

Εντολές

pwd

Η εντολή **pwd** (print working directory) εμφανίζει τον ενεργό ή τρέχοντα κατάλογο ή κατάλογο εργασίας μας.

mkdir

Με την εντολή **mkdir** (make directory) μπορούμε να δημιουργήσουμε έναν ή περισσότερους καταλόγους.

Π.χ

mkdir katalog1 δημιουργεί υποκατάλογο στον τρέχοντα

rmdir

Με την εντολή **rmdir** (remove directory) μπορούμε να διαγράψουμε καταλόγους. Ο κατάλογος προς διαγραφή πρέπει να είναι κενός, δηλ. να μην περιέχει αρχεία και άλλους καταλόγους και να μην είναι ο τρέχων κατάλογος.

rmdir katalog1

cd

Με την εντολή *cd* (*change directory*) μπορούμε να αλλάξουμε τον τρέχοντα κατάλογο ή κατάλογο εργασίας μας. Μπορούμε να πάμε απευθείας σ' έναν συγκεκριμένο κατάλογο γράφοντας το όνομα διαδρομής ή το όνομα του καταλόγου, ως εξής :

cd John

Με την εντολή **cd ..** πηγαίνουμε στον γονικό κατάλογο του τρέχοντος καταλόγου (του καταλόγου δηλαδή στον οποίο βρισκόμαστε).

Με την εντολή **cd /** πηγαίνουμε στον αρχικό κατάλογο του συστήματος ή τη ρίζα (root).

Με την εντολή **cd** χωρίς ορίσματα μπορούμε να πάμε στον προσωπικό μας κατάλογο, σ' αυτόν δηλαδή που μας έχει εκχωρηθεί από τον διαχειριστή του Linux του συστήματος, για να αποθηκεύουμε τα αρχεία μας ως χρήστης.

ls

Η εντολή **ls** (list directory) εμφανίζει τα περιεχόμενα του τρέχοντος καταλόγου.

-l μπορούμε να δούμε όλα τα στοιχεία των αρχείων και των καταλόγων του τρέχοντος καταλόγου. Ο πρώτος χαρακτήρας μπορεί να έχει την τιμή **d**, που σημαίνει ότι πρόκειται για κατάλογο, ή την τιμή **-**, που σημαίνει ότι πρόκειται για αρχείο.

Οι επόμενοι 9 χαρακτήρες, που ανήκουν στην πρώτη στήλη, εμφανίζουν τα δικαιώματα read, write και execute για ένα αρχείο ή κατάλογο για τις τρεις κατηγορίες χρηστών (κάτοχος, ομάδα και άλλοι). Στη δεύτερη

στήλη εμφανίζεται το πλήθος των συνδέσμων (links) του αρχείου, μετά το όνομα του κατόχου του (owner), μετά το όνομα της ομάδας χρηστών στην οποία ανήκει το αρχείο, μετά το μέγεθος του αρχείου σε bytes, η ημερομηνία και ώρα δημιουργίας ή τελευταίας τροποποίησής του και τέλος το όνομα του αρχείου.

Στο επόμενο παράδειγμα εκτελούμε την εντολή από το root directory:

```
$ ls -l
```

σύνολο 213

```
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:10 bin
dr-xr-xr-x  1 Owner None    0 Νοε  30 22:38 cygdrive
-rwxr-xr-x  1 Owner root   57 Οκτ  20 11:11 Cygwin.bat
-rw-r--r--  1 Owner root 157097  20 11:11 Cygwin.ico
-rw-r--r--  1 Owner root  53342  20 11:11 Cygwin-Terminal.ico
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:11 dev
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:11 etc
drwxrwxrwt+ 1 Owner None    0 Νοε   6 12:28 home
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:10 lib
dr-xr-xr-x  9 Owner None    0 Νοε  30 22:38 proc
drwxrwxrwt+ 1 Owner None    0 Οκτ  20 11:11 tmp
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:07 usr
drwxr-xr-x+ 1 Owner None    0 Οκτ  20 11:07 var
```

χαρακτήρες υποκατάστασης (μπαλαντέρ): (? *)

* Χρησιμοποιείται στην αρχή ενδιάμεσα ή στο τέλος του ονόματος.

Υποκαθιστά στο όνομα εκεί που εμφανίζεται οποιοδήποτε πλήθος χαρακτήρων.

? Υποκαθιστά ένα χαρακτήρα στη θέση που εμφανίζεται.

mv

Με την εντολή **mv** (move) μπορούμε να μετονομάσουμε αρχεία και καταλόγους ή να τα μετακινήσουμε σ' άλλη θέση ή να τα μετονομάσουμε και να τα μετακινήσουμε σε άλλη θέση με την ίδια εντολή.

Η επόμενη εντολή μετονομάζει το αρχείο file1 σε file2.

mv file1 file2

Η επόμενη εντολή αλλάζει το όνομα του καταλόγου katalog1 σε katalog2, ανεξάρτητα από το αν αυτός είναι κενός ή όχι.

mv katalog1 katalog2

Η επόμενη εντολή μετακινεί το αρχείο file1 από τον κατάλογο d1 στον κατάλογο d2.

mv /home/user1/d1/file1 /home/user1/d2/file1

Η επόμενη εντολή μετακινεί το αρχείο file1 από τον κατάλογο d1 στον κατάλογο d2 και συγχρόνως αλλάζει το όνομα του αρχείου από file1 σε file2.

mv /home/user1/d1/file1 /home/user1/d2/file2

ΠΡΟΣΟΧΗ. Η εντολή `mv` αντικαθιστά χωρίς να ζητάει επικύρωση αρχεία και καταλόγους που έχουν το ίδιο όνομα με το νέο όνομα που επιλέγουμε.

cp

Με την εντολή **`cp`** (`copy`) μπορούμε να αντιγράψουμε αρχεία και καταλόγους. Η απλούστερη μορφή της εντολής φαίνεται παρακάτω, όπου δημιουργούμε ένα αντίγραφο του αρχείου `file1` με το όνομα `file2` και στον ίδιο κατάλογο.

`cp file1 file2`

Η επόμενη εντολή αντιγράφει όλα τα αρχεία του καταλόγου `catalog1` στον κατάλογο `catalog2`.

`cp catalog1/* catalog2`

rm

Με την εντολή **`rm`** (**`remove`**) μπορούμε να διαγράψουμε οριστικά αρχεία (δεν μπορούμε να επαναφέρουμε).

Ποιο το αποτέλεσμα των επόμενων εντολών;

`rm file`

`rm file1 file2`

`rm file*`

rm file??

Ιδιοκτήτες και ομάδες αρχείων

Κάθε αρχείο/κατάλογος έχει έναν ιδιοκτήτη (owner), τον χρήστη που το δημιούργησε, επιπλέον το αρχείο ανήκει σε μια ομάδα (group), που είναι ομάδα στην οποία ανήκει και ο ιδιοκτήτης. Στις ρυθμίσεις του ιδιοκτήτη και της ομάδας περιλαμβάνονται και εκείνες που αφορούν τις άδειες (permissions), που ελέγχουν την πρόσβαση σε ένα αρχείο ή κατάλογο.

Παράδειγμα:

Όταν δημιουργείται ένα νέο αρχείο, ως ιδιοκτήτης καθορίζεται ο χρήστης που το δημιούργησε. Το αρχείο αποκτά το userID του χρήστη που το δημιούργησε με groupID για το αρχείο εκείνο της ομάδας στην οποία ανήκει ο χρήστης. Να σημειώσουμε ότι ο χρήστης μπορεί να ανηκει σε περισσότερες από μία ομάδες και να εναλλάσσεται μεταξύ τους.

Άδειες αρχείων

Υπάρχουν τρία είδη αδειών πρόσβασης σε αρχείο:

r

Άδεια ανάγνωσης (read). Επιτρέπεται η ανάγνωση των περιεχομένων του αρχείου.

W

Άδεια εγγραφής (write). Επιτρέπεται η τροποποίηση των περιεχομένων του αρχείου.

X

Άδεια εκτέλεσης (execute). Επιτρέπεται η εκτέλεση του αρχείου ως πρόγραμμα. (Εννοείται ότι έχει νόημα η απόδοση αυτής της άδειας όταν το αρχείο περιλαμβάνει εντολές προγράμματος).

Άδειες καταλόγων

Οι αντίστοιχες άδειες για καταλόγους:

r

Επιτρέπεται η εμφάνιση των περιεχομένων του καταλόγου, δηλαδή τα ονόματα των αρχείων και καταλόγων που περιέχει.

W

Επιτρέπεται η εισαγωγή και διαγραφή αρχείων στον κατάλογο.

X

Επιτρέπεται η προσπέλαση των αρχείων στον κατάλογο.

Για κάθε αρχείο καθορίζεται ένα σύνολο αδειών πρόσβασης που ελέγχουν ποιος μπορεί να χειρίζεται το αρχείο.

Υπάρχουν τρία σύνολα αδειών πρόσβασης:

u Άδειες χρήστη (user). Ελέγχει την πρόσβαση από τον ιδιοκτήτη του αρχείου.

g Άδειες ομάδας (group). Ελέγχει την πρόσβαση από τους χρήστες της ομάδας.

o Άδειες άλλων (others). Ελέγχει την πρόσβαση για οποιοδήποτε άλλο (που δεν είναι ιδιοκτήτης ή μέλος ομάδας).

Θα εξετάσουμε το θέμα των αδειών αρχείου μέσα από ένα παράδειγμα:
Έστω ότι δημιουργήσαμε το αρχείο file1 με την εντολή `cat > file1` και έχουμε εισαγάγει κάποιο περιεχόμενο σε αυτό. Στη συνέχεια εκτελούμε την εντολή `ls`:

```
ls -la file1
-rw-r--r-- 1 John omada1 54 Φεβ 28 15:34 file1
```

Οι πληροφορίες αυτές που αφορούν το αρχείο file1 αναλύονται στον επόμενο πίνακα:

-	rw-	r--	r--	1	John	omada1	54	28 Φεβ 15:34	file1
τύπος αρχείου	user	group	others	αρ. συμβολικών δεσμών	ιδιοκτήτης (user)	ομάδα	μέγεθος αρχείου	ημερομηνία/ώρα τροποποίησης	όνομα αρχείου

Τύποι αρχείου:

κανονικό αρχείο (-)

αρχείο καταλόγου (d)

σύνδεσμος (l)

Υπάρχουν και τύποι αρχείων που αναφέρονται σε ειδικά αρχεία του

συστήματος που βρίσκονται στον κατάλογο /dev:

για συσκευή block (b), για συσκευή χαρακτήρων (c), κ.α

Οι άδειες αρχείου/καταλόγου εμφανίζονται με την εξής σειρά: rwx. Η

εμφάνιση παύλας στη θέση κάποιας άδειας σημαίνει ότι η συγκεκριμένη

άδεια δεν έχει δοθεί.

Αλλαγή αδειών σε αρχείο/κατάλογο:

Η εντολή **chmod** (change mode) αλλάζει τις άδειες πρόσβασης ενός αρχείου ή καταλόγου.

Τροποποίηση αδειών με χρήση της συμβολικής μορφής της chmod

Για το προηγούμενο παράδειγμα:

chmod u+x file1

δίνει στον ιδιοκτήτη του αρχείου, επιπλέον των αδειών που ήδη έχει και άδεια εκτέλεσης του αρχείου.

```
ls -la file1
```

```
-rwxr--r-- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod g+wx file1

δίνει στην ομάδα, επιπλέον των αδειών που ήδη έχει και τις άδειες εγγραφής και εκτέλεσης του αρχείου.

```
ls -la file1  
-rwxrwxr-- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod g-x file1

αφαιρεί από την ομάδα την άδεια εκτέλεσης του αρχείου, χωρίς να αλλάξει τις άλλες άδειες που έχουν δοθεί.

```
ls -la file1  
-rwxrw-rw- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod go-w file1

αφαιρεί από την ομάδα και από τα άτομα που δεν ανήκουν σε ομάδα την άδεια τροποποίησης εγγραφών στο αρχείο, χωρίς να αλλάξει τις άλλες άδειες που έχουν δοθεί.

```
ls -la file1  
-rwxr--r-- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod ugo+rwx file1

Δίνει σε όλους (u, g, o) όλες τις άδειες.

```
ls -la file1  
-rwxrwxrwx 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod o=r file1

Επανακαθορίζει τις άδειες που αφορούν την πρόσβαση για άτομα εκτός ομάδων, δίνοντάς τους τις άδειες που αναφέρονται μετά το =

```
ls -la file1
```

```
-rwxrwxr-- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod go= file1

```
ls -la file1
```

```
-rwx----- 1 John omada1 54 Φεβ 28 15:34 file1
```

chmod a=r file1

Επανακαθορίζει για όλους (u, g, o) τις άδειες, δίνοντάς τους τις άδειες που αναφέρονται μετά το =

```
ls -la file1
```

```
-r--r--r-- 1 John omada1 54 Φεβ 28 15:34 file1
```

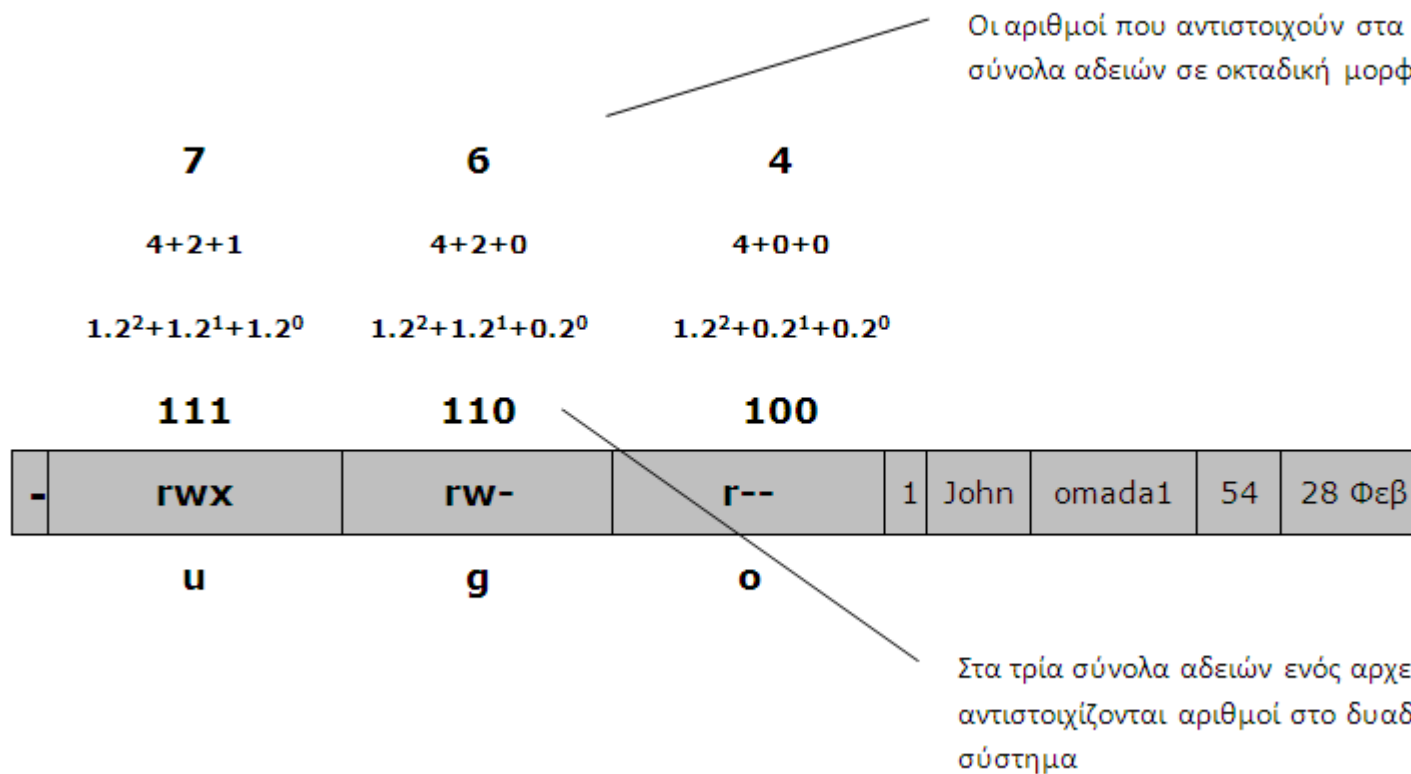
chmod a=rwx file1

Επανακαθορίζει για όλους (u, g, o) τις άδειες δίνοντας σε όλους τις άδειες που αναφέρονται μετά το =

```
ls -la file1
```

```
-rwxrwxrwx 1 John omada1 54 Φεβ 28 15:34 file1
```

Επανακαθορισμός αδειών με χρήση της αριθμητικής μορφής της chmod



Η εντολή που επανακαθορίζει τις άδειες σύμφωνα με το παραπάνω υπόδειγμα συντάσσεται ως εξής:

chmod 764 file1

Πρακτικά:

Σε κάθε σύνολο αδειών αντιστοιχεί ένας αριθμός (0-7), που η τιμή του προκύπτει από την πρόσθεση των τιμών των αντιστοιχών αδειών του συνόλου.

Οι τιμές που μπορεί να πάρουν οι άδειες είναι:

• 4 = άδεια ανάγνωσης

• 2 = άδεια εγγραφής

• 1 = άδεια εκτέλεσης

• 0 = καμία άδεια

Μερικά παραδείγματα ακόμη:

r	w	x	-	-	x	-	-	-
(4 + 2 + 1)	(0 + 0 + 1)	(0 + 0 + 0)						
7	1	0						

chmod 710 file1

r	-	x	r	-	x	-	-	x
(4 + 0 + 1)	(4 + 0 + 1)	(0 + 0 + 1)						
5	5	1						

chmod 551 file1