί πληροφορίες του filesystem** όπως user, group permissions, dates, directory structures.
  - TAR: Tape Archive
- Παραδείγματα
  - **tar -cvf** archive.tar foo bar
    - Δημιουργεί το *tar\_file* από τα αρχεία foo και bar.
  - **tar -xvf** archive.tar
    - Εξάγει όλα τα αρχεία από το *tar\_file*.
  - **tar -cvfz** archive.tar.gz foo bar
    - Δημιουργεί συμπιεσμένο *tar\_file* (με τον αλγόριθμο *gzip*) από τα αρχεία foo και bar

# Παράδειγμα Εργαλείου Backup



```
# Execute every day at 02:30
# crontab -e
# Add: 30 2 * * * /home/faculty/dzeina/backups/anyplace/wwwbackup.sh

#!/bin/bash
date

# carry out remote backup
echo -e "Backup WWW Play Web  ap.cs.ucy.ac.cy .."
ssh -l anyplace ap.cs.ucy.ac.cy tar -zcvf /tmp/anyplace-www-daily-backup.tar.gz /home/anyplace
echo "Done"

# transfer on csucy remote backup
echo -e "Transferring remote /tmp/anyplace-www-daily-backup.tar.gz locally ..."
scp anyplace@ap.cs.ucy.ac.cy:/tmp/anyplace-www-daily-backup.tar.gz
/home/faculty/dzeina/backups/anyplace/anyplace-www-daily-backup.tar.gz
echo "Done"

# delete remote backup
ssh -l anyplace ap.cs.ucy.ac.cy rm -rf /tmp/anyplace-www-daily-backup.tar.gz
```



# Συμβολικοί Σύνδεσμοι

- Ειδικός τύπος αρχείου: Όχι αρχείο δεδομένων, αλλά αρχείο που περιέχει ένα *δείκτη* σε κάποιο άλλο αρχείο (δηλ., σαν Shortcut στα Windows).
  - Δρα ως συντόμευση
    - Συντόμευση σε ένα κατάλογο
    - Παρέχει ένα γρήγορο σύνδεσμο σε οποιοδήποτε αρχείο.

```
cs4038.in.cs.ucy.ac.cy - PuTTY
bash-3.1$ ls -l test/LinkToFile2.txt
lrwxrwxrwx 1 cspgcc1 cspg 20 Jan 24 10:32 test/LinkToFile2.txt -> test/test2/test2.txt
bash-3.1$
```



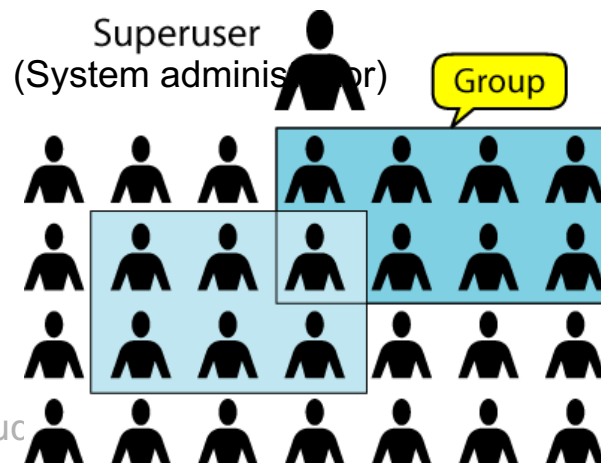
# Σύνδεσμοι

- Εντολή ***ln*** (επιλογή ***-s***) ***<TARGET> <NAME>***
  - ***-s*** δηλώνει ένα **συμβολικό σύνδεσμο (symbolic link)**
  - χωρίς την επιλογή αυτή, ένας **σκληρός σύνδεσμος (hard link)** δημιουργείται με το περιεχόμενο του *target* αρχείου.
  - Π.χ., `ln -s oldfile symboliclinkfile`
- Διαφορά μεταξύ ενός σκληρού και ενός συμβολικού συνδέσμου:
  - Θα εξηγηθεί αργότερα πιο αναλυτικά.
  - Προκαταρκτική επεξήγηση:
    - **Συμβολικός σύνδεσμος:** Ένα αρχείο (ή κατάλογος) X, μαζί με πολλές εγγραφές μετα-πληροφοριών (inodes) που αναφέρονται στο X.
      - Κάθε inode μετα-πληροφοριών είναι σαν Shortcut
    - **Σκληρός σύνδεσμος:** Ένα αρχείο (ή κατάλογος) X, μαζί με 1 εγγραφή μετα-πληροφοριών (inode) που αναφέρεται στο X. Η εγγραφή μετα-πληροφοριών (inode) καταγράφει πόσοι αναφέρονται στο εν λόγω αρχείο.

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- **Χρήστης (User)**: οποιοσδήποτε έχει λογαριασμό στο UNIX σύστημα
- Οι χρήστες οργανώνονται σε **Ομάδες (Groups)**.
  - ένας ή περισσότεροι χρήστες μπορούν να ανήκουν σε **πολλαπλές ομάδες**.



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Για να βρεις πληροφορίες σε ποια/ες ομάδα/ες ανήκει ένας χρήστης:
  - Εντολή **groups** <username>

```
bash-3.1$ groups dzeina
```

```
dzeina : faculty ep1371
```

```
bash-3.1$ groups cchrys
```

```
cchrys : tspecial ep1001 csphd visiting ep1371
```

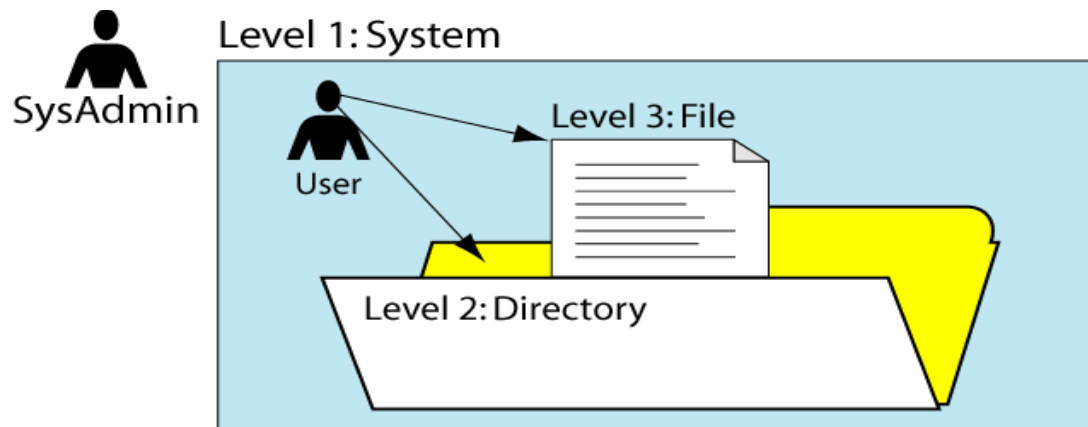
- Σημείωση: Στο UNIX, κάθε χρήστης **ΠΡΕΠΕΙ** να είναι μέλος **τουλάχιστο μιας ομάδας** (αυτή που ορίζεται από το GID μέσα στο **/etc/passwd**)



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



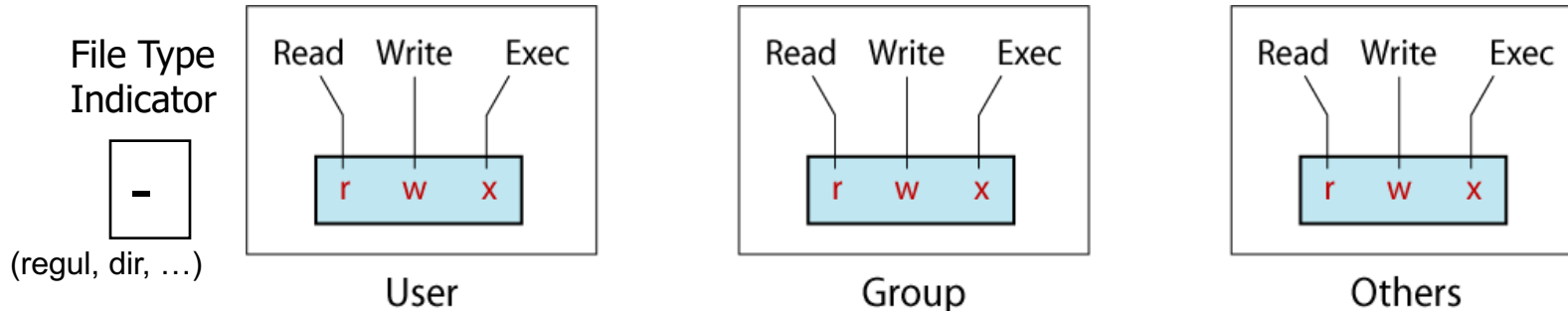
- Επίπεδα Ασφάλειας:
  - Σύστημα, Κατάλογος, Αρχείο
  - Ασφάλεια Συστήματος: ελέγχεται από τον διαχειριστή του συστήματος (system administrator)
  - Κατάλογος και Αρχείο: ελέγχεται από το χρήστη στον οποίο ανήκει



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



## • Κώδικας Δικαιωμάτων Πρόσβασης\*



- **User (Χρήστης-Ιδιοκτήτης):** Ο δημιουργός του αρχείου
- **Group (Ομάδα):** Σειτ από χρήστες που ομαδοποιούνται
- **Others (Υπόλοιποι):** Οποιοσδήποτε λογαριασμός που δεν ανήκει στην Ομάδα αλλά ανήκει σε άλλη ομάδα
- Τρεις τύποι δικαιωμάτων πρόσβασης:
  - **r** read
  - **w** write
  - **x** execute
  - **-** permission denied

\* `ls -al <file | directory>`

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Τι υποδηλώνει το κάθε είδος πρόσβασης;

Τύπος Πρόσβασης	Σημασία για <u>Αρχείο</u>	Σημασία για <u>Κατάλογο</u>
<b>r</b> (read)	View file contents	List directory contents
<b>w</b> (write)	Change file contents	- Change directory contents (create and remove files in that dir.)
<b>x</b> (execute)	Run executable file	- Access files explicitly (by name) in the given folder
<b>-</b>	Permission denied	Permission denied

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Δείκτης Τύπου Αρχείου

Χαρακτήρας	Δείκτης Τύπου Αρχείου
<b>d</b>	<b>D</b> irectory
<b>b</b>	<b>B</b> lock-type special file (π.χ., DVD, CDROM, DISK)
<b>c</b>	<b>C</b> harakter-type special file (π.χ., terminals, printers και networks)
<b>l</b>	<b>S</b> ymbolic <b>L</b> ink
<b>p</b>	<b>P</b> ipe
<b>s</b>	<b>S</b> ocket
<b>-</b>	<b>R</b> egular file

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Ελέγχοντας τα Δικαιώματα:
  - Εντολή **ls -l** <file\_OR\_dir\_name>

```
bash-3.1$ ls -lR test/ | head
test/:
total 4
-rw-r--r-- 2 cspgcc1 cspg 28 Jan 25 14:35 HardLinkToFile1.txt
lrwxrwxrwx 1 cspgcc1 cspg 20 Jan 24 10:50 SymbLinkToFile2.txt -> test/test2/test2.txt
drwxr-xr-x 4 cspgcc1 cspg 50 Jan 28 17:25 test1
drwxr-xr-x 2 cspgcc1 cspg 22 Jan 24 10:42 test2
```

*total size of all files in the list (measured in 512 B)*

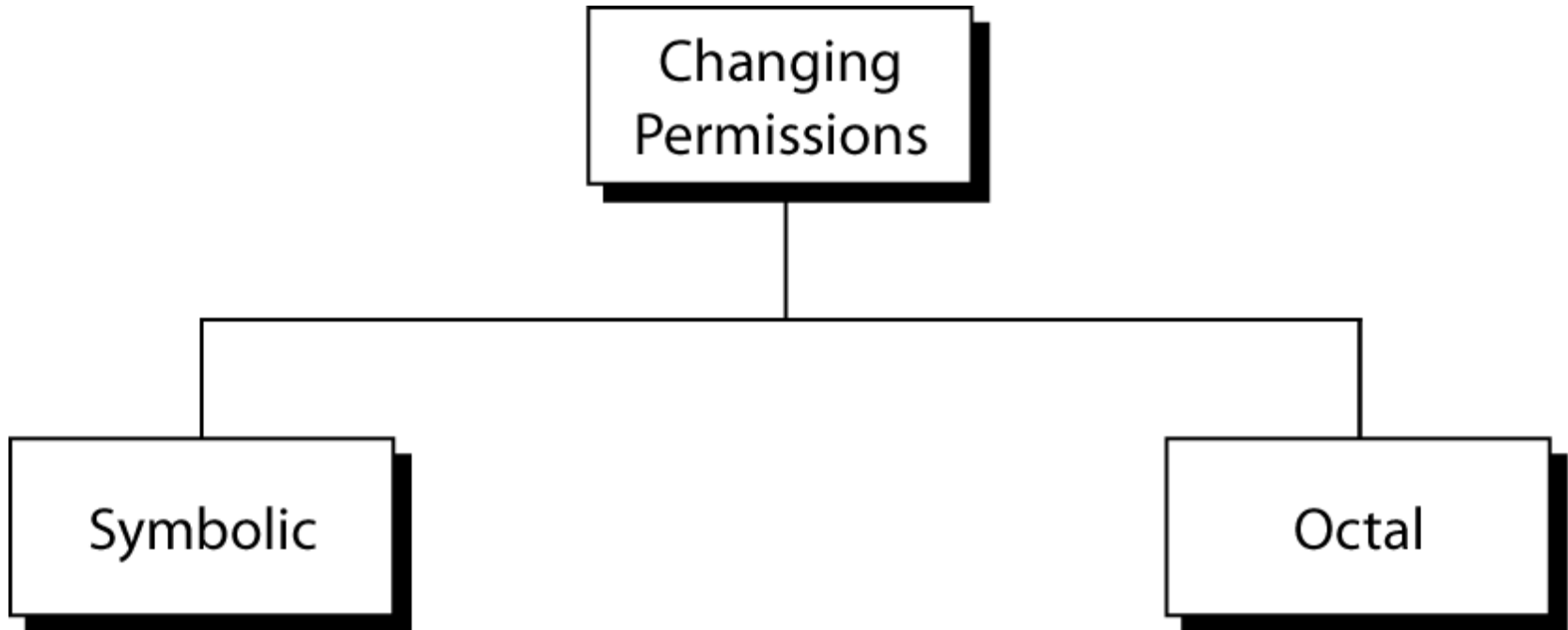
*Permissions, Hard-Links, User, Group, FileSize, Last Modified Date + Time, Filename*

```
test/test1:
total 4
drwxr-xr-x 2 cspgcc1 cspg 24 Jan 23 22:41 test1_1
```

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



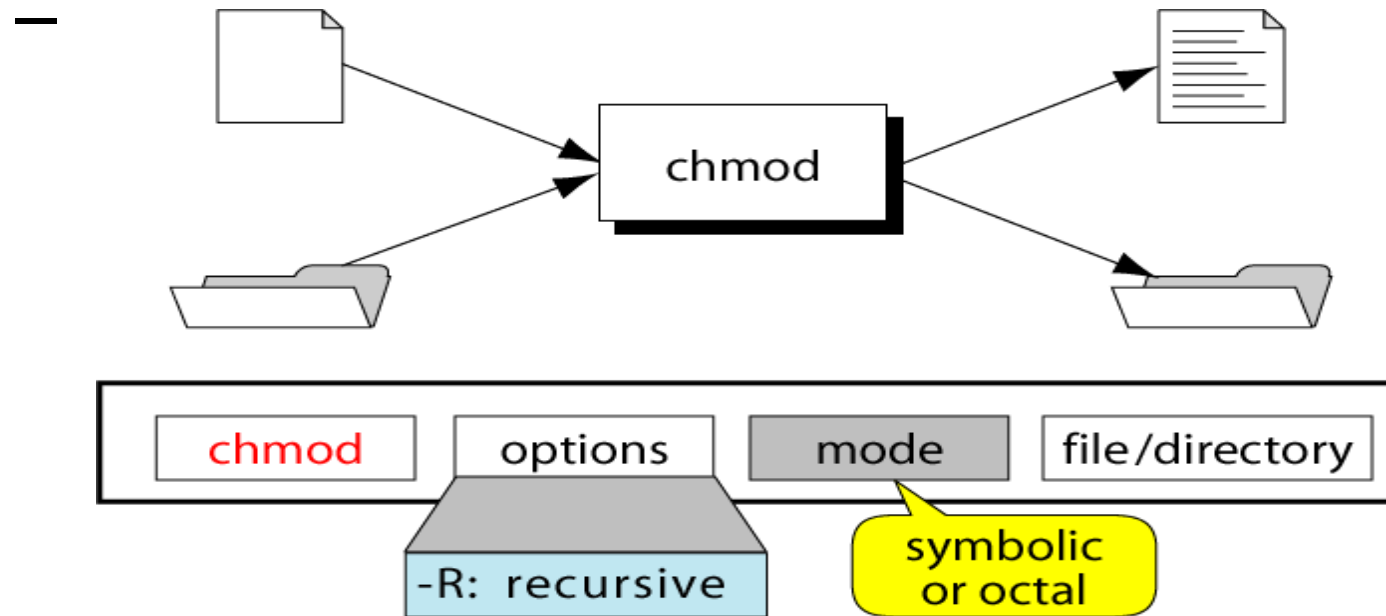
- Αλλαγή Δικαιωμάτων:



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



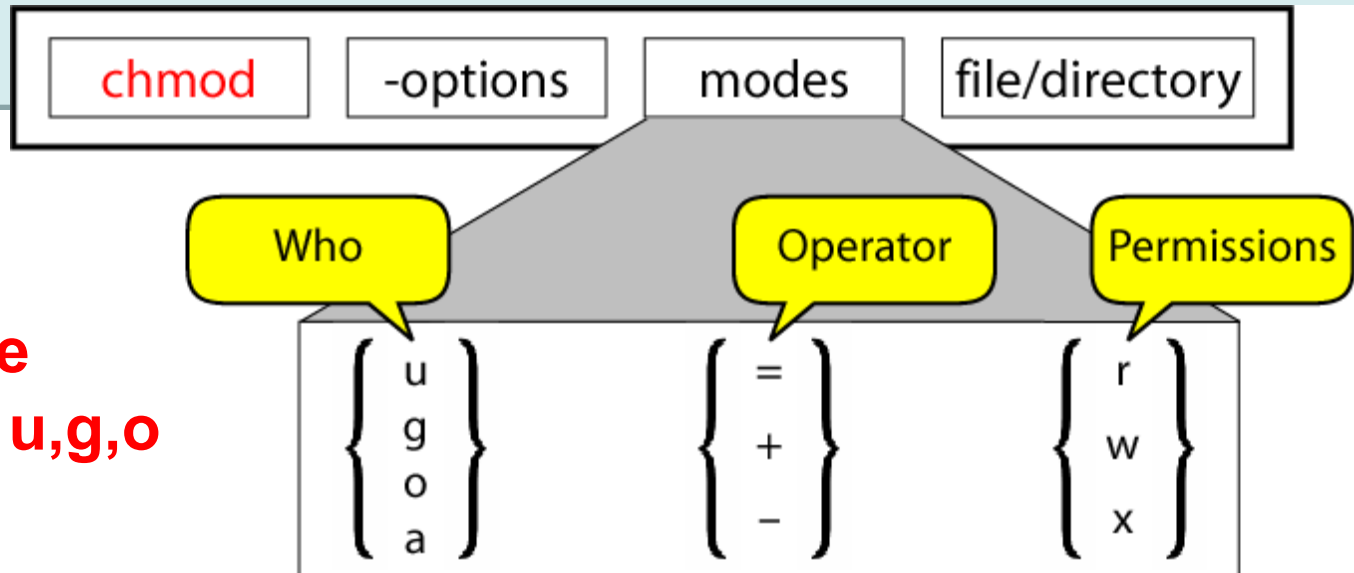
- Αλλαγή Δικαιωμάτων:
  - Εντολή **chmod** → **μόνον ο ιδιοκτήτης (ο su)**  
(και άλλα μέλη της ομάδας κάτω από ορθά δικαιώματα μπορούν να το πράξουν!)



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Δικαιωμάτων: Συμβολική μορφή
  - Εντολή *chmod*



**Chmod +x file**  
**> Adds +x to u,g,o**

Example

```
chmod u=rwx,g+w,o-w memo.doc
```



# chmod +x



```
$touch testfile
```

```
$ls -al testfile
```

```
-rw-r--r-- 1 dzeina staff 0 Feb 14 11:27 testfile
```

```
$chmod +x testfile
```

```
$ls -al testfile
```

```
-rwxr-xr-x 1 dzeina staff 0 Feb 14 11:27 testfile
```

```
$chmod = testfile
```

```
$ ls -al testfile
```

```
------ 1 dzeina staff 0 Feb 14 11:27 testfile
```

```
$chmod 777 testfile
```

```
$ ls -al testfile
```

```
-rwxrwxrwx 1 dzeina staff 0 Feb 14 11:27 testfile
```

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Δικαιωμάτων: Συμβολική μορφή  
– Εντολή *chmod*

*chmod* who operation permissions filename

*u* for user  
*g* for group  
*o* for others  
*a* for all

*+* for add  
*-* for remove  
*=* for assign  
(set)

*r* for read  
*w* for write  
*x* for execute

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Δικαιωμάτων: Συμβολική μορφή  
– Παράδειγμα:

```
bash-3.1$ ls -l test.txt  
-rw-r--r-- 1 cspgcc1 cspg 0 Jan 30 19:38 test.txt
```

Αλλαγή των δικαιωμάτων πρόσβασης έτσι ώστε **όλοι να μπορούν να το διαβάζουν** και να το **εκτελούν** και **μόνο ο ιδιοκτήτης** και η **ομάδα να μπορούν να γράφουν** σ' αυτό (**rwX|rwX|r-X**):

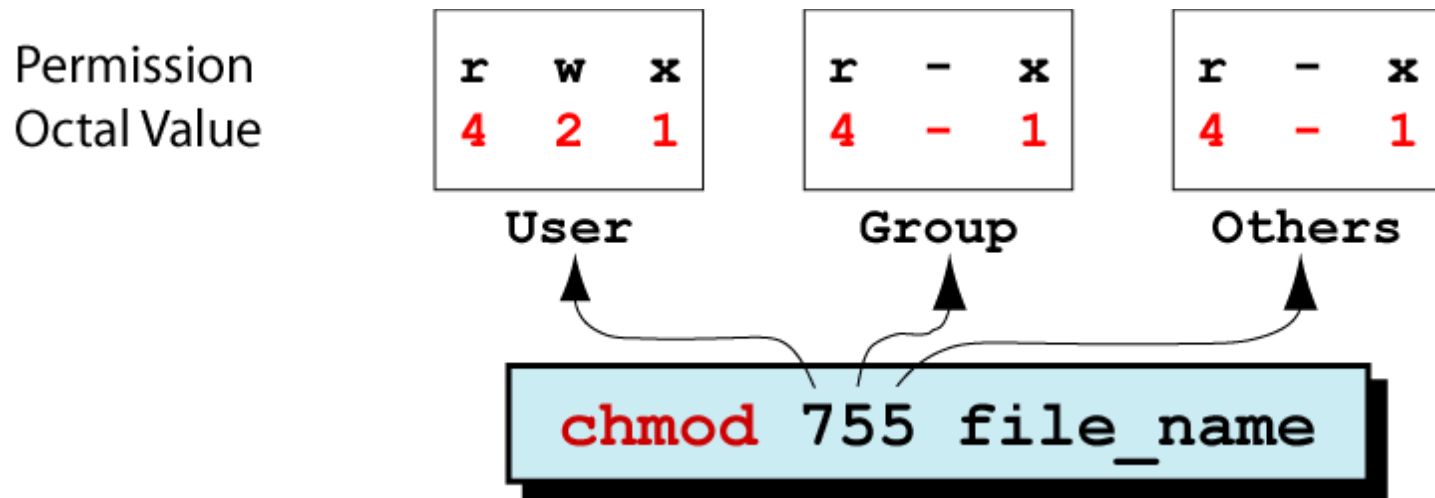
```
bash-3.1$ chmod ug=rwx,o+x test.txt  
bash-3.1$ ls -l test.txt  
-rwxrwxr-x 1 cspgcc1 cspg 0 Jan 30 19:40 test.txt
```

- **chmod o= test.txt** → αφαιρεί τα δικαιώματα από O group

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Δικαιωμάτων: Οκταδική μορφή



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Ιδιοκτησίας

- Εντολή **chown** *<new\_owner> <filename>*

- Αλλαγή ιδιοκτητησίας ενός αρχείου → **μόνον ο ιδιοκτήτης (και ο su) μπορεί να το πράξει!**

- Με την αλλαγή, ο νέος ιδιοκτήτης είναι ο μόνος που μπορεί να δώσει τα δικαιώματα πίσω (και ο su).

**chown root /directory**

Change the owner of /directory to "root". The group of /directory is changed to root's default group (i.e., root).

**chown root:staff /directory**

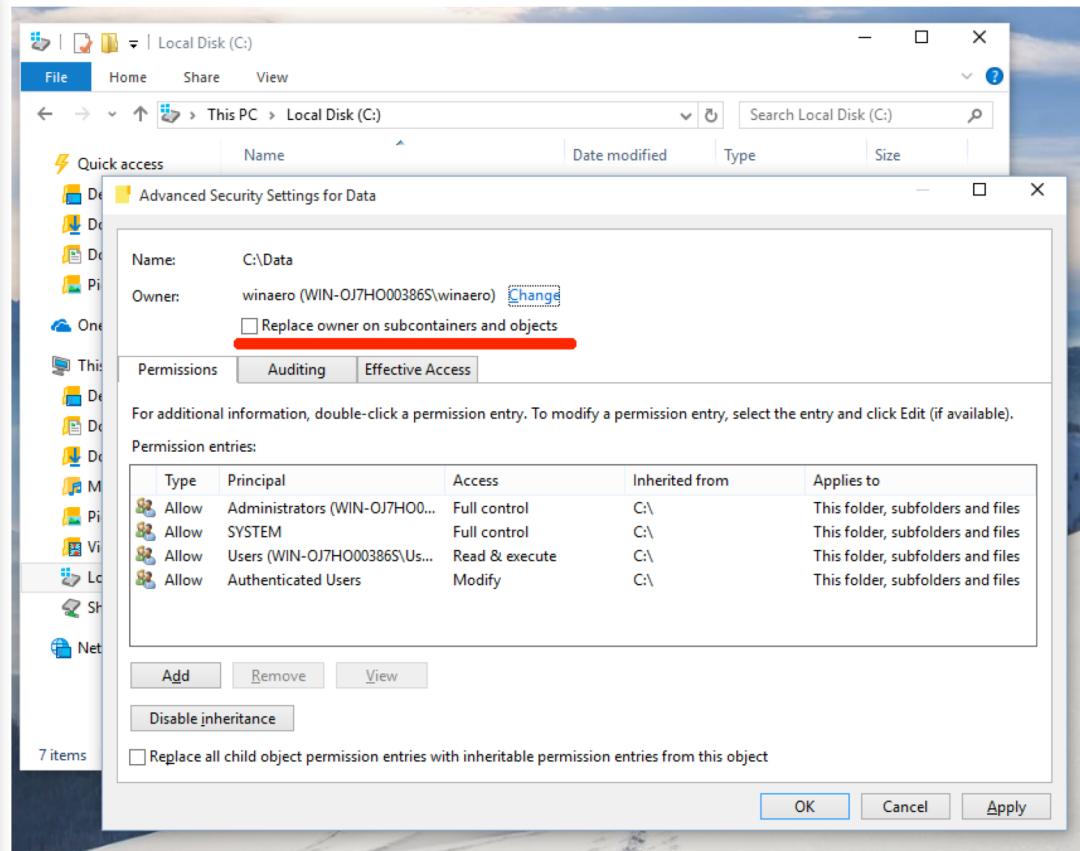
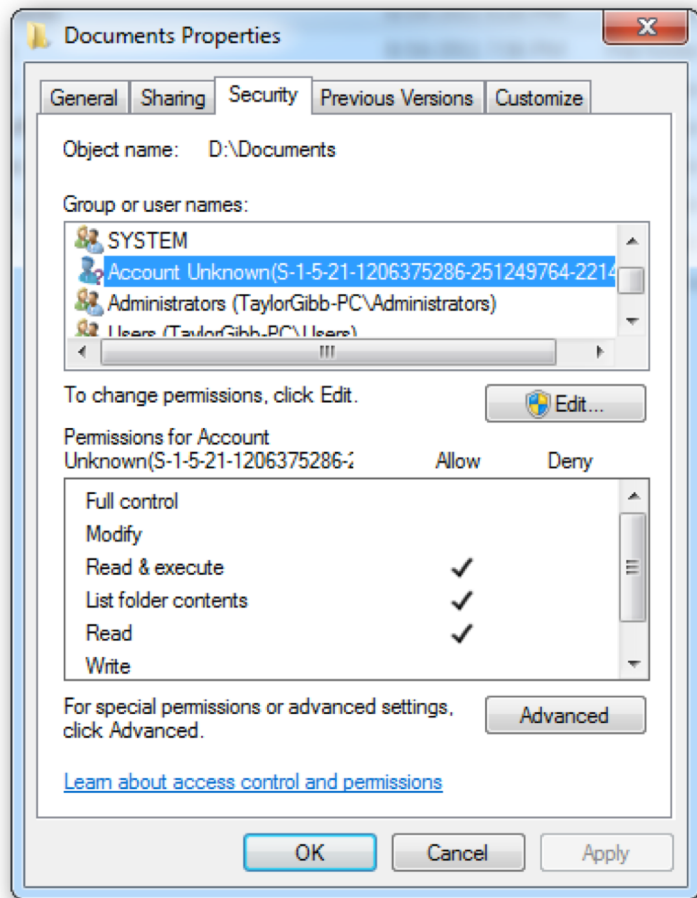
Likewise, but explicitly change the group of /directory to "staff". (recall that a user might belong to several groups)

**chown -hR root /directory**

Change the owner of /directory **recursively (-R)** to "root" (including the **traversal of symbolic links. Used**

**(-h) to exclude them.**)

# Taking Ownership in Windows (Fixing Orphaned Users)



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Αλλαγή Ομάδας

- Εντολή **chgrp** *<new\_group> <filename>*

- Αλλαγή ομάδας (effective group) ενός αρχείου σε μια άλλη ομάδα στην οποία ανήκει ο χρήστης
    - Μπορεί να επιτευχθεί και με την chown

```
bash-3.1$ ls -l test.c
```

```
-rw-r--r-- 1 cchrys tspecial 55 Mar 17 2015 test.c
```

```
bash-3.1$ groups cchrys
```

```
cchrys : tspecial ep1001 csphd visiting ep1371
```

```
bash-3.1$ chgrp visiting test.c
```

```
bash-3.1$ ls -l test.c
```

```
-rw-r--r-- 1 cchrys visiting 55 Mar 17 2015 test.c
```

```
bash-3.1$ groups cchrys
```

```
cchrys : tspecial ep1001 csphd visiting ep1371
```

**Δείτε Επόμενη Διαφάνεια για βασική & συμπληρωματική ομάδα!**

# Primary / Supplementary Groups (groups, id)



- Ανά πάσα στιγμή, ένας UNIX χρήστης φέρει ένα user id (uid) και ένα group id (gid).
- Παράλληλα, ένας χρήστης μπορεί να ανήκει και σε άλλα groups, το πρώτο (default) εκ των οποίων ονομάζεται **βασική ομάδα (Primary Group)**, ενώ τα υπόλοιπα **συμπληρωματικές ομάδες (Supplementary Groups)**

**\$ groups # print the groups a user is in**

```
faculty epl1371 epl132 anyplacegrp epl646 colloqgrp crowdgrp smartpgrp  
smartgrp smartlgrp smartnet tvmgrp
```

**\$ id # print group names and their group IDs**

```
uid=1240(dzeina) gid=231(faculty)  
groups=231(faculty),306(epl1371),314(epl132),348(anyplacegrp),411(epl646)  
,426(colloqgrp),446(crowdgrp),453(smartpgrp),466(smartgrp),483(smartlgrp)  
,488(smartnet),505(tvmgrp)
```

**\$ newgrp epl1371 # αλλαγή βασικής ομάδας**

```
uid=1240(dzeina) gid=306(epl1371)  
groups=231(faculty),306(epl1371),314(epl132),348(anyplacegrp),411(epl646),  
426(colloqgrp),446(crowdgrp),453(smartpgrp),466(smartgrp),483(smartlgrp),  
488(smartnet),505(tvmgrp)
```



# Άντληση Ταυτοτήτων από Εξυπηρετητή Ταυτοποίησης LDAP - Εντολή getent



## Παρουσίαση χρηστών (αντίστοιχο του /etc/passwd)

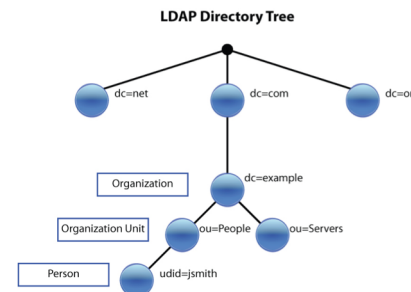
**\$getent passwd**

```
aandre28:*:2923:472:Andreou
Andre:/home/students/cs/2011/aandre28:/bin/bash
acosta01:*:2776:472:Andri
Costa:/home/students/cs/2011/acosta01:/bin/bash
ageorg35:*:2743:472:Anna
Georgiou:/home/students/cs/2011/ageorg35:/bin/bash
```

## Παρουσίαση στοιχείων ομάδας (αντίστοιχο του /etc/groups)

**\$getent group cs11**

**cs11:\*:472:**



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Δικαιώματα Πρόσβασης αρχείων και καταλόγων κατά τη δημιουργία τους
  - Εντολή **umask** (*Ορίζει τι δικαιώματα αφαιρούνται*)
  - Τα εξ' ορισμού (προεπιλεγμένα) δικαιώματα ενός δημιουργηθέντος αρχείου ή καταλόγου ρυθμίζονται αρχικά χρησιμοποιώντας μια μεταβλητή **3-ψηφίων** σε οκταδικό σύστημα, που ονομάζεται **user mask**.
  - Αυτό το **user mask** έχει ορισθεί από τον **system administrator** όταν δημιουργήθηκε ο λογαριασμός του κάθε χρήστη.
  - Το **user mask** περιέχει τις ρυθμίσεις σε οκταδικό για τα δικαιώματα πρόσβασης που ΑΦΑΙΡΟΥΝΤΑΙ από το μέγιστο όταν ένας **κατάλογος** ή **αρχείο** δημιουργείται.

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Μέγιστα Δικαιώματα **κατά την δημιουργία**:
  - **Καταλόγου**: 777 (δηλ., `rwxrwxrwx`)
  - **Αρχείου**: 666 (δηλ., `rw-rw-rw-`)
    - Π.χ., με “touch a” τα δικαιώματα του a είναι “666 – umask”
- Γιατί 666 σε αρχεία και 777 σε καταλόγους;
  - Το UNIX προσπαθεί να συμπεριφέρεται έξυπνα όσον αφορά τα **execute** δικαιώμα.
    - Πρακτικά τα αρχεία θεωρούνται ότι δεν έχουν ΠΟΤΕ execute δικαιώματα κατά τη δημιουργία τους.
    - Αφαιρώντας εξ'ορισμού το execute σε αρχεία επιτρέπει το σύστημα να έχει περισσότερη ασφάλεια.
    - Το execute σε καταλόγους δεν αφαιρείται εφόσον μπορεί να θέλουμε να παρέχουμε access by file name

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



## – Παράδειγμα

Το επιπλέον 0 θα εξηγηθεί σε λίγο

```
bash-3.1$ umask
```

**0022**

```
bash-3.1$ touch test.txt
```

```
bash-3.1$ ls -l test.txt
```

```
-rw-r--r-- 1 cchrys tspecial 0 Jan 31 06:27 test.txt
```

```
bash-3.1$ mkdir test-perm
```

```
bash-3.1$ ls -ld test-perm/
```

```
drwxr-xr-x 2 cchrys tspecial 4096 Jan 31 06:28 test-  
perm/
```

File

Directory

**666-022 = 644**

**777-022=755**

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



## – Παράδειγμα (συνέχεια)

```
bash-3.1$ umask 077 (group & others: NO permissions)
bash-3.1$ touch test2.txt
bash-3.1$ ls -l test2.txt
-rw----- 1 cchrys tspecial 0 Jan 31 06:59 test2.txt
bash-3.1$ mkdir test2-perm
bash-3.1$ ls -ld test2-perm (-d:show dir not content)
drwx----- 2 cchrys tspecial 4096 Jan 31 07:00 test2-
perm
```

File:  $666 - 077 = 600$

Directory:  $777 - 077 = 700$

## – Το **umask** ρυθμίζεται μια φορά και ισχύει μέχρι την αποσύνδεση του **session**.

- Κάθε φορά που συνδεόμαστε (log in) στο σύστημα, το **umask** κρατά την προεπλεγμένη του τιμή (αργότερα θα μιλήσουμε για το προφίλ)

# Ειδικά Δικαιώματα Πρόσβασης



- Σε πολλές περιπτώσεις προκύπτει η ανάγκη για προσωρινά αναβαθμισμένα δικαιώματα.
  - Π.χ., ο χρήστης dzeina:faculty να έχει δικαιώματα root:root κατά την εκτέλεση την εντολής /usr/bin/passwd
- Υπάρχουν οι ακόλουθοι τρόποι:
  1. Login as <newuser> (π.χ., **su – newuser**)
  2. Run command with sudo (π.χ., **sudo –u newuser /usr/bin/passwd**, requires the existence of configurations in /etc/sudoers/)
  3. Special Permissions on Executable (Ειδικά Δικαιώματα)
    - **Set User ID (SUID)** -- για εκτελέσιμα αρχεία
    - **Set Group ID (SGID)** -- για εκτελέσιμα αρχεία
    - **Sticky bit (STB)** -- για καταλόγους

# Ειδικά Δικαιώματα Πρόσβασης:

## Αρχεία



Ενδέχεται να είναι  
και στο τέλος

- Ρύθμιση Ειδικών Δικαιωμάτων
  - Χρήση της εντολής *chmod* σε οκταδική μορφή:
    - π.χ. *chmod 7777 filename*

suid	sgid	stb	r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1	4	2	1
7			7			7			7		
Special			user			group			others		

# Ειδικά Δικαιώματα Πρόσβασης: Αρχεία



- Ειδικά Δικαιώματα: **Set User ID (SUID)**

- **SUID** επιτρέπει στους χρήστες να εκτελέσουν ένα αρχείο και να γίνουν **προσωρινοί ιδιοκτήτες** (Owners) του αρχείου (κατά τη διάρκεια της εκτέλεσης).

- Στο Linux / Unix αγνοείται για καταλόγους (μόνο για αρχεία)

- **Παράδειγμα:** Η εντολή *passwd* ή *ping* με ιδιοκτήτη τον *root* έχει τις ακόλουθες ειδικές ρυθμίσεις:

```
bash-3.1$ ls -l /usr/bin/passwd
```

```
-r-s--x--x 1 root root 21944 Feb 12 2006 /usr/bin/passwd
```

```
bash-3.1$ ls -al /bin/ping
```

**SUID**

```
-rwsr-xr-x 1 root root 40760 Sep 26 2013 /bin/ping
```

- Όταν ένας χρήστης εκτελεί την εντολή *passwd*, ο χρήστης γίνεται προσωρινά ο «*root*» χρήστης για **όσο τρέχει η εντολή** (δηλ., η διεργασία θα έχει τα **ίδια δικαιώματα** όπως αυτά του **ιδιοκτήτη** του αρχείου)

- Θα δούμε σε λίγο πως τίθεται η επιλογή «s»

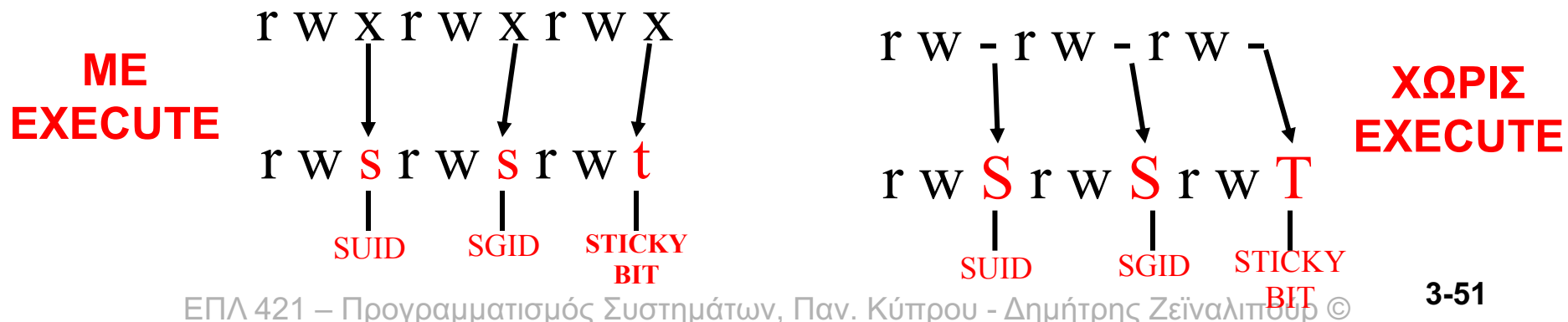


# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- **Παρουσίαση Ειδικών Δικαιωμάτων**

- Η κατάσταση των δικαιωμάτων πρόσβασης που εμφανίζεται με την εντολή «ls -l» **δεν έχει ξεχωριστό τμήμα για τα ειδικά δικαιώματα σε πολλές υλοποιήσεις** ☹.
- Επειδή τα ειδικά δικαιώματα απαιτούν συνήθως **«execute»**, καλύπτουν/αντικαθιστούν το **δικαίωμα execute** στην παρουσίαση της εντολής «ls -l».



# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Ειδικά Δικαιώματα: **Set Group ID (SGID)**
  - Η **SGID** κάνει τους χρήστες να **γίνουν μέλος της ομάδας** του γονικού καταλόγου
  - Χρήσιμο για δημιουργία κοινόχρηστης πρόσβασης σε άτομα από διαφορετικές ομάδες (π.χ., student, faculty)
- Υπάρχει κάποια ομάδα “GALL” με μέλη τους A:G1, B:G2, C:G3 (username:group)
- Ο κατάλογος /projects/ ανήκει στο GALL.

suid	sgid	stb
4	2	1
7		
Special		

Ο A:G1 εκτελεί «touch file» (δημιουργία αρχείου)

- Τώρα το **file** είναι του **A:G1** (δείτε ls -al file)

Ενώ εάν εκτελούσαμε το πιο κάτω πριν το touch:

- chmod **2777** /projects/

- Τότε το **file** θα είχε δικαιώματα **A:GALL**

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



## Παράδειγμα SGID

\$ls -ald anyplace/ # d shows directory not content

```
drwxrwsr-x 21 kgeorg10 anyplacegrp 4096 Jan  2 00:02 anyplace/
```

**Πλέον, ότι αρχεία δημιουργούνται ανήκουν στο group anyplacegrp στο οποίο ανήκουν και οι δυο χρήστες (βολικό για αλλαγές και από τους δυο)**

\$ ls -al

total 180

```
drwsrwsr-x 21 kgeorg10 anyplacegrp 4096 Jan  2 00:02 .
```

```
drwxr-xr-x 68 root    root      8192 Sep 30 10:02 ..
```

```
drwsr-sr-x  2 dzeina anyplacegrp 4096 Sep 19 14:57 architect
```

```
drwsr-sr-x  2 dzeina  anyplacegrp 4096 Sep 19 14:57 contact
```

```
-rwxrw-rw-  1 dzeina  anyplacegrp 181 Sep 18 13:55 contact.html
```

```
drwxrwxr-x  2 kgeorg10 anyplacegrp 4096 Nov 27 13:14 css
```

# Ιδιοκτησία και Δικαιώματα Πρόσβασης



- Ειδικά Δικαιώματα: **Sticky Bit (STB)**
  - Εάν εφαρμοστεί το **Sticky Bit** τότε η **διαγραφή αρχείων/καταλόγων** μπορεί να γίνει από ένα **χρήστη μόνο** σε αρχεία που έχει **προσθέσει ο ίδιος ο χρήστης**.

- Το *Sticky Bit* εκτελεί μια χρήσιμη λειτουργία στους καταλόγους, π.χ., στο /tmp (κοινόχρηστος χώρος)

- Παράδειγμα:

```
bash-3.1$ ls -ld /tmp
```

```
drwxrwxrwt 71 root root 16384 Jan 31 04:10 /tmp
```

→ Sticky Bit

- Εάν το /tmp ήταν απλά 777 τότε θα μπορούσε οποιοσδήποτε να διαγράψει καταλόγους/αρχεία άλλων χρηστών (όχι μόνο προσωπικούς).

# Προσθήκη Τοπικών Χρηστών (Εντολή useradd – LDAP luseradd / lgroupadd)



- **# less /etc/default/useradd**  
GROUP=100  
HOME=/home  
INACTIVE=-1  
EXPIRE=  
SHELL=/bin/bash  
SKEL=/etc/skel  
CREATE\_MAIL\_SPOOL=yes

To change the **default home directory** location for all new users  
# useradd -D -b /opt/users

To change the **default login shell**  
# useradd -D -s /bin/sh

Create multiple users with same UID

```
# useradd -o chrisk -u 501  
# useradd -o chris -u 501  
# useradd -o user -u 501
```

**Verify the UID of the newly create users**

```
# grep 501 /etc/passwd  
chrisk:x:501:501::/home/chrisk:/bin/sh  
chris:x:501:504::/home/chris:/bin/sh  
user:x:501:505::/home/user:/bin/sh
```

**Manually assign a UID to the user**  
# useradd -u 550 chrisk

**Add user to different primary group**

```
# useradd -g admin,dba chris
```