

2.Shell, users, groups και privileges

2.0 Εισαγωγή

Όλα τα σύγχρονα λειτουργικά συστήματα έχουν γραφικό περιβάλλον που τα κάνουν πολύ φιλικά προς το χρήστη. Το ίδιο ισχύει και στις διανομές του Linux, όπως είπαμε και στην εισαγωγή έχουμε ένα σύνολο από desktop environments από όπου μπορούμε να διαλέξουμε πιο θα χρησιμοποιήσουμε. Πολλές φορές θα πρέπει να χρησιμοποιήσουμε όμως ένα τερματικό για να κάνουμε κάποιες λειτουργίες, κυρίως όταν πρέπει να αλλάξουμε αρχεία που πρόσβαση έχει μόνο ο διαχειριστής του συστήματος. Αυτό γίνεται ακόμα πιο έντονο σε στις εκδόσεις server όπου πολλές διανομές δεν έχουν εγκατεστημένο κάποιο desktop environment, κάτι που γίνεται για εξοικονόμηση πόρων του συστήματος.

2.1 Κέλυφος(shell)

Το shell είναι η διεπαφή(interface) επικοινωνίας του χρήστη με τον πυρήνα του λειτουργικού συστήματος. Το κέλυφος δέχεται εντολές τις οποίες μεταφράζει και στην συνέχεια εκτελεί. Στο Linux υπάρχουν αρκετά διαφορετικά κελύφη που μπορούμε να χρησιμοποιήσουμε, με διαφορετικά χαρακτηριστικά και δυνατότητες το καθένα.

Μερικά shells που μπορούμε να χρησιμοποιήσουμε είναι :

i) bash - Bourne Again Shell

Το Bash είναι ακρωνύμιο του Bourne Again Shell, κάτι που δείχνει ότι έχει βασιστεί στο Bourne Shell. Το Bourne Shell είναι το κέλυφος το οποίο έγραψε ο Steve Bourne για το Unix Version 7. Το bash συνδιάζει αρκετα χαρακτηριστικά από το Bourne shell, Korn Shell και το C Shell. Επειδή το bash είναι το default shell στις περισσότερες διανομές Linux, στο κεφάλαιο αυτό θα περιγράψουμε την χρήση του bash καθώς και σε επόμενο κεφάλαιο θα περιγράψουμε την δημιουργία scripts για το bash shell. Το Bash είναι ένα από τα σημαντικότερα προγράμματα του GNU Project.

ii) tcsh/csh - C Shell

Το C Shell γράφτηκε από τον Bill Joy για το BSD Unix ως το default shell. Η διαφορά με το Bourne Shell είναι ότι το C Shell, όπως δίνει και η ονομασία του, έχει παρόμοια σύνταξη με την γλώσσα προγραμματισμού C. Επίσης για την εποχή του πρόσφερε αρκετές καινούριες δυνατότητες όπως το ιστορικό εντολών. Στις διανομές Linux βρίσκουμε το shell tcsh που είναι μια βελτιωμένη έκδοση του csh.

ii) ash - Almquist Shell

Το ash δημιουργήθηκε από τον Keneth Almquist για το NetBSD, αν και δεν προσφέρει τις δυνατότητες που προσφέρει το bash, έχει μικρότερο footprint κάτι που το κάνει ιδιαίτερα χρήσιμο σε embedded συστήματα που έχουν περιορισμένους πόρους.

Μία έκδοση του ash για το Linux είναι η dash(Debian Almquist Shell). Όπως γίνεται εύκολα κατανοητό από το dash δημιουργήθηκε από την κοινότητα του Debian Linux. Επειδή είναι πιο μικρό σαν πρόγραμμα το dash χρησιμοποιείται από τις διανομές Debian και Ubuntu κατά την διάρκεια της εκκίνησης του συστήματος, κάτι που οδηγεί σε μικρότερους χρόνους εκκίνησης.

iii) ksh - Korn Shell

Το Korn Shell δημιουργήθηκε από τον David G. Korn στο Bell Laboratories της AT&T και χρησιμοποιήθηκε ως το default shell στο Unix System V. Το ksh συνδυάζει τις δυνατότητες που προσφέρουν το Bourne Shell και το C Shell καθώς επίσης προσφέρει και καινούριες δυνατότητες όπως η εύκολη δημιουργία νέων εντολών για το shell.

iv)zsh

Το zsh είναι ένα ακόμα shell που μοιάζει αρκετά με το ksh αλλά προσφέροντας αρκετές δυνατότητες που έχουν πιο σύγχρονα shell όπως το bash.

Terminal Window

Επειδή όλες οι διανομές χρησιμοποιούν κάποιο παραθυρικό περιβάλλον, έχουν δημιουργηθεί εφαρμογές που προσφέρουν πρόσβαση σε κάποιο shell. Το πιο απλό είναι το Xterm που έχει δημιουργηθεί για το X Window System. Ανάλογα με ποιο desktop environment έχετε επιλέξει θα χρησιμοποιήσετε άλλο terminal window. Το KDE προσφέρει το Konsole ενώ το Gnome προσφέρει το Gnome Terminal. Οι εφαρμογές Konsole και Gnome Terminal είναι πιο εξελιγμένες από το xterm και προσφέρουν την δυνατότητα αποκοπεί/αντιγραφείς επικόλλησης κ.α.



Εικόνα 2.1

Στην εικόνα 2.1 βλέπουμε το Gnome Terminal. Στο τερματικό παρατηρούμε το command prompt που είναι της μορφής `user@nodename:cwd>` στην συγκεκριμένη περίπτωση ο χρήστης(user) είναι `medialab`, στη δικιά σας περίπτωση θα είναι το όνομα του χρήστη που έχετε δημιουργήσει κατά την εγκατάσταση και είναι πάντα το όνομα του χρήστη του λογαριασμού με τον οποίο έχετε κάνει login.

Το `cwd`(current working directory) είναι το path που δείχνει τον φάκελο στον οποίο βρισκόμαστε, πάντα όταν ανοίγουμε ένα τερματικό ξεκινάει στον home φάκελο του χρήστη με τον οποίο έχουμε συνδεθεί. Ο φάκελος του χρήστη συμβολίζεται με το σύμβολο `~` που στην συγκεκριμένη περίπτωση είναι το `/home/medialab` όπως φαίνεται και στην εικόνα 2.2.



Εικόνα 2.2

2.1.1 Είδη Κελύφων

Υπάρχουν δύο είδη κελύφων τα login shell και τα non-login shell. Το login shell ανοίγει όταν κάνουμε login στο σύστημα χρησιμοποιώντας το username και το password. Τα non-login shell μπορούμε να τα ανοίξουμε αφού έχουμε εισέλθει στο σύστημα για να κάνουμε κάποια εργασία ή να εκτελέσουμε κάποιο πρόγραμμα. Κάθε φορά που δουλεύουμε με ένα λογαριασμό(χρήστη), θα έχουμε ένα login shell και θεωρητικά απεριόριστο δυνατό αριθμό από non-login shell.

2.1.2 Αρχεία Ρύθμισης Κελύφων

Για καθένα από τα δύο είδη κελύφων έχουμε και ένα αρχείο στο οποίο μπορούμε να περάσουμε ρυθμίσεις, κάθε φορά που ξεκινάει κάποιο κέλυφος, διαβάζει το αντίστοιχο αρχείο με τις ρυθμίσεις. Το login shell διαβάζει το αρχείο που καθορίζεται από το path `~/profile` και τα non-login shell διαβάζουν το αρχείο που καθορίζεται από το path `~/bashrc`.

Στα αρχεία αυτά μπορούμε να περάσουμε εντολές οι οποίες θα εκτελούνται κατά την εκκίνηση του κελύφους, αλλά και aliases για εντολές και environmental variables τα οποία θα τα δούμε παρακάτω. Στη διανομή OpenSUSE το `~/bashrc` γίνεται source στο `~/profile`, δηλαδή οι ρυθμίσεις του bashrc

γίνονται και στο login shell, με αυτό τον τρόπο περνάμε αλλαγές μόνο στο bashrc.

2.1.3 Aliases

Τα aliases μας επιτρέπουν να εκτελέσουμε μια εντολή με διαφορετικό όνομα. Ένα alias συνήθως χρησιμοποιείται για να εκτελείται μια εντολή με συγκεκριμένες επιλογές (switches). Για να θέσουμε κάποιο alias χρησιμοποιούμε την εντολή alias όνομα=εντολή. Για να ισχύουν τα aliases κάθε φορά που ανοίγουμε κάποιο κέλυφος τότε θα περάσουμε τα aliases στο αρχείο bashrc.

2.1.4 Ολοκλήρωση εντολών

Το κέλυφος bash έχει την δυνατότητα, ελέγχοντας το τι έχουμε πληκτρολογήσει, να βρίσκει την εντολή που θέλουμε και να την συμπληρώνει, σε περίπτωση που είναι μια, ή να μας εμφανίζει ένα σύνολο από πιθανές εντολές. Η δυνατότητα αυτή δεν ισχύει μόνο για το όνομα της εντολής αλλά και όταν πληκτρολογούμε μια διαδρομή (path) προς ένα φάκελο ή ένα αρχείο. Η ολοκλήρωση εντολών γίνεται με το πλήκτρο Tab.

2.1.5 Ιστορικό εντολών

Εκτός από την ολοκλήρωση εντολών το bash έχει και άλλη μια πολύ χρήσιμη δυνατότητα, το ιστορικό εντολών. Το bash καθώς εκτελούμε εντολές τις αποθηκεύει στο αρχείο .bash_history. Πατώντας το πάνω βελάκι εμφανίζεται ή προηγούμενη εντολή που είχαμε εκτελέσει.

Η εντολή history μας εκτυπώνει όλο το ιστορικό εντολών αριθμημένο, δηλαδή κάθε εντολή έχει ένα μοναδικό αύξων αριθμό. Γνωρίζοντας το αύξων αριθμό μπορούμε να εκτελέσουμε την συγκεκριμένη εντολή πληκτρολογώντας !n όπου n είναι ο αύξων αριθμός.

Επίσης με έναν text editor μπορούμε να ανοίξουμε και να επεξεργαστούμε το .bash_history. Το ιστορικό εντολών μαζί με την ολοκλήρωση εντολών κάνουν το bash εύκολο αλλά και γρήγορο στη χρήση του.

2.1.6 Μεταβλητές Περιβάλλοντος

Οι μεταβλητές περιβάλλοντος (environmental variables) χρησιμοποιούνται για να αποθηκευθούν προσωρινές ή μόνιμες ρυθμίσεις. Με αυτό τον τρόπο καθορίζουμε την λειτουργία σε ορισμένες διεργασίες. Για να δούμε τις μεταβλητές και τι τιμή έχουν πληκτρολογούμε την εντολή printenv. Για να θέσουμε μια μεταβλητή τότε πληκτρολογούμε

```
export <όνομα_μεταβλητής>=<τιμή_μεταβλητής>
```

Κάποιες μεταβλητές είναι ήδη καθορισμένες όπως θα διαπιστώσετε όταν πληκτρολογήσετε την εντολή printenv. Στις μεταβλητές αυτές θα βρείτε την HOME όπου περιέχει την διαδρομή προς τον φάκελό σας. Την PATH, η οποία περιέχει ένα σύνολο από διαδρομές προς εκτελέσιμα αρχεία, χωρισμένες με άνω και κάτω τελεία. Όταν πληκτρολογούμε κάποια εντολή τότε το κέλυφος ψάχνει σε αυτές τις

διαδρομές για να βρει το εκτελέσιμο αρχείο της εντολής.

Η μεταβλητή PS1 περιέχει μια σειρά από χαρακτήρες η οποία εμφανίζεται στο prompt του κελύφους, ενώ η μεταβλητή _ περιέχει την διαδρομή προς την τελευταία εντολή που εκτελέσαμε. Σε περίπτωση που θέλουμε κάποια μεταβλητή περιβάλλοντος να είναι καθορισμένη κάθε φορά που ανοίγουμε ένα τερματικό ή κάθε φορά που εισερχόμαστε στο σύστημα, τότε θα περάσουμε της εντολές που θέτουν τις μεταβλητές αυτές στα αρχεία .bashrc και .profile

2.1.7 Pattern Matching

Σε όλα τα συστήματα Unix και Linux το κέλυφος σου δίνει την δυνατότητα να βρίσκεις αλλά και να εκτελείς εντολές σε ένα ή παραπάνω αρχεία ή φακέλους. Όπως θα δούμε και παρακάτω σε όλες τις εντολές υπάρχει τουλάχιστον ένα όρισμα που είναι η διαδρομή προς ένα αρχείο ή ένα φάκελο.

Στα πρότυπα αυτά υπάρχουν και κάποιοι ειδικοί χαρακτήρες που αντικαθιστούν ένα ή περισσότερα γράμματα. Οι χαρακτήρες αυτοί είναι το *, το οποίο μπορεί να αντικαταστήσει οποιοδήποτε χαρακτήρα και οποιοδήποτε πλήθος από χαρακτήρες. Επίσης χρησιμοποιείτε το αγγλικό ερωτηματικό (?), το οποίο μπορεί να αντικαταστήσει έναν οποιονδήποτε χαρακτήρα.

2.1.8 Διοχέτευση εντολών

Οι εντολές στο Linux είναι φτιαγμένες έτσι ώστε να είναι όσο πιο απλές γίνεται. Οι πιο σύνθετες εντολές δημιουργούνται συνδυάζοντας δύο ή παραπάνω εντολές. Ο συνδυασμός των εντολών γίνεται δίνοντας ως είσοδο σε μια εντολή την έξοδο μιας άλλης. Για να συνδυάσουμε δύο εντολές χρησιμοποιούμε το σύμβολο pipe (|) και η εντολή πλέον συντάσσεται

```
command1 | command2
```

2.1.9 Ανακατεύθυνση εντολών

Μία από τις καινοτομίες του Unix ήταν ότι είχε abstract ροές εισόδου/εξόδου, κάτι που έκανε την σύνδεση περιφερειακών πολύ εύκολη χωρίς να χρειάζεται να γραφτεί ειδικό πρόγραμμα για να τα συνδέσουμε. Οι βασικές ροές δεδομένων για την αλληλεπίδραση του χρήστη με τον υπολογιστή υπάρχουν και στο Linux και είναι η stdin(standar input) για είσοδο δεδομένων, η stdout(standar output) για την εμφάνιση των αποτελεσμάτων και η stderr(standar error) για την εμφάνιση τυχών λαθών κατά την εκτέλεση μιας εντολής.

Από προεπιλογή η ροή stdin παίρνει δεδομένα από το πληκτρολόγιο ενώ η stdout και η stderr εκτυπώνουν στην οθόνη. Τις ροές αυτές εισόδου/εξόδου μπορούμε να τις ανακατευθύνουμε και έτσι για παράδειγμα αντί η έξοδος ενός προγράμματος να εμφανιστεί στην οθόνη να αποθηκευθεί σε ένα αρχείο, το οποίο μπορούμε αργότερα να το επεξεργαστούμε

Η ανακατεύθυνση των ροής εξόδου γίνεται με το σύμβολο >, έτσι για να αποθηκεύσουμε την έξοδο μιας εντολής θα πληκτρολογήσουμε

```
εντολή > αρχείο.
```

Η ανακατεύθυνση της εισόδου γίνεται με το σύμβολο <, δηλαδή για να δώσουμε είσοδο σε μία εντολή από ένα αρχείο θα πληκτρολογήσουμε

εντολή < αρχείο.

Επίσης μπορούμε να ανακατευθύνουμε την είσοδο και τη έξοδο στην ίδια εντολή αν πληκτρολογήσουμε

εντολή < αρχείο_εισόδου > αρχείο_εξόδου.

Αν χρησιμοποιήσουμε τα σύμβολα > τότε το αρχείο που έχουμε δώσει ως όρισμα αν υπάρχει θα σβηστεί και θα δημιουργηθεί καινούριο, σε περίπτωση που θέλουμε να προσθέσουμε το αποτέλεσμα σε ένα αρχείο τότε θα πληκτρολογήσουμε το σύμβολο >>.

2.2 Βασικές εντολές του Bash

Η γενική μορφή των εντολών είναι `command [-option] [-argument]`. Το options είναι επιλογές που επηρεάζουν την έξοδο της εντολής ενώ τα arguments είναι η είσοδος της εντολής, συνήθως κάποιο path για ένα αρχείο ή φάκελο. Το πλήθος των options και των arguments εξαρτάται από την εντολή.

Μπορούμε να βρούμε βοήθεια για τις εντολές του bash χρησιμοποιώντας την εντολή `man` και την εντολή `info`. Στις 2 εντολές σαν argument χρησιμοποιούμε την εντολή για την οποία θέλουμε να βρούμε πληροφορίες, π.χ. `man cp` ή `info cp` και σαν αποτέλεσμα παίρνουμε μια συνοπτική περιγραφή της εντολής με όλα τα options που δέχεται η εντολή και το τι κάνει το καθένα.

Τις εντολές μπορούμε να τις χωρίσουμε στις εξής κατηγορίες :

1. περιήγηση στο σύστημα αρχείων και επεξεργασία φακέλων και αρχείων
2. έλεγχος λειτουργιών συστήματος

Ανάλογα με τα δικαιώματα και τις λειτουργίες της εντολής ορισμένες από αυτές μπορεί να τις εκτελέσει μόνο ο διαχειριστής του συστήματος.

2.2.1 Ιεραρχία συστήματος αρχείων

Στο Linux το σύστημα αρχείων έχει την μορφή δένδρου, όλα τα αρχεία και οι φάκελοι βρίσκονται κάτω από το root directory που συμβολίζεται με '/'. Η ιεραρχία, τα βασικά directories και τα περιεχόμενα τους καθορίζονται στο Filesystem Hierarchy Standard. Το standard δεν καθορίζει απόλυτα το σύστημα αρχείων καθώς επίσης δεν το ακολουθούν και όλες οι διανομές κατά γράμμα. Στις περισσότερες διανομές όμως υπάρχουν ορισμένα directories με συγκεκριμένο όνομα και περιεχόμενο. Ορισμένα από αυτά φαίνονται στον παρακάτω πίνακα

Directory	Σημείωση
/bin	Περιέχει εντολές που χρησιμοποιούν οι χρήστες του συστήματος
/boot	Περιέχει τον πυρήνα Linux και τα αρχεία ρύθμισης του bootloader
/dev	Περιέχει αρχεία που αντιπροσωπεύουν περιφερειακές συσκευές
/etc	Περιέχει αρχεία ρύθμισης των προγραμμάτων
/home	Περιέχει τον home φάκελο του κάθε χρήστη
/media	Περιέχει αρχεία που αντιπροσωπεύουν εξωτερικά μέσα αποθήκευσης, όπως το cdrom
/mnt	Πριν καθοριστεί το /media χρησιμοποιείται για την προσάρτηση συσκευών
/proc	Περιέχει πληροφορίες για τις processes που εκτελούνται στο σύστημα
/root	Το home directory του administrator(root user)
/sbin	Περιέχει εντολές που μπορεί να εκτελέσει μόνο ο administrator
/tmp	Προσωρινά αρχεία των εφαρμογών
/usr	Περιέχει προγράμματα, βιβλιοθήκες, τεκμηρίωση και άλλα αρχεία που δεν χρειάζονται κατά την εκκίνηση του συστήματος
/var	Περιέχει directories και δεδομένα που χρησιμοποιούν διάφορα προγράμματα, όπως ο ftp server και ο web server

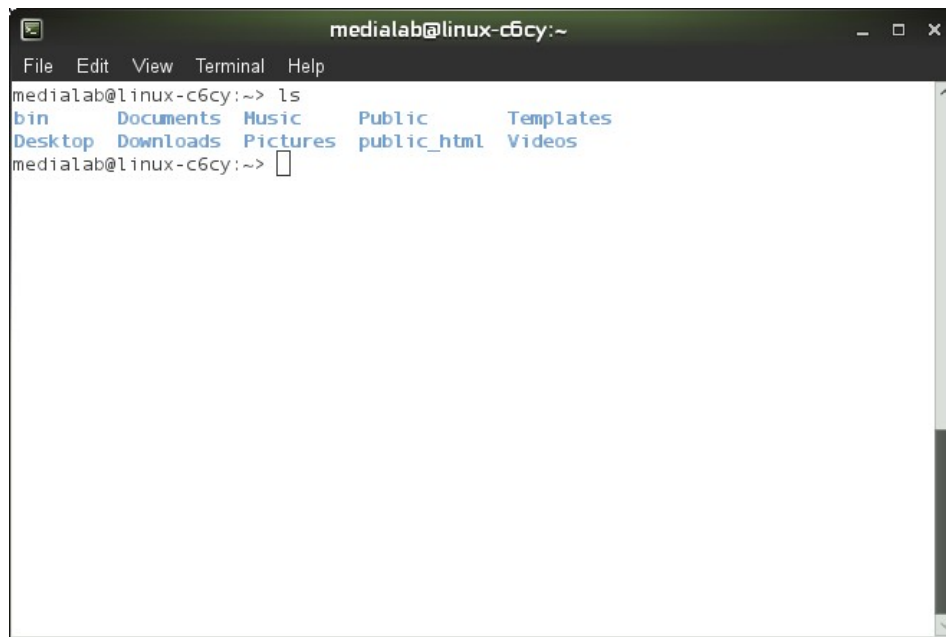
2.2.1 Εντολές περιήγησης στο σύστημα αρχείων και επεξεργασίας αρχείων και φακέλων

Εκτύπωση του τρέχοντος directory στην οθόνη

Όπως είδαμε στο prompt του shell εμφανίζεται το directory στο οποίο βρισκόμαστε, αυτό συμβαίνει γιατί έχει την κατάλληλη τιμή η μεταβλητή περιβάλλοντος PS1. Αν σε μια διανομή που χρησιμοποιούμε στο prompt δεν εμφανίζεται το τρέχον directory, μπορούμε να το δούμε με την εντολή pwd(print working directory).

Εκτύπωση περιεχομένων φακέλου στην οθόνη

Μπορούμε να δούμε τα περιεχόμενα ενός φακέλου με την εντολή ls(list segments ή list sort). Στην εικόνα 2.3 βλέπουμε την έξοδο της εντολής ls.



```
medialab@linux-c6cy:~  
File Edit View Terminal Help  
medialab@linux-c6cy:~> ls  
bin Documents Music Public Templates  
Desktop Downloads Pictures public_html Videos  
medialab@linux-c6cy:~> █
```

Εικόνα 2.3

Η εντολή ls χωρίς ορίσματα μας εμφανίζει τα περιεχόμενα του φακέλου στον οποίο βρισκόμαστε. Στην συγκεκριμένη περίπτωση μας εμφάνισε τα περιεχόμενα του φακέλου του χρήστη που είναι μόνο φάκελοι. Εδώ πρέπει να παρατηρήσουμε ότι τα ονόματα των φακέλων εμφανίστηκαν με μπλε χρώμα. Αυτός είναι ένας γρήγορος τρόπος να καταλάβουμε τον τύπο ενός αρχείου αφού στα Unix και στα Linux δεν είναι απαραίτητο ο τύπος του αρχείου να προσδιορίζεται από την κατάληξή του. Σε άλλες περιπτώσεις όπου ένας φάκελος δεν περιέχει μόνο φακέλους θα δούμε ότι τα απλά αρχεία εμφανίζονται χωρίς χρώμα, τα εκτελέσιμα με πράσινο και τα συμπιεσμένα με κόκκινο, φυσικά υπάρχουν και άλλοι τύποι αρχείων που αναγνωρίζει το bash και τα εμφανίζει με άλλο χρώμα.

Μερικά από τα options που μπορούμε να χρησιμοποιήσουμε με την εντολή ls είναι :

ls -a



```
medialab@linux-c6cy:~> ls -a
.          .dircache .gnome2   Music     .recently-used.xbel
..         Documents .gnome2_private .nautilus Templates
.bash_history Downloads .gstreamer-0.10 .ooo3     .thumbnails
.bashrc    .emacs    .gtk-bookmarks .Pictures Videos
.bin       .esd_auth .gvfs      .profile  .vimrc
.cache     .fontconfig .ICEauthority .Public   .xim.template
.config    .fonts     .inputrc   .public_html .xinitrc.template
.dbus      .gconf     .local     .pulse     .xsession-errors
.Desktop   .gconfd   .mozilla   .pulse-cookie
```

Εικόνα 2.4

Όπως βλέπουμε στην εικόνα 2.4 η έξοδος της εντολής έχει αλλάξει. Η διαφορά είναι ότι πλέον εμφανίζονται επιπλέον ο φάκελος “.”, ο φάκελος “..” και τα κρυφά αρχεία και φακέλοι. Οι φάκελοι “.” και “..” είναι 2 ειδικοί φάκελοι, στην πραγματικότητα δεν είναι φάκελοι αλλά δείκτες σε φακέλους, που υπάρχουν σε κάθε φάκελο του συστήματος. Ο “.” δείχνει τον φάκελο που είμαστε ενώ ο “..” δείχνει τον γονικό φάκελο. Τα ονόματα των κρυφών αρχείων και φακέλων ξεκινάνε με “.”, π.χ. .hidden.txt.

ls -l

Με το option -l η εντολή ls δίπλα στο όνομα του αρχείου η φακέλου εμφανίζει επιπλέον πληροφορίες όπως τα δικαιώματα, τον ιδιοκτήτη, την ομάδα την ημερομηνία κ.λ.π. Ίδια αποτελέσματα με την εντολή ls -l μπορούμε να πάρουμε και με την εντολή dir.

ls -R

Με το option -R η εντολή εκτελείται αναδρομικά σε όλους τους φακέλους που εκτυπώνει. Έτσι αν εκτελέσουμε την εντολή ls -R στο φάκελο root(/) θα μας εκτυπώσει όλα τα αρχεία και τους φακέλους του συστήματος.

Όπως αναφέραμε και πιο πάνω περισσότερες πληροφορίες μπορούμε να βρούμε για την εντολή και τα options με τις εντολές man και info. Επίσης το πλήθος των options είναι μεταβλητό, δηλαδή δεν είμαστε αναγκασμένοι να χρησιμοποιήσουμε μόνο ένα option αλλά μπορούμε να χρησιμοποιήσουμε οποιοδήποτε συνδυασμό τους.

Αντιγραφή αρχείου ή φακέλου

Η αντιγραφή των αρχείων ή των φακέλων γίνεται με την εντολή `cp(copy)`. Στην εντολή αυτή πάντα έχουμε 2 ορίσματα, την πηγή και τον προορισμό(`cp πηγή προορισμός`). Η πηγή είναι το `path` που μας οδηγεί σε ένα αρχείο, ενώ ο προορισμός είναι το `path` προς τον φάκελο στον οποίο θέλουμε να αντιγράψουμε το αρχείο. Έτσι αν έχουμε για παράδειγμα ένα αρχείο `text.txt` στο φάκελο `Documents` και θέλουμε να το αντιγράψουμε στο φάκελο `Backup` που έχουμε δημιουργήσει θα πληκτρολογήσαμε

```
cp /home/<user>/Documents/text.txt /home/<user>/Documents/Backup/
```

Στην παραπάνω εντολή έχουμε δώσει την διαδρομή προς την πηγή και τον προορισμό ως απόλυτα μονοπάτια από το φάκελο `root`. Αυτό δεν είναι απαραίτητο διότι μπορούμε να δώσουμε την διαδρομή ως σχετικά μονοπάτια από το φάκελο στον οποίο βρισκόμαστε, δηλαδή αν είμαστε στον φάκελο `Documents` θα είχαμε το ίδιο αποτέλεσμα αν πληκτρολογήσουμε `cp text.txt Backup/`. Επίσης μπορούμε να χρησιμοποιήσουμε τα “~”, “.” και “..” για να δώσουμε τη διαδρομή της πηγής και του προορισμού.

cp -i

Με το `option -i` η εντολή γίνεται `interactive`, δηλαδή πριν εκτελεστεί μας ζητάει επιβεβαίωση. Αυτό είναι πολύ χρήσιμο για να αποφύγουμε η εντολή να εκτελεστεί κατά λάθος σε κάποιο αρχείο.

cp -v

Το `option -v` κάνει την εντολή `verbose`, δηλαδή μετά την εκτέλεση της τυπώνει στην οθόνη πια αρχεία ή φακέλους αντέγραψε.

cp -r

Κανονικά με την εντολή `cp` δεν μπορούμε να αντιγράψουμε φακέλους. Με το `option -r` η εντολή εκτελείται αναδρομικά με αποτέλεσμα την αντιγραφή φακέλων.

Μεταφορά αρχείου ή φακέλου

Η μεταφορά των αρχείων και των φακέλων γίνεται με την εντολή `mv(move)`. Η χρήση της είναι αντίστοιχη με την εντολή `cp`. Πάντα έχουμε 2 διαδρομές σαν `arguments`, ένα για την πηγή και ένα για τον προορισμό. Αν υποθέσουμε τις συνθήκες του προηγούμενου παραδείγματος τότε για να μεταφέρουμε το αρχείο θα γράφαμε

```
mv ~/Documents/text.txt ~/Documents/Backup/
```

Κατά τη μεταφορά στη διαδρομή του προορισμού μπορούμε να δώσουμε και καινούριο όνομα στο αρχείο, δηλαδή `mv text.txt Backup/moved.txt`. Έτσι το αρχείο `text.txt` θα μετακινηθεί στο φάκελο `Backup` και πλέον θα έχει το όνομα `moved.txt`.

Η εντολή `mv` μπορεί να χρησιμοποιηθεί και για έναν ακόμα λόγο, την μετονομασία αρχείων και φακέλων, αυτό γίνεται δίνοντας ως διαδρομή πηγής και προορισμού ίδια αλλά με διαφορετικό όνομα αρχείου, δηλαδή `mv text.txt renamed.txt`. Τα `options -i` και `-v` ισχύουν και στην εντολή `mv` με αντίστοιχα αποτελέσματα.

Διαγραφή αρχείου ή φακέλου

Η διαγραφή ενός αρχείου ή ενός φακέλου γίνεται με την εντολή `rm` (remove). Η εντολή `rm` θέλει μεγάλη προσοχή όταν χρησιμοποιείτε γιατί τα αρχεία που διαγράφουμε δεν μεταφέρονται στον κάδο ανακύκλωσης και έτσι δεν έχουμε την ευκαιρία να επαναφέρουμε τα αρχεία μας.

Η εντολή `rm` παίρνει ένα όρισμα μόνο, τη διαδρομή προς το αρχείο ή το φάκελο που θέλουμε να σβήσουμε. Έτσι αν θέλαμε να σβήσουμε ένα αρχείο από το φάκελο Documents θα γράφαμε

```
rm ~/Documents/text.txt
```

Για να σβήσουμε ένα φάκελο πρέπει να δώσουμε και το option `-rf` ώστε η εντολή να εκτελεστεί αναδρομικά και να σβήσει τα περιεχόμενα του φακέλου και μετά τον ίδιο τον φάκελο.

Επίσης 2 option που μπορούμε να δώσουμε για να κάνουμε πιο ασφαλής τη χρήση της εντολής `rm` είναι τα `-i` και `-v` που όπως είδαμε και παραπάνω κάνουν την εντολή `interactive` και `verbose` αντίστοιχα.

Μετακίνηση στο σύστημα αρχείων

Με την εντολή `cd` (change directory) μπορούμε να μετακινηθούμε στο σύστημα αρχείων του λειτουργικού. Όπως αναφέραμε και πιο πάνω όταν ανοίγουμε ένα καινούριο τερματικό πάντα βρίσκεται στο φάκελο `home` του χρήστη που τον άνοιξε. Η εντολή `cd` δέχεται μόνο ένα όρισμα, τη διαδρομή προς το αρχείο που θέλουμε να πάμε, και κανένα option. Έτσι αν θέλουμε να πάμε στο φάκελο Documents θα πληκτρολογήσουμε

```
cd Documents/
```

Δημιουργία φακέλων

Η δημιουργία φακέλων γίνεται με την εντολή `mkdir` (make directory). Για να δημιουργήσουμε ένα φάκελο μέσα στον φάκελο που βρισκόμαστε, θα πληκτρολογήσουμε

```
mkdir newDirectory
```

Αν θέλουμε να δημιουργήσουμε πολλούς φακέλους τον ένα μέσα στον άλλο θα χρησιμοποιήσουμε το option `-p`, έτσι θα πληκτρολογήσουμε

```
mkdir -p newDirectory/newDirectory2/newDirectory3
```

Μετά την εκτέλεση της εντολής θα έχει δημιουργηθεί ο φάκελος με το όνομα `newDirectory3` μέσα στο φάκελό `newDirectory2` και αυτός με τη σειρά του θα είναι μέσα στο φάκελο `newDirectory`.

Διαγραφή φακέλων

Μπορούμε να διαγράψουμε άδειους φακέλους με την εντολή `rmdir` (remove directory). Σε περίπτωση που ένας φάκελος είναι άδειος τότε η εντολή θα αποτύχει. Όπως είπαμε και πιο πάνω για να διαγράψουμε φακέλους που δεν είναι άδειοι τότε καλούμε την εντολή `rm` με το option `-rf`.

Ένωση αρχείων

Με την εντολή `cat`(concatenate) μπορούμε να ενώσουμε τα περιεχόμενα δυο αρχείων σε ένα. Η εντολή γράφεται ως

```
cat file1 file2
```

Το αποτέλεσμα της εντολής εκτυπώνεται στην οθόνη. Για να αποθηκεύσουμε το αποτέλεσμα σε ένα αρχείο, όπως έχουμε αναφέρει, πρέπει να ανακατευθύνουμε την έξοδο

```
cat file1 file2 > outputfile
```

Εκτύπωση περιεχομένων αρχείων

Λόγο της εκτύπωσης του αποτελέσματος στην οθόνη, η εντολή `cat` μπορεί να χρησιμοποιηθεί και για να δούμε τα περιεχόμενα ενός αρχείου στην οθόνη δίνοντας σαν όρισμα μόνο ένα αρχείο. Έτσι αν πληκτρολογήσουμε `cat file1` θα εκτυπωθούν στην οθόνη τα περιεχόμενα του αρχείου `file1`.

Μπορούμε επίσης να δούμε ένα μέρος των περιεχομένων ενός αρχείου χρησιμοποιώντας τις εντολές `head` και `tail`. Χωρίς options οι δυο αυτές εντολές εκτυπώνουν τις 10 πρώτες και τελευταίες γραμμές ενός αρχείου αντίστοιχα.

`head -n N` ή `tail -n N`

Με το όρισμα `-n` μπορούμε να εκτυπώσουμε στην οθόνη παραπάνω ή λιγότερες από 10 γραμμές. Έτσι για να δούμε τις 15 πρώτες γραμμές ενός αρχείου θα πληκτρολογήσουμε

```
head -n 15 file
```

`head -c N` ή `tail -c N`

Αντίστοιχα με το option `-n` αλλά αυτή τη φορά καθορίζουμε τον αριθμό των bytes που θέλουμε να εμφανιστούν στην οθόνη.

`find` και `locate`

Η εντολή `find` μας επιτρέπει να εκτελέσουμε αναζήτηση στο σύστημα αρχείων για ένα συγκεκριμένο αρχείο ή φάκελο. Η σύνταξη της εντολής `find` είναι `find <path> <criteria> <action>`, όπου `path` είναι η διαδρομή προς κάποιο φάκελο από την οποία θα ξεκινήσει η αναζήτηση, `criteria` είναι ένα σύνολο από επιλογές, που δέχεται η εντολή σαν ορίσματα, όπου αλλάζουν την συμπεριφορά εκτέλεσης της εντολής και `action` είναι μια εντολή που θέλουμε να εκτελεστεί στα αποτελέσματα της αναζήτησης.

Μερικά κριτήρια αναζήτησης που μπορούμε να δώσουμε σαν επιλογές είναι :

- `name name`, για να αναζητήσουμε αρχεία με ένα συγκεκριμένο όνομα `name`,
- `type f`, όταν θέλουμε να αναζητήσουμε μόνο απλά αρχεία και όχι φακέλους,
- `user u`, για να βρούμε αρχεία που ανήκουν σε έναν συγκεκριμένο χρήστη `u` και
- `mmmin n`, για αρχεία που άλλαξαν πριν από `n` λεπτά.

Επίσης μπορούμε να δώσουμε εντολές με την μορφή επιλογών π.χ. -delete για να διαγράψουμε τα αρχεία που βρέθηκαν από την αναζήτηση. Η εντολή find έχει πολλά ακόμα κριτήρια αναζήτησης και εντολές που μπορούν να εκτελεστούν κάνοντας την μία από τις πιο δυνατές εντολές.

Παραδείγματα της εντολής find είναι:

1. find / -name bin, για να δούμε όλους τους φακέλους που περιέχουν binary αρχεία, συνήθως είναι σε φακέλους με το όνομα bin
2. find tmp -name *.txt -delete, για να διαγράψουμε όλα τα αρχεία με κατάληξη txt από τον ένα φάκελο tmp.

Εκτός από την εντολή find υπάρχει και μια δεύτερη εντολή που μπορούμε να χρησιμοποιήσουμε για να αναζητήσουμε ένα αρχείο στο σύστημά μας, η εντολή locate. Η εντολή locate αντί να ψάχνει το σύστημα αρχείων εκτελεί την αναζήτηση σε μία βάση δεδομένων η οποία ενημερώνεται ανά τακτά χρονικά διαστήματα. Επίσης μπορούμε να ενημερώσουμε την βάση με την εντολή updatedb.

Η σύνταξη της εντολής locate είναι locate [-options] <filename>. Μερικές επιλογές είναι η -c όπου εμφανίζει μόνο τον αριθμό των αρχείων και των φακέλων που βρέθηκαν, η -i, ώστε να μην γίνεται διάκριση μεταξύ πεζών και κεφαλαίων γραμμάτων και η -e όπου εκτυπώνει τις καταχωρήσεις για αρχεία που βρίσκονται στο σύστημα την στιγμή που εκτελείται η εντολή. Η εντολή locate είναι πιο γρήγορη από την εντολή find αλλά μπορεί τα αποτελέσματα που θα πάρουμε να μην είναι σωστά σε περίπτωση που δεν είναι ενημερωμένη η βάση. Επίσης δεν έχει την δυνατότητα για σύνθετες αναζητήσεις όπως η find. Έτσι μπορούμε να πούμε, αν και δεν ισχύει πάντα, ότι σε ένα οικιακό σύστημα η χρήση της εντολής find είναι πιο κατάλληλη αντίθετα σε ένα σύστημα εταιρικό, όπου το σύστημα αρχείων είναι πολύ μεγάλο, ο χρόνος που χρειάζεται η find για να εκτελέσει την αναζήτηση κάνει την locate πιο κατάλληλη για χρήση.

2.2.2 Εντολές Ελέγχου Λειτουργιών Συστήματος

Οι εντολές που είδαμε μέχρι τώρα μας βοηθούν να δούμε πληροφορίες και περιεχόμενα για αρχεία και φακέλους που έχουμε στο σύστημα αρχείων μας. Σαν χρήστες όμως και ειδικά διαχειριστές θα χρειαστεί να χρησιμοποιήσουμε και εντολές για να ελέγξουμε πόρους του συστήματος.

top - Display Linux Tasks

Με την εντολή top μπορούμε να δούμε τις διεργασίες που τρέχουν στο σύστημα μας. Με την εκτέλεση της εντολής εμφανίζεται ένας πίνακας με ορισμένες από τις διεργασίες που τρέχουν δίνοντας και ορισμένες πληροφορίες για αυτές. όπως φαίνεται και στην παρακάτω εικόνα.

Εικόνα `top`

Οι πληροφορίες που μπορούμε να δούμε είναι το pid(process id, ένας μοναδικός αριθμός που αποδίδεται σε κάθε διεργασία όταν ξεκινάει η εκτέλεση της), user(ο χρήστης στο οποίο ανήκει/ξεκίνησε την διεργασία, %CPU(το ποσοστό χρόνου εκτέλεσης που τρέχει στον επεξεργαστή), %MEM(το ποσοστό μνήμης που έχει πάρει από το σύστημα), command(η εντολή που ξεκίνησε την διεργασία).

Εκτός από το να δούμε ποιες διεργασίες τρέχουν στο σύστημα μας και να ελέγξουμε κάποιες πληροφορίες για αυτές μπορούμε να τερματίσουμε και όσες διεργασίες θέλουμε και μπορούμε. Για να τερματίσουμε μια διεργασία, όσο εκτελείται η εντολή top, πατάμε το πλήκτρο k(kill), μας ζητείται να πληκτρολογήσουμε το pid της διεργασίας, το οποίο μπορούμε να δούμε από τον πίνακα, και στην συνέχεια πρέπει να επιβεβαιώσουμε ότι θέλουμε να τερματίσουμε την διεργασία.

Για να τερματίσουμε την εκτέλεση της εντολής top πατάμε το πλήκτρο q(quit). Μπορούμε να δούμε περισσότερες πληροφορίες για την χρήση της εντολής top πατώντας το πλήκτρο h ή ?.

ps - report a snapshot of the current processes.

Ένας άλλος τρόπος να δούμε πληροφορίες για τις διεργασίες που τρέχουν στο σύστημα μας είναι η εντολή ps. Σε αντίθεση με την εντολή top η εντολή ps δεν ανανεώνεται σε τακτά χρονιά διαστήματα, απλά εκτυπώνει στην οθόνη πληροφορίες για τις διεργασίες που τρέχουν την συγκεκριμένη στιγμή.

Στην περίπτωση που εκτελέσουμε την εντολή ps χωρίς ορίσματα θα μας εμφανίσει πληροφορίες μόνο για τις διεργασίες που έχουν ξεκινήσει από τον χρήστη που είμαστε συνδεδεμένοι και μόνο για τις διεργασίες που έχουν ξεκινήσει να εκτελούνται από το bash που τρέχουμε την συγκεκριμένη στιγμή. Όπως θα δούμε παρακάτω μπορούμε εκτελούμε διεργασίες στο background, προς το παρών εάν εκτελέσουμε την εντολή ps θα μας εμφανίσει πληροφορίες για την διεργασία ps και της διεργασίας bash.

ps -e

Δίνοντας το όρισμα e στην εντολή ps εκτυπώνονται πληροφορίες για όλες τις διεργασίες του συστήματος.

ps -ejH

Με τα ορίσματα -ejH εκτυπώνεται στην οθόνη ένα δένδρο από τις διεργασίες που τρέχουν στο σύστημα. Με την έννοια δένδρο εννοούμε ότι εκτός από τις πληροφορίες που μπορούμε να δούμε με την εκτέλεση της εντολής ps -e μπορούμε να δούμε και την σχέση που έχουν μεταξύ τους, δηλαδή ποια διεργασία ξεκίνησε κάποια άλλη. Την σχέση αυτή θα την αναλύσουμε σε επόμενο κεφάλαιο.

Μπορούμε να χρησιμοποιήσουμε και άλλα ορίσματα για να δούμε διαφορετικές πληροφορίες για τις διεργασίες που τρέχουν στο σύστημα μας. Για να δείτε περισσότερες πληροφορίες για την εντολή ps μπορείτε να δείτε το documentation εκτελώντας την εντολή man ps.

kill

Με την εντολή kill μπορούμε να τερματίσουμε την εκτέλεση μιας διεργασίας. Η εντολή kill είναι ενσωματωμένη εντολή του bash. Υπάρχει και άλλη εντολή kill η οποία είναι εντολή συστήματος και χρησιμοποιείται για να στείλουμε ένα signal σε μια διεργασία. Για να χρησιμοποιήσουμε την εντολή kill αρκεί να ξέρουμε το pid της διεργασίας που θέλουμε να τερματίσουμε. Για παράδειγμα εάν έχουμε εκτελέσει την εντολή ps και είδαμε ότι η διεργασία που θέλουμε να τερματίσουμε έχει pid 2025 τότε θα εκτελέσουμε την εντολή

kill 2025

su - change user ID or become superuser

Η εντολή su χρησιμοποιείται για να αλλάξουμε τον χρήστη με τον οποίο είμαστε συνδεδεμένοι. Μπορούμε να δώσουμε μόνο ένα όρισμα και αυτό είναι το username του χρήστη που θέλουμε να συνδεθούμε.

Η εντολή su είναι ένας από τους 2 τρόπους για να αποκτήσουμε δικαιώματα διαχειριστή, εάν δεν έχουμε κάνει login ως root. Για να γίνει αυτό εκτελούμε την εντολή

```
su root  
ή απλώς  
su
```

Για να επιστρέψουμε στον κανονικό μας χρήστη εκτελούμε την εντολή exit.

sudo - execute a command as another user

Μια ακόμη εντολή που χρησιμοποιείται για να αποκτήσουμε δικαιώματα διαχειριστή είναι η εντολή sudo. Σε αντίθεση με την εντολή su όπου χρησιμοποιείται για να αλλάξουμε τον χρήστη με τον οποίο έχουμε συνδεθεί, με την εντολή sudo απλώς εκτελούμε μόνο μια εντολή με δικαιώματα διαχειριστή. Η παράμετρος που δέχεται η εντολή sudo είναι η εντολή που θέλουμε να εκτελέσουμε ως διαχειριστές με τα όρισματα και τις παραμέτρους της.

Δεν μπορούν όλοι οι χρήστες του συστήματος να χρησιμοποιήσουν την εντολή sudo. Οι χρήστες που μπορούν να την χρησιμοποιήσουν ονομάζονται sudoers και καθορίζονται στο αρχείο /etc/sudoers. Συνήθως αυτό γίνεται για μια ομάδα από χρήστες, για παράδειγμα στην διανομή ubuntu στο αρχείο sudoers υπάρχουν ο χρήστης root και οι ομάδες admin και sudo. Στην συγκεκριμένη διανομή ο root χρήστης δεν ενεργοποιείται κατά την εγκατάσταση όπως σε άλλες διανομές και ο πρώτος χρήστης που δημιουργούμε ανήκει στην ομάδα admin, για αυτό και μπορεί να εκτελέσει την εντολή sudo για να διαχειριστεί το σύστημα.

2.2.3 Προγράμματα που χρησιμοποιούνται συχνά με εντολές

more & less

Οι εντολές more και less ονομάζονται pagers. Αυτό συμβαίνει διότι οργανώνουν το περιεχόμενο που δέχονται ως είσοδο σε σελίδες. Οι εντολές είναι χρήσιμες διότι όπως θα παρατηρήσετε τα περιεχόμενα ενός αρχείου ή η έξοδος μια εντολής πολλές φορές επεκτείνονται σε περισσότερες γραμμές από ότι μπορεί να εμφανιστεί στην οθόνη ενός τερματικού.

Και με τις δυο εντολές το περιεχόμενο ενός αρχείου εμφανίζεται στην οθόνη ανά σελίδα, χρησιμοποιώντας το space ή το enter βλέπουμε την επόμενη γραμμή ή σελίδα αντίστοιχα. Η εντολή less, κατά ένα περίεργο τρόπο, μας προσφέρει περισσότερες επιλογές, όπως το να χρησιμοποιούμε κουμπιά για να πηγαίνουμε προς τα πίσω ή να χρησιμοποιούμε τη ροδέλα του ποντικιού για να βλέπουμε διαφορετικά σημεία του αρχείου.

Για να σελιδοποιήσουμε την έξοδο μια εντολής και να δούμε τα αποτελέσματα στο αντίστοιχο pager αρκεί να χρησιμοποιήσουμε την διοχέτευση εντολών. Για παράδειγμα μια εντολή που είδαμε η

οποία τις περισσότερες φορές παράγει αποτελέσματα που ξεπερνούν το εύρος που μπορεί να δείξει ένα τερματικό είναι η ps. Για παράδειγμα, έστω ότι θέλουμε να βρούμε το pid της διεργασίας όπου εκτελείτε ο firefox ένας τρόπος είναι να σελιδοποιήσουμε τα αποτελέσματα της ps ώστε να είμαστε σίγουροι ότι θα μπορούσαμε να έχουμε πρόσβαση στις πληροφορίες για όλες τις διεργασίες ανεξάρτητα από το πόσες είναι αυτές. Για να σελιδοποιήσουμε τα αποτελέσματα της εντολής ps εκτελούμε την εντολή

```
ps -e | less
```

Στην συνέχεια μπορούμε να αναζητήσουμε την διεργασία που τρέχει ο firefox και να δούμε το pid που του έχει ανατεθεί.

Αναζήτηση σε Αρχείο με την Εντολή grep

Η αναζήτηση στα περιεχόμενα ενός αρχείου γίνεται με την εντολή grep. Η εντολή grep δέχεται δυο ορίσματα, το πρώτο όρισμα είναι η λέξη για την οποία ψάχνουμε και το δεύτερο όρισμα είναι η διαδρομή προς το αρχείο που θέλουμε να κάνουμε την αναζήτηση.

Μερικές από τις επιλογές που δέχεται η εντολή είναι η -i, όπου κατά την αναζήτηση δεν κάνει διάκριση μεταξύ πεζών και κεφαλαίων. Η επιλογή -v όπου εμφανίζει τις γραμμές στις οποίες δεν βρίσκει την λέξη που ψάχνουμε και η επιλογή -w όπου θα πρέπει να κάνει ταυτοποίηση της λέξης που ψάχνουμε με μια ολόκληρη λέξη από το αρχείο και όχι με μέρος μιας λέξης.

Σε περίπτωση που δεν καθορίσουμε κάποιο αρχείο τότε η εντολή grep εκτελεί την αναζήτηση στην προκαθορισμένη είσοδο. Μπορούμε να εκμεταλλευτούμε πάλι την διοχέτευση εντολών ώστε να φιλτράρουμε την έξοδο μιας εντολής.

Συνεχίζοντας το παράδειγμα με την εντολή ps, για να βρούμε το pid της διεργασίας που εκτελείτε ο firefox εκτελούμε την εντολή

```
ps -e | grep firefox
```

Και θα εμφανιστεί μόνο η γραμμή με τις πληροφορίες για την συγκεκριμένη διεργασία όπου περιέχει και το pid της διεργασίας.

Ταξινόμηση περιεχομένων αρχείων

Για να ταξινομήσουμε τα περιεχόμενα ενός αρχείου χρησιμοποιούμε την εντολή sort. Η εντολή sort είναι προτιμότερο να χρησιμοποιείται σε αρχεία όπου έχουν μορφή πίνακα, δηλαδή κάθε γραμμή είναι μια εγγραφή(record) και τα πεδία της καταχώρησης χωρίζονται μεταξύ τους από ένα οριοθέτη(delimiter). Επίσης σε κάθε εγγραφή τα δεδομένα μια στήλης πρέπει να είναι του ίδου τύπου. Τέτοια αρχεία είναι συνήθως τα αρχεία καταγραφής συμβάντων που παράγει το σύστημα μας, είτε ο πυρήνας είτε οι εφαρμογές που τρέχουμε, όπως και τα csv(comma separated values) που μπορούμε να εξάγουμε από μια βάση δεδομένων.

Η γενική μορφή σύνταξης της εντολής sort είναι: `sort [options] [file]`

Ορισμένα options που χρησιμοποιούνται με την εντολή sort είναι τα παρακάτω :

- **f** : Μετατρέπει όλες τις εγγραφές σε κεφαλαία γράμματα πριν την ταξινόμηση.
- **r** : Η σειρά εμφάνισης των αποτελεσμάτων είναι από το μεγαλύτερο στο μικρότερο.
- **n** : Ταξινόμηση με βάση την αριθμική τιμή.
- **tx** : Ο οριοθέτης είναι ο χαρακτήρας x, π.χ. αν το αρχείο είναι csv τότε χρησιμοποιούμε -t,

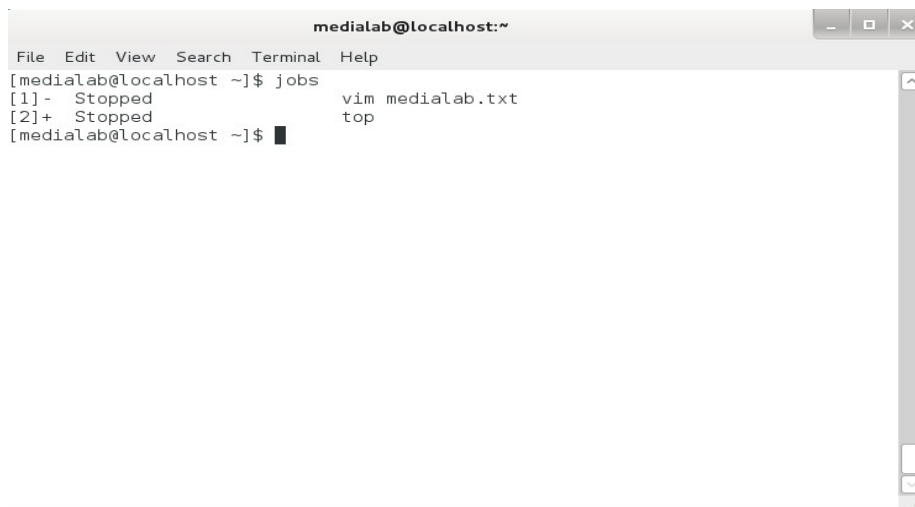
Όπως και με τις προηγούμενες εντολές, έτσι και η sort μπορεί να χρησιμοποιηθεί με την διοχέτευση εντολών για να ταξινομήσει αποτελέσματα εντολών όπου έχουν την απαραίτητη μορφοποίηση. Έχουμε ήδη δει 2 τέτοιες εντολές και αυτές είναι η ls -l και η ps

Εκτέλεση πολλαπλών διεργασιών

Ως τώρα έχουμε δει πως μπορούμε να εκτελέσουμε μερικές εντολές στο bash, κάθε εντολή που έχουμε εκτελέσει θα πρέπει να τελειώσει την εργασία της ώστε να μπορέσουμε να εκτελέσουμε την επόμενη. Αν και για τις περισσότερες εντολές δεν μας απασχολεί αφού τερματίζουν σε σύντομο χρονικό διάστημα δεν ισχύει το ίδιο για προγράμματα τα οποία ξεκινάμε από το τερματικό όπως ένας text editor. Σε αυτή την περίπτωση ίσως θέλουμε να τρέξουμε κάποια εντολή χωρίς να τερματίσουμε την λειτουργία του προγράμματος.

Αυτό μπορούμε να το κάνουμε διακόπτοντας την λειτουργία και στέλνοντας την διεργασία στο background με τον συνδυασμό Ctrl-Z. Επίσης μπορούμε να ξεκινήσουμε μια διεργασία η οποία εκτελείται στο background προσθέτοντας στο τέλος της εντολής το σύμβολο &.

Για να δούμε ποιες διεργασίες εκτελούνται στο background χρησιμοποιούμε την εντολή jobs, το αποτέλεσμα της οποίας εμφανίζεται στην παρακάτω εικόνα. Όπως βλέπουμε μερικές από τις διεργασίες εκτελούνται ακόμα(Running) ενώ άλλες είναι σταματημένες(Stopped). Ένα σημείο που πρέπει να προσέξουμε είναι ότι όλες οι διεργασίες έχουν την ίδια standar έξοδο, δηλαδή την οθόνη μας. Για αυτό το λόγο καλό είναι όταν εκτελούμε μια διεργασία στο background να ανακατευθύνουμε την έξοδο της σε κάποιο αρχείο ή στο /dev/null αν δεν θέλουμε να αποθηκευτεί κάπου.



```
medialab@localhost:~
File Edit View Search Terminal Help
[medialab@localhost ~]$ jobs
[1] - Stopped          vim medialab.txt
[2]+ Stopped          top
[medialab@localhost ~]$
```

Για να διαχειριστούμε τις διεργασίες αυτές χρησιμοποιούμε 2 εντολές, την `bg(background)` και την `fg(foreground)`. Για να επαναφέρουμε μια διεργασία από το `background` χρησιμοποιούμε την εντολή `fg`, ενώ αν θέλουμε να συνεχίσουμε την εκτέλεση μιας σταματημένης διεργασίας στο `background` χρησιμοποιούμε την εντολή `bg`. Και οι δυο εντολές δέχονται ένα όρισμα και η σύνταξή τους είναι : `bg/fg [job_spec]`

Το όρισμα `job_spec` μπορεί να πάρει τις παρακάτω τιμές :

1. `%` : Αναφορά στην τελευταία διεργασία που στάλθηκε στο `background`.
2. `%com` : Αναφέρεται στην διεργασία όπου η εντολή που την ξεκίνησε αρχίζει με την λέξη `com`.
3. `[%]com` : Σχεδόν ίδια με την προηγούμενη περίπτωση, με την διαφορά ότι η λέξη `com` δεν χρειάζεται να είναι στην αρχή της εντολής.
4. `%--` : Αναφέρεται στην διεργασία που στάλθηκε στο `background` προτελευταία.

2.3 Χρήστες, ομάδες και δικαιώματα

Το Linux είναι ένα πολυχρηστικό σύστημα, ένα χαρακτηριστικό που έχει κληρονομήσει από το Unix, αυτό σημαίνει ότι πολλοί χρήστες μπορούν να χρησιμοποιούν το σύστημα ταυτόχρονα (αν και αυτό δεν γίνεται στους προσωπικούς υπολογιστές). Έτσι κάθε χρήστης έχει το δικό του λογαριασμό στο σύστημα ώστε να μπορεί να αναγνωριστεί από αυτό. Επίσης για λόγους ασφαλείας, για κάθε χρήστη ορίζεται αυστηρά το τι μπορεί να κάνει στο σύστημα, αυτό επιτυγχάνεται μέσω των δικαιωμάτων.

2.3.1 Χρήστες

Όλο οι χρήστες του συστήματος υπάρχουν στο αρχείο `/etc/passwd`. Οι εκχωρήσεις του αρχείου έχουν την μορφή

```
medialab:x:1001:100:medialab:/home/medialab:/bin/bash
```

Τα πεδία της καταχώρησης χωρίζονται από τον χαρακτήρα `:` και η σημασία κάθε πεδίου είναι :

1. `username`
2. `password`
3. `user id`
4. `group id`
5. `user info`
6. `user home directory`
7. `command/shell`

Όπως φαίνεται στην καταχώρηση, το δεύτερο πεδίο θα έπρεπε να περιέχει το `password` του χρήστη, όμως απλά έχει ένα `x`. Αυτό σημαίνει ότι το `password` του χρήστη είναι αποθηκευμένο, σε κωδικοποιημένη μορφή για ασφάλεια, στο αρχείο `/etc/shadow`, στο οποίο πρόσβαση έχει μόνο ο διαχειριστής του συστήματος και οι χρήστες που ανήκουν στην ομάδα `shadow`.

2.3.2 Ομάδες

Όλοι οι χρήστες ανήκουν σε μία ή περισσότερες ομάδες, ανάλογα σε ποιες ομάδες ανήκει ένας χρήστης θα έχει πλήρη ή μερική πρόσβαση σε ορισμένα αρχεία, όπως και σε κάποιες λειτουργίες του συστήματος, όπως η αναπαραγωγή πολυμέσων. Αυτό γίνεται διότι οι διεργασίες(daemon) που ελέγχουν τις λειτουργίες αυτές τρέχουν ως χρήστες του συστήματος, έχοντας και τις αντίστοιχες ομάδες. Για να δούμε σε ποιές ομάδες ανήκουμε πληκτρολογούμε την εντολή groups.

A screenshot of a terminal window titled "medialab@linux-c6cy:~". The terminal shows the command "groups" being executed, resulting in the output "users video". The prompt "medialab@linux-c6cy:~>" is visible at the end of the line.

```
medialab@linux-c6cy:~> groups
users video
medialab@linux-c6cy:~>
```

Εικόνα 2.5

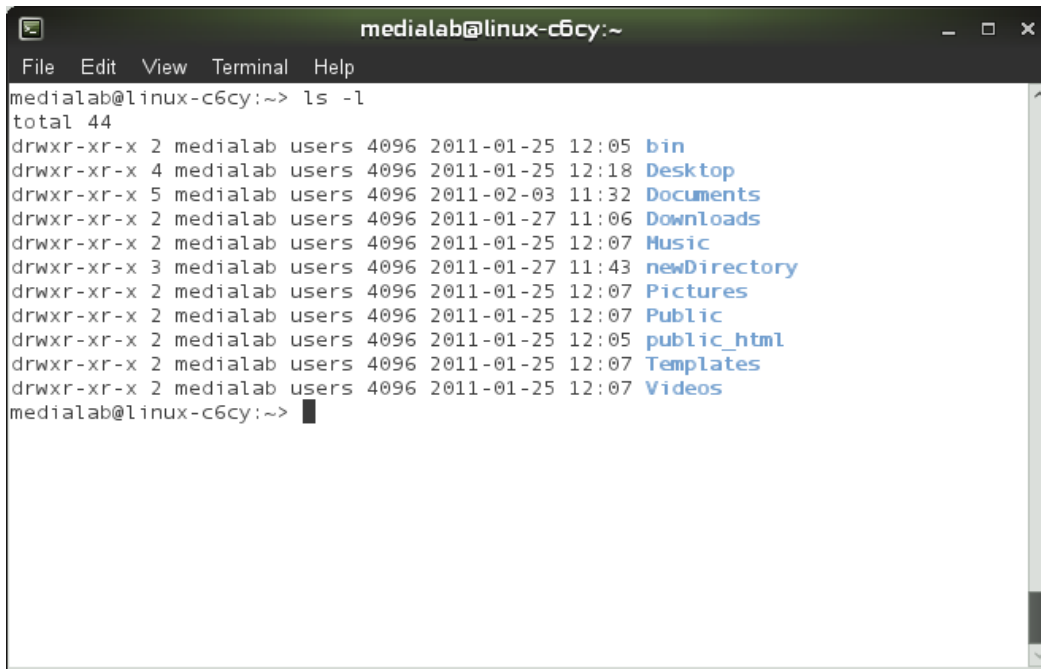
Όπως φαίνεται και στην εικόνα ο λογαριασμός που έχουμε δημιουργήσει κατά την εγκατάσταση ανήκει σε δυο ομάδες, στην users και στην video.

Δικαιώματα

Στα συστήματα Linux μπορούμε να ορίσουμε σε κάθε αρχείο και φάκελο του συστήματος τι δικαιώματα έχει ο κάθε χρήστης σε αυτό. Τα δικαιώματα που μπορεί να έχει ένας χρήστης είναι διάβασμα(read), εγγραφή(write) και εκτέλεση(execute).

Κάθε αρχείο ή φάκελος στο σύστημα αρχείων ανήκει σε έναν χρήστη(owner), συνήθως αυτός που το δημιούργησε. Με αυτό το χαρακτηριστικό το Linux δίνει δικαιώματα στον ιδιοκτήτη, στην ομάδα του ιδιοκτήτη(group) και σε όλους τους άλλους χρήστες(others).

Όπως είπαμε και πιο πάνω με την εντολή ls -l εμφανίζονται τα περιεχόμενα ενός φακέλου μαζί με πληροφορίες για αυτά.



```
medialab@linux-c6cy:~  
File Edit View Terminal Help  
medialab@linux-c6cy:~> ls -l  
total 44  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:05 bin  
drwxr-xr-x 4 medialab users 4096 2011-01-25 12:18 Desktop  
drwxr-xr-x 5 medialab users 4096 2011-02-03 11:32 Documents  
drwxr-xr-x 2 medialab users 4096 2011-01-27 11:06 Downloads  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:07 Music  
drwxr-xr-x 3 medialab users 4096 2011-01-27 11:43 newDirectory  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:07 Pictures  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:07 Public  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:05 public_html  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:07 Templates  
drwxr-xr-x 2 medialab users 4096 2011-01-25 12:07 Videos  
medialab@linux-c6cy:~>
```

Εικόνα 2.6

Ας πάρουμε ως παράδειγμα τον φάκελο `bin`, τα δικαιώματα είναι `rwxr-xr-x`, το `d` στην αρχή της γραμμής δείχνει ότι είναι φάκελος(directory). Παρατηρούμε ότι είναι τρεις τριάδες. Η πρώτη τριάδα αντιστοιχεί στον ιδιοκτήτη, η δεύτερη στην ομάδα του χρήστη και η τρίτη στους υπόλοιπους χρήστες. Το `r`, `w`, `x` είναι διάβασμα(read), εγγραφή(write) και εκτέλεση(execute) αντίστοιχα.

Σύμφωνα με τα παραπάνω ο ιδιοκτήτης του φακέλου έχει δικαιώματα ανάγνωσης, εγγραφής και εκτέλεσης. Η ομάδα του ιδιοκτήτη έχει δικαιώματα ανάγνωσης και εκτέλεσης και οι υπόλοιποι χρήστες έχουν δικαιώματα ανάγνωσης και εκτέλεσης, ως εκτέλεση σε ένα φάκελο ορίζεται το άνοιγμά του για την εμφάνιση των περιεχομένων του.

2.3.4 Εντολές διαχείρισης χρηστών και ομάδων

Κάθε χρήστης που χρησιμοποιεί έχει το δικό του λογαριασμό στο σύστημα καθώς και τον δικό του χώρο, το `home directory`, για την αποθήκευση αρχείων. Όταν εγκαταστήσουμε μια διανομή μας δίνεται η δυνατότητα να δημιουργήσουμε ένα χρήστη, ο οποίος συνήθως είναι και διαχειριστής του συστήματος. Ο διαχειριστής στην συνέχεια είναι υπεύθυνος να προσθέσει, να αφαιρέσει και γενικά να διαχειριστή τους χρήστες και τις ομάδες.

Κάθε διανομή προσφέρει τα δικά της εργαλεία για την διαχείριση των χρηστών και των ομάδων, προσφέροντας την ευκολία του γραφικού περιβάλλοντος ή όχι. Εμείς θα περιγράψουμε τις εντολές `useradd`, `userdel` και τις αντίστοιχες εντολές για τις ομάδες `groupadd` και `groupdel`.

useradd

Για να δημιουργήσουμε ένα νέο χρήστη στο σύστημα χρησιμοποιούμε την εντολή `useradd`. Η σύνταξη της εντολής `useradd` είναι : `useradd [options] username`. Η εντολή δέχεται αρκετά options με τα οποία μπορούμε ρυθμίσουμε παραμέτρους και χαρακτηριστικά για τον καινούριο χρήστη. Αν παραλείψουμε τα options τότε θα χρησιμοποιηθούν οι default ρυθμίσεις. Έτσι εάν θέλουμε να

δημιουργήσουμε έναν χρήστη με username panagiotis αρκεί να εκτελέσουμε την εντολή *useradd panagiotis*. Για να δούμε τις default ρυθμίσεις για έναν χρήστη εκτελούμε την εντολή *useradd -D*.

Αφού δημιουργηθεί ο καινούριος χρήστης για να μπορεί να εισέλθει στο σύστημα θα πρέπει να ορισθεί και ένας κωδικός. Αυτό γίνεται με την εντολή *password*, η σύνταξη της οποίας είναι *passwd username*. Μετά την εκτέλεση της εντολής θα μας ζητηθεί να πληκτρολογήσουμε τον κωδικό για τον συγκεκριμένο χρήστη.

userdel

Για να διαγράψουμε κάποιον χρήστη χρησιμοποιούμε την εντολή *userdel*. Η σύνταξη της *userdel* είναι *userdel [options] username*. Τα *options* που μπορούμε να ορίσουμε είναι δύο :

- a. *-f* : Η διαγραφή γίνεται ακόμα και εάν ο χρήστης είναι συνδεδεμένος στο σύστημα.
- b. *-r* : Εκτός από την διαγραφή του χρήστη διαγράφεται και το *home directory* του με τα περιεχόμενα του.

groupadd & groupdel

Αντίστοιχα με τις εντολές *useradd* και *userdel* για την δημιουργία και διαγραφή ενός χρήστη, χρησιμοποιούμε τις εντολές *groupadd* και *groupdel* για την δημιουργία και διαγραφή, αντίστοιχα μιας ομάδας. Η σύνταξη της εντολής *groupadd* είναι : *groupadd [options] groupname* και η σύνταξη της εντολής *groupdel* είναι : *groupdel groupname*.

2.3.5 Αλλαγή των δικαιωμάτων με την εντολή *chmod*

Μπορούμε να αλλάξουμε τα δικαιώματα σε ένα αρχείο ή φάκελο με την εντολή *chmod*. Η εντολή *chmod* συντάσσεται ως

chmod [χρήστες] [τελεστής] [δικαιώματα] [αρχεία ή φάκελοι]

Το όρισμα *χρήστες* δέχεται τρεις τιμές, *u* για τον ιδιοκτήτη, *g* για την ομάδα του ιδιοκτήτη, *o* για τους υπόλοιπους χρήστες ή οποιονδήποτε συνδυασμό τους, αν θέλουμε να αναφερθούμε και στους τρεις τότε μπορούμε να βάλουμε και *a* all. Το όρισμα *τελεστής* δέχεται τρεις τιμές + - και =. Όταν βάζουμε + τότε προστίθενται τα δικαιώματα που καθορίζουμε στα ήδη υπάρχον, με το - αφαιρούνται από τα ήδη υπάρχον ενώ με το = ... μόνο τα δικαιώματα που καθορίζουμε. Το όρισμα *δικαιώματα* δέχεται και αυτό τρεις τιμές *r,w,x* με το νόημα που έχουμε αναφέρει πιο πάνω. Μπορούμε να δώσουμε οποιοδήποτε συνδυασμό τους και όχι απαραίτητα μόνο μια τιμή. Μπορούμε επίσης να καθορίσουμε τα δικαιώματα χρησιμοποιώντας τρία ψηφία του οκταδικού συστήματος. Κάθε ψηφίο σε αυτό το σύστημα παίρνει τιμές από το 0 έως το 7. Το νόημα κάθε ψηφίου φαίνεται στον παρακάτω πίνακα.

Δικαιώματα	Δυαδικό	Οκταδικό	Περιγραφή
---	000	0	Κανένα Δικαίωμα
--x	001	1	Μόνο εκτέλεση
-w-	010	2	Μόνο εγγραφή
-wx	011	3	Εγγραφή και εκτέλεση
r--	100	4	Μόνο ανάγνωση
r-x	101	5	Ανάγνωση και εκτέλεση
rw-	110	6	Ανάγνωση και εγγραφή
rwX	111	7	Ανάγνωση, εγγραφή και εκτέλεση

Εδώ πρέπει να παρατηρήσουμε την αντιστοιχία των δικαιωμάτων στην μορφή `gwx` και του δυαδικού. Αν θέλουμε να υπάρχει το δικαίωμα τότε στη αντίστοιχη θέση έχουμε 1 ενώ αν δεν θέλουμε να υπάρχει έχουμε 0. Σύμφωνα με τον παραπάνω πίνακα αν θέλουμε σε ένα αρχείο να δώσουμε δικαιώματα ανάγνωσης, εγγραφής και εκτέλεσης σε όλους τους χρήστες τότε θα εκτελέσουμε την εντολή

```
chmod 777 <file>
```

Το πρώτο ψηφίο αναφέρεται στον ιδιοκτήτη, το δεύτερο στην ομάδα του και το τρίτο στους υπόλοιπους χρήστες.