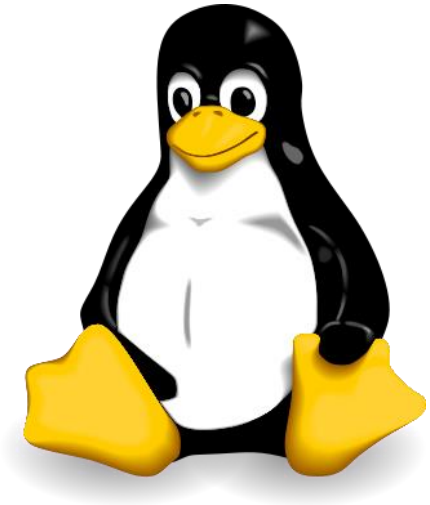


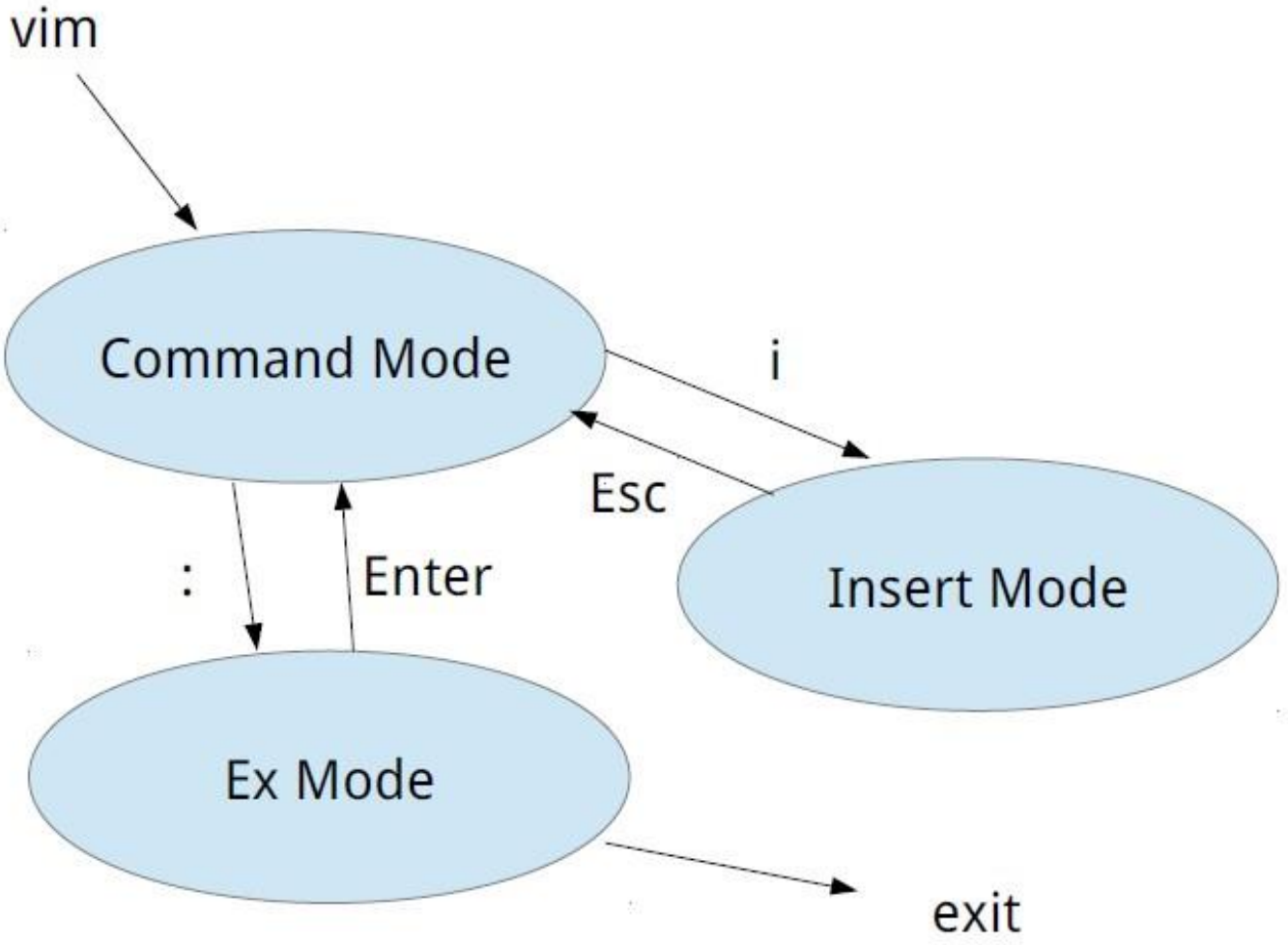
Linux Essentials
(Linux Level 1)

Լինուքսի Հիմունքներ
(Լինուքս փուլ 1)

Մաս 2

Խմբագիրներ

- **vi /vim** Standard UNIX editor



Խմբագիրներ

- **vi /vim** Standard UNIX editor
- **nano** Simple display-oriented text editor
- **mcedit** Midnight Commander internal editor
- **joe** Joe editor
- **gedit/kate** Graphical editors

Disk/file space usage (df, du)

du stands for "disk usage."
Reports the amount of disk space used by the directory.

Useful switches:
du -h Human readable; uses friendly k, M, and G indicators to show file size
du -s Displays only a total for each argument (same as -d 0)
du --max-depth=1 Limit max depth to N (if N=1 will print totals of subdirectories only)

du -hs /etc 2>/dev/null
Show total folder size for **/etc** , send error messages to **/dev/null**

du -h --max-depth=1 /usr/ 2>/dev/null | sort –hr
du -hd1 /var
du -sh /var/* | sort -rh
Show size of each folder under **/usr**, sort by human readable size,
send error messages to **/dev/null**

Disk/file space usage (df, du)

df stands for "disk free."

Reports the amount of disk space used and available on mounted file systems.

Useful switches:

df -h Human readable; uses friendly k, M, and G indicators to show file size rather than listing them in bytes

df -T Show filesystem types

Ֆայլերի որոնում

find search for files

```
find /etc -name "passwd*" 2> /dev/null

find ~ -name "f*“

find ~ -type l 2> /dev/null
          -type   d   directory
                  f   regular file
                  l   symbolic  link
find ~ -name "f*" -type l 2> /dev/null

find /bin -size +20000k
find /usr/sbin -size 2M -name "t*" -exec ls -l {} \;
find /usr/sbin -size +100k -name "t*" -ok cp {} /tmp \;
```

Ֆայլերի որոնում

locate - find files by name

locate – works similar to ‘find’, but it searches not in the real filesystem, but in the previously built database.

yum search locate
yum install mlocate
/etc/cron.daily/ mlocate.cron

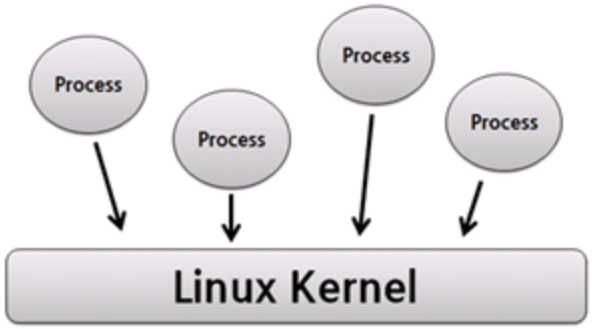
Պրոցեսներ

What is a process?

- A binary command
- Loaded into memory
- Processed by CPU
- Release memory when quit
- Identified by PID

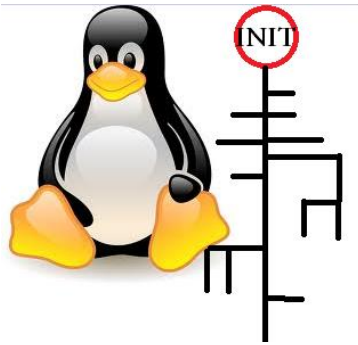
Պրոցեսներ

- **Պրոցես = աշխատող ծրագիր**



Պրոցեսի պարամետրեր՝

- **Եզակի համար** **Process ID (PID)**
- **Ծնողի համար** **Parent Process ID (PPID)**
- **Վիճակ** **State**
- **Տերմինալ** **TTY** (daemons run without TTY)
- **Իրական օգտագործողի համար Real user ID (RUID)**
- **Գործող օգտագործողի համար Effective user ID (EUID)**
- **Իրական խմբի համար Real group ID (RGID)**
- **Գործող խմբի համար Effective group ID (EGID)**



Պրոցեսներ

Վիճակներ/States՝

- Running / Runnable (R)
- Sleeping
  - Interruptable (S)
  - Uninterruptible (D)
- Stopped (T)
- Defunct / Zombie (Z)

Պրոցեսներ

Գործիքներ՝

- ps
- pstree
- top
- htop
- gnome-system-monitor

Օրինակներ՝

- **ps**
- **ps l**
- **ps -o pid,ppid,user,cmd**

- **ps aux**
- **ps -eo pid,ppid,user,cmd**

- **pstree**
- **pstree -u –p**
- **top** ելք՝ “q”

Պրոցեսներ

Բացեք 3 տերմինալ,

երկու պատուհանում տվեք **cat** հրամանը

3-րդ տերմինալում տվեք հետևյալ հրամանը

ps aux | grep cat

Պրոցեսներ

Պրոցեսի կառավարում՝

- **kill**
- **kill -l**
- **kill [-signnum] <PID>**
- **pkill <process_name>**
- **pgrep -l bash**
- **pidof <process_name>**
- **killall [-signum] <process_name>**

signum	*Name*	*Signal Effect*
1	HUP	RELOAD CONFIGURATION
2	INT	INTERRUPT. Is sent when you press **Ctrl+C**
3	QUIT	QUIT WITH SOME ACTIONS. Is sent when you press **Ctrl+**
9	**KILL**	TERMINATE IMMEDIATELY. ***CAN'T BE IGNORED***
15	TERM	TERMINATE NORMALLY (closing open files, etc.). ***DEFAULT SIGNAL***
18	CONT	CONTINUE—resume processing; undo the effect of a SIGSTOP signal.
19	**STOP**	STOP—suspend program operation. Is sent when you press **Ctrl+Z**

Պրոցեսներ

Ետին պլանի պրոցեսներ՝

Հրամաններից հետո & նշան դնելիս ուղարկում ենք ետին պլան (background)
command &

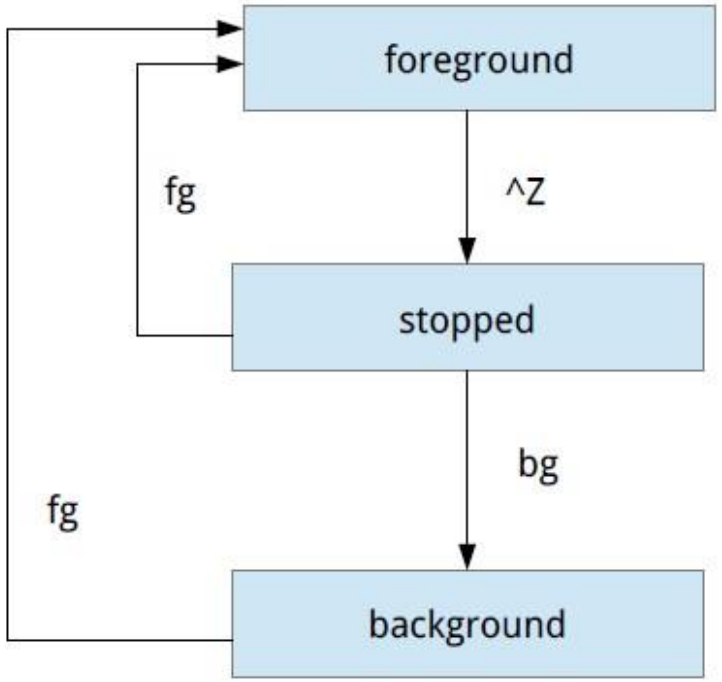
- **jobs** - List the active jobs
- **kill %n** - Kill the foreground job
- **fg [#n]** - Resume job in the foreground, and make it the current job.

Օրինակներ՝

- **jobs**

[1]- Stopped man ls
[2]+ Stopped top

- **fg 1**



Այսօր Լինուքսը մատչելի է զանազան տարբերակների՝ **հավաքածուների** ձևով (Linux Distributions)

Օրինակներ՝

- RedHat Family
  - Fedora
  - CentOS Stream
  - RHEL
  - Rocky Linux, Alma Linux
- Debian
- Ubuntu
- …
- Arch Linux
- Alpine Linux
- OpenWRT (for embedded devices)

Linux Distributions

Լինուքսի հավաքածուները

(Linux Distributions)

- **ներառում են բազմաթիվ տարատեսակ ԾՐԱԳՐԵՐ**
- **ունեն ԾՐԱԳՐԵՐԻ օնլայն պահոցներ (Repositories)**

Linux Packages & Repositories

Package Management System

- **Red Hat (RPM) Package Manager (YUM/RPM)**
- **Debian (DEB) GNU/LINUX Package Manager (APT/DPKG)**

Package Management Systems benefits:

- **INSTALLING**
- **UPDATING**
- **REMOVING**
- **MANAGING**

Online Repositories

- **Allow online package management**

Example: Android Repositories

(APK - Android packages)

Google Play, Amazon Appstore

Linux Package Managers
RPM - RPM/RedHat Package Manager

- **RPM - package manager**

  Examples of **rpm** usage:

  - **rpm -qa** # show the list of installed packages
    - **rpm –qa | less**
  - **rpm -qi** **${package name}** # show info about the package
    - **rpm –qi mc**
  - **rpm -ql** **${package name}** # show the list files that belong to the installed package
    - **rpm –ql mc**
  - **rpm -qf** **${file name}** # show which installed package does the file belong to
    - **rpm –qf /usr/bin/mc**
  - **rpm -ivh** **${package-file-name/url}** # install package
  - **rpm -Uvh** **${package-file-name/url}** # update package
  - **rpm -e** **${package name}** # erase (remove) package

EPEL Repository

- **EPEL - Extra Packages for Enterprise Linux**

Additional packages for Enterprise Linux

- Red Hat Enterprise Linux (RHEL)
- CentOS
- Scientific Linux (SL)
- Oracle Linux (OL)
- AlmaLinux (AL)
- Rocky Linux (RL)

- **dnf config-manager --set-enabled powertools** (for version **8**)
- **dnf config-manager --set-enabled crb** (for version **9**)
- **dnf install epel-release**

PowerTools is not available in RHEL 9-based systems instead its equivalent repository which is known as CRB (Code Ready Builder)

Linux Package Managers

YUM - Yellowdog Updater Modified (RH/CentOS 5-7), **DNF** - Dandified YUM

- List all available packages: **yum/dnf list**
  - **yum/dnf list | less**
- List installed packages: **yum/dnf list installed**
  - **yum/dnf list installed | less**
- Get information about a package: **yum/dnf info package**
  - **yum/dnf info mc**
- Show which package does the file belong to: **yum/dnf provides 'filename'**
  - **yum/dnf provides /etc/passwd**
- Check for Available Updates: **yum/dnf check-update**
  - **yum/dnf check-update | less**
- Update all installed packages: **yum/dnf update**
  - **yum/dnf update**
- Update a specific package: **yum/dnf -y update <package-name>**
  - **yum/dnf update mc**

Linux Package Managers

YUM - Yellowdog Updater Modified, **DNF** - Dandified YUM (RH/CentOS 8…)

- Search for a package: **yum/dnf search** [keyword]
  - **yum/dnf search**
- Install a package: **yum/dnf install** [package]
  - **yum/dnf -y install mc**
- Remove a package: **yum/dnf -y remove** [package]
  - **yum/dnf -y remove mc**
- List the Groups of packages: **yum/dnf grouplist**
  - **yum/dnf grouplist | less**
- Get information about the Groups of packages: **yum dnf groupinfo**
  - **yum/dnf groupinfo "DNS Name Server" | less**
- Install a Group of packages: **yum/dnf groupinstall** [“package group”]
  - **yum/dnf groupinstall "Development Tools"**
- Remove a Group of packages: **yum/dnf groupremove** [“package group”]
  - **yum/dnf groupremove "Development Tools"**
  - yum --setopt=group_package_types=optional groupinstall "Package Group"

Linux File Archives (tar,gzip,bzip2,lzma)

- **tar**
  - **tar cvf f.tar /etc** - Create a tar archive
  - **tar xvf f.tar** - Extract tar archive
  - **tar tvf f.tar | less** - Test/View tar archive

gzip

gzip f.tar
gunzip f.tar.gz
tar zcvf f.tar.gz /etc
tar zxvf filename.tar.gz
tar ztvf filename.tar.gz

Bzip2

bzip2 f.tar
bunzip2 f.tar.bz2
tar jcvf f.tar.bz2 /etc
tar jxvf filename.tar.bz2
tar jtvf filename.tar.bz2

xz / lzma

tar Jcvf f.tar.xz /etc
tar Jxvf filename.tar.xz
tar Jtvf filename.tar.xz

Source Code Packages install

- Source code Linux packages are basically one of the following:
  - **<file>.tgz**
  - **<file>.tar.gz**
  - **<file>.tar.bz2**
  - **<file>.tar.xz**

- Source code install consists of the following steps:
  - **tar zxvf <file>.tgz**
  - **cd <dir>**
  - **./configure**
  - **make**
  - **make install**

Source Code Install Example htop

- **dnf install gcc make autoconf automake**

- https://github.com/htop-dev/htop/releases/
  - **wget https://github.com/htop-dev/htop/archive/refs/tags/3.2.2.tar.gz**
  - **tar zxvf 3.2.2.tar.gz**
  - **cd htop-3.2.2**
  - **./autogen.sh && ./configure && make**
  - **./htop --version**
  - **htop --version**

  What's the difference ?

  How to make last command work ?

  How to install htop from repository ?

  Can we have two versions of same program ?

Տեքստային տվյալների մշակում

- cat
- tac
- head
- tail
- less
- sort
- wc
- grep
- tee
- cut
- awk

Տեքստային տվյալների մշակում

- **cat / tac / head / tail**
- **cat > f11** ֆայլի ստեղծում, ավարտը **Ctrl-D**
- **head -15 f7**
- **tail -1 f7**
- **tail -f /var/log/messages**

Տեքստային տվյալների մշակում

less

Enter/DOWNARROW – մեկ տող ներքև
SPACE / Page Down – մեկ էկրան ներքև
PgUp / b – մեկ էկրան վերև
UPARROW – մեկ տող վերև
/ – որոնում
Home – անցնել տեքստի սկիզբը
End – անցնել տեքստի վերջը
q – ելք

Տեքստային տվյալների մշակում

sort /etc/passwd > sorted.txt

wc -l /etc/passwd

wc -l < /etc/passwd

ps ax | wc -l

Տեքստային տվյալների մշակում

grep - line text search utility (Global Regular Expression Print).

Options:

-v Invert the sense of matching, to select non-matching lines

-i Ignore case distinctions in both the pattern and the input files

-r Process all directories recursively

-s Suppress error messages about nonexistent or unreadable files

-n Show line number within its input file

Տեքստային տվյալների մշակում

Regular expressions

^	beginning of the line
$	end of the line
[ ]	any symbol within brackets like [0-9] – numbers, [a-zA-Z] – all latin characters
[^]	any symbol except those in brackets
****	causes the metacharacter to be treated as a literal character. For example \$ means $, and not a \ at the end of line, and \\$ means \ at the end of line
.	any single symbol
*****	repetition of previous wildcard 0 or more times. Thus **.*** expression means - ”any amount of any symbols”

Օրինակներ՝

grep 'bash$' /etc/passwd

select the line with bash at the end from /etc/passwd

grep ^s.s /etc/passwd

select all lines in the file that begin with the letter 's', followed by any one character, then the letter 's' from /etc/passwd

grep ^r /etc/passwd

select all lines from /etc/passwd that begin with ‘r’ from /etc/passwd

ls /bin | grep ^f

select all files from ls output that begin with ‘f’ (i.e. all files beginning with f in /bin directory)

grep -r -n bash /etc/* | grep -v bashrc

cat /etc/passwd | grep -E ^'(rt|us|ap)‘

grep -E ^'(rt|us|ap)' /etc/passwd

Select only lines beginning with either ‘**rt**’ or ’**us**’ or ‘**ap**’

-E use Extended Regular Expressions

Տեքստային տվյալների մշակում

tee - read from standard input and write to standard output and files

cat /etc/passwd | grep -E ^'(r|u|a)' | tee /tmp/rua

cut - extract sections/fields from each line (of files)

-f field number

-d delimiter symbol

cat /etc/passwd | cut -f1,3 -d":"
cut -f1,3 -d":" /etc/passwd
Select 1-st and 3-rd fields from /etc/passwd separated by “**:**”

grep -r -n bash /etc/sys* 2>/dev/null | grep -v bashrc | cut -f1 -d":"

Տեքստային տվյալների մշակում

awk - extract sections/fields from each line of files

```
cat /etc/passwd  | awk  -F":"  '{print $1}'
awk  -F":"  '{print $1}'   /etc/passwd

cat /etc/passwd | grep ^r | awk -F":" '{print $1} '

tail -10 /etc/passwd | awk -F":" '{print $3"--"$1} '

cat /etc/passwd | grep -E ^'(rt|us|bi)' | awk -F":" '{print "TEST: "$3"  "$1}'

ls -l /usr/sbin | grep -E '(b|c)'$ | awk '{print "FILE: "$9}'
```

Տեքստային տվյալների մշակում

awk Examples

Count sum and average (NR – number of lines)

cat >f11

cat f11 | awk '{ s += $1 } END { print "sum is", s, " average is", s/NR }‘

Numbered list:

cat f11 |awk 'BEGIN { print "Numbered list:" } { print NR, ":\t", $0 }'

Move multiple files in another directory and rename

touch ff1 ff2 ff3 ff4 ; mkdir dd11

ls ff* | awk '{print "mv "$0" dd11/"$0".dat"}' | bash

ls dd11

Login / Logout

Global Login Config Files:

/etc/profile

/etc/bash.bashrc

Per-User Login Config Files:

~/.bash_profile / ~/.bash_login / ~/.profile

~/.bashrc

~/.bash_logout

Կարևոր ֆայլերի կառուցվածք (/etc/passwd)

1 2 3 4 5 6 7

root:x:0:0:Administrator:/root:/bin/bash

1	**account**	The user's login name.
2	**password**	The users encrypted password or a place holding character **(x)** if the system is using shadow passwords and storing the password in the **/etc/shadow** file which is readable only by root.
3	**UID**	The users numerical identificator
4	**GID**	The number of the primary group for the user.
5	**GECOS**	Usually has the full user name. This field is only for information purposes and is optional. This information is sometimes called the user's finger information.
6	**directory**	The full path of the user's home directory.
7	**shell**	The full path and filename of the user's shell. If no value is here /bin/sh is assumed. Each user can change his/her shell with the **chsh** command.

Կարևոր ֆայլերի կառուցվածք (/etc/shadow)

1 2 3 4 5 6 7 8

root:$1$1U0StR.p$qS9b5rzLf/tNS0BjttvNB0:11156:0:__:__:__:__:

1	account	The user's login name.
2	password	The users encrypted password. Good password should be minimum 6-8 characters long including special characters/digits
3	change	Days since Jan 1, 1970 that password was last changed
4	min	The minimum number of days required between password changes i.e. the number of days left before the user is allowed to change his/her password
5	max	The maximum number of days the password is valid (after that user is forced to change his/her password)
6	warn	The number of days before password is to expire that user is warned that his/her password must be changed
7	inactive	The number of days after password expires that account is disabled
8	expire	Days since Jan 1, 1970 that account is disabled i.e. an absolute date specifying when the login may no longer be used

The first field is a numerical number that tell's you the hashing algorithm that's being used.

- **$1** = MD5 hashing algorithm.
- **$2** =Blowfish Algorithm is in use.
- **$2a**=eksblowfish Algorithm
- **$5** =SHA-256 Algorithm
- **$6** =SHA-512 Algorithm

The second field is the salt value. Salt value is nothing but a random data that's generated to combine with the original password, inorder to increase the strength of the hash.

The last field is the hash value of salt+user password.

Կարևոր ֆայլերի կառուցվածք (/etc/group)

1 2 3 4

root:x:0:root,user

1	**groupname**	Group name
2	**grouppasswd**	Encrypted group password (If this field is empty, no password is needed)
3	**GID**	The group numerical identificator
4	**userlist**	Comma-separated list of group members' usernames (those, having other primary group in /etc/passwd)

Primary and additional groups

Every user must be a member of at least one group, called his primary group.

Primary group is specified in **/etc/passwd**

User can be a memeber of a number of additional groups specified in **/etc/group**

id command will display user's current groups

Պարբերական պրոցեսներ - Cron

- **cron** համակարգը թույլ է տալիս կարգավորել որոշակի պորցեսների (ինչպես օրինակ՝ լոգ ֆայլերի ռոտացիան, արխիվայցումը, ժամանակավոր ֆայլերի հեռացումը, և այլն) պարբերական կատարումը:
- Դրա համար համակարգում պետք է տեղադրված լինի **cron** փաթեթը և աշխատի **crond** պրոցեսը, որն էլ իրականացնում է տվյալ գործը:
- /etc/crontab – **cron** փաթեթի գլխավոր կարգավորման ֆայլ
- Լրացուցիչ կարգավորման դիրեկտորիաներ/ֆայլեր:
  - **/var/spool/cron/**
  - **/etc/cron.hourly/**
  - **/etc/cron.daily/**
  - **/etc/cron.weekly/**
  - **/etc/cron.monthly/**

Պարբերական պրոցեսներ - Cron

- Նոր պարբերական պրոցես գրանցելու համար օգտագործողը պետք է տա հետևյալ հրամանը՝
  crontab -e (**crontab -l** **crontab -r**)
- Այն կբացի ստանդարտ տեքստային խմբագրիչը (default editor) որն մեծ մասամբ **vi**-ն է:
- Այլ խմբագրիչ նշանակելու համար պետք է կատարել հետևյալ կարգավորումներ՝
  export EDITOR=/usr/bin/nano
- Ամուռքագրեք հետևյալը և պահպանեք՝
  ***/2 * * * * /bin/date >> /tmp/dout**

```
.---------------- minute (0-59)
|  .------------- hour (0-23)
|  |  .---------- day of month (1-31)
|  |  |  .------- month (1-12) OR jan,feb,mar,apr,...
|  |  |  |  .---- day of week (0-6) (Sunday=0/7) OR sun,mon,tue,wed,thu,...
|  |  |  |  |
*  *  *  *  * username  command to be executed
*/7 * *  *  * root     cd / && run-parts --report /etc/cron.hourly
25  6 *  *  * root     test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.daily )
47  3 *  *  7 root     test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.weekly )
52  4 1  *  * root     test -x /usr/sbin/anacron || ( cd / && run-parts --report /etc/cron.monthly)
```

username դաշտը առկա է միայն գլխավոր **/etc/crontab** ֆայլում

https://crontab.guru/

Shell-scripting basics

- First line of Shell Script should look like:

#!/bin/bash

- Test comparisons

File check

[-e FILE]	**#FILE exists**
[-f FILE]	**#FILE exists and is a regular file**
[-d FILE]	**#FILE exists and is a directory**
[-h FILE]	**#FILE exists and is a symbolic link**
[-r FILE]	**#FILE exists and read permission is granted**
[-w FILE]	**#FILE exists and write permission is granted**
[-x FILE]	**#FILE exists and execute permission is granted**
[-s FILE]	**#FILE exists and has a size greater than zero**
[-u FILE]	**#FILE exists and its set-user-ID (SUID) bit is set**

Shell-scripting basics

- Test comparisons

String check

[-z $STRING] **#the length of STRING is zero**

[-n $STRING] **#the length of STRING is nonzero**

[$STRING1 = $STRING2] #the strings are equal

[$STRING1 != $STRING2] #the strings are not equal

Numeric check

[$INTEGER1 -eq $INTEGER2] **#INTEGER1 is equal to INTEGER2**

[$INTEGER1 -ne $INTEGER2] **#INTEGER1 is not equal to INTEGER2**

[$INTEGER1 -gt $INTEGER2] **#INTEGER1 is greater than INTEGER2**

[$INTEGER1 -lt $INTEGER2] **#INTEGER1 is less than INTEGER2**

Shell-scripting basics - Flow control

Bash supports the following flow control constructs:

if / else	Execute a list of statements if a certain condition is / is not true
for	Execute a list of statements a fixed number of times
while	Execute a list of statements repeatedly while a certain condition holds true
until	Execute a list of statements repeatedly until a certain condition holds true
case	Execute one of several lists of statements depending on the value of a variable
select	Generate simple menus

Shell-scripting basics - Script examples

```
#!/bin/bash
if [ -f /etc/passwd ]
then
  echo "/etc/passwd - regular file"
else
echo "/etc/passwd - not regular file"
fi
```

```
#!/bin/bash
if [ -f $1 ]
then
  echo "$1 - regular file"
else
echo "$1 - not regular file"
fi
```

cat >> p1
mcedit p1
nano p1

chmod +x p1

./p1 /etc
./p1 /etc/group

/etc – not regular file
/etc/group – regular file

Shell-scripting basics - Script examples

```
#!/bin/bash
if [ -z $1 ]; then
echo "Usage: $0 number of loops"
exit
fi
clear
COUNTER=0
while [ $COUNTER -lt $1 ]
do
echo "State (for $1 seconds)"
echo "second:$COUNTER"
echo "-- Users --"
w
echo "----------------"
/bin/sleep 1
clear
COUNTER=`expr  $COUNTER + 1`
done
```

Shell-scripting basics - Script examples

```
#!/bin/bash
echo "How do you like it:"
for (( i=1; i<=5; i++ ))
do
    for (( j=1; j<=i;  j++ ))
    do
     echo -n "$i"
    done
    echo ""
done
```

http://www.freeos.com/guides/lsst/ch08.html#q7
http://linuxcommand.org