53 – Users and Group administration

- **Purpose of Users in Linux**

  - Security
  - Own work space
  - Processes belonging to users

- **Adding Users**

  - `yast`
  - `useradd -m` *`username`* and `passwd` *`username`*
    (`-m` is for copying a skeleton in user home directory)
  - Purpose of `/etc/skel` and `/home` directories
  - Purpose of `/etc/default/useradd` file Parameters
    used by YAST (entering new users)
  - Purpose of `/etc/login.defs`

- **Key files, *terms*, and utilities include:**

  `/etc/passwd`
  `/etc/shadow`
  `/etc/group`
  `/etc/gshadow`

  `useradd`
  `usermod`
  `userdel`
  `passwd`
  `chage`

  `groupadd`
  `groupmod`
  `groupdel`
  `gpasswd`

  `pwconv`
  `pwunconv`
  `grpconv`
  `grpunconv`

- **The `/etc/passwd` file purpose and format**

  This file contains the users account info. One per line. Fields are separated by '`:`'.
  File Format:

```
 username : x : userID : GroupID : UserInfo : HomeDir : Shell
    1       2      3        4          5          6        7
```

  Field 2= `Password`, `x`=reference to `/etc/shadow`, empty=no password, `*`or `!`=no login possible

- **The `/etc/shadow` file purpose and format**

  If the shadow password system is installed, this file contains the encrypted passwords for each user and their expiry parameters.

  Line fields: The fields are separated by the char. '`:`'.
  Here are the fields' sequence:

  1. User login name
  2. Encrypted password **:** (empty=no Passwd, *=no login possible)
  3. Days since Jan 1, 1970 password was last changed **:** (never empty)
  4. Days until change allowed **:** (0=always allowed to change)
  5. Days before change required **:** (Normal is 9999 days)
  6. Days warning before expiration **:** (empty=no warning)
  7. Days before account inactive **:** (empty= never inactive)
  8. Days since Jan 1,1970 when account will be disabled **:**
     (empty = will never be disabled)
  9. Reserved for future use

- **User accounts administration.**
  The user accounts are located in the file `/etc/passwd`, their encrypted passwords are in `/etc/shadow` (if the shadow password system is installed). When a new user account is created (using `useradd`) the default template (`-m` option) used to create the user's personal and work directory is `/etc/skel`.

  Users admin commands:

  `useradd [OPTIONS] username` Adds a user to the system

  options

  `-d default_home_directory_path`
  (Note: *username* will NOT be added to the path)

  `-e default_expire_date`
  The date on which the user account is disabled.

  `-f default_inactive` The number of days after a password has expired before the account will be disabled.

  `-g default_group` The group name or GID for a new user's main group.

  `-s default_shell` The name and location of the new user's login shell.
  Command allowing a user to change permanently his own shell in `/etc/passwd`:
  eg. `chsh -s /bin/ash`
  Shell must be in the list of valid shells`/etc/shells` .

  `-m` Copies the directory `/etc/skel` to user home directory

  `-k template_dir` Combined with `-m` will copy use the `template_dir` instead of `/etc/skel`.

  Note: When certain defaults are not given via options then they are taken from the file `/etc/default/useradd` file. These default parameters can also be seen using : `useradd -D`

  eg.
  ```
  GROUP=100
  HOME=/home
  INACTIVE=-1
  EXPIRE=
  SHELL=/bin/bash
  SKEL=/etc/skel
  GROUPS=dialout,video
  ```

  `/etc/login.defs` This file is for setting the extra defaults after login.

  `usermod username` Modifies the existing user's login parameters
  ```
  [-c comment]          [-d home_dir [ -m]]
  [-e expire_date]      [-f inactive_time]
  [-g initial_group]    [-G group[,...]]
  [-l login_name]       [-p passwd]
  [-s shell]            [-u uid [ -o]] [-L|-U]
  ```

  `userdel username` Deletes a user from the system
  `-r` Deletes the user's home directory as well !!!

  `passwd [username]` Changes the password of user.
  Allowed characters in password are:
  `# * , . ; : _ - + ! $ % & / | ? { [ ( ) ] }`

chage [*options*] *username*

Used to list (-l) or change the user's password expiry parameters. Options:

```
[-D binddn]     [-P path]     [-m mindays]
[-M maxdays]    [-d lastday] [-I inactive]
[-E expiredate] [-W warndays]
```

newusers *Filename* Update and create new users in batch mode.

chpasswd *UserSPaSSFILE*

Modifies the password of multiple users in batch mode
One line per user: *username password*(clear text)

Extra and login user related commands:

id -ng [*username*]	Shows the Present Effective group of a user.
id -nG [*username*]	Shows all the groups the present user belongs to.
groups [*username*]	" " " " " " " " " " " "
id -nu [*username*]	Shows the current username of a user.
echo $USER	" " " " " " " "
id -u	" " " " " " user ID of a user.
users	users presently logged in locally (short format)
who	"" "" " "" "" (long format)
w	"" "" " locally (long format)
finger [-l *username*]	"" "" " locally or remotely (long format) (Information in [room_no] below will not be shown)

chfn [*options*] Changes the users info(field 5) in the /etc/passwd. (for scripting purposes) Options:

```
[-f full_name] [-r room_no] [-w work_ph]
[-h home_ph]   [-o other]   [user]
```

Each field will separated with comas (**,**)
Characters NOT allowed are: **,** **:** **=** or Ctrl chars.

Note: Information of the user can be displayed with the command:

```
finger -l username
```

lastlog Shows the last logins that happened since the log file /var/log/lastlog (binary format!!) was created . The list includes booting and shutdowns and login in previous days. [-u *username*] [-t *days_before*]

last Displays all the proper logins that happened since the last creation of the (binary format!!) log file /var/log/wtmp..
This file is regularly compressed and made new.

lastb Displays all the proper logins that happened since the last creation of the (binary format!!) log file /var/log/btmp.
This file is regularly compressed and made new.

- **Groups Administration:**
  - A user can be participant to more than one group at the same time.
  - A user who is member of a group can change to that group without password but a user NOT member can only change to that group if the group password exist and the user gives it.
  - One or more users can become group administrators for specific groups.
    - Group Administrators can:
      - ➢ add/change/delete the password of the group
      - ➢ add/delete users to the group
      - ➢ reserve the group to members-only

- **Groups administration commands:**

`groupadd [options] group`

System administrator (root) adds a group to the system. Options:

`-g gid` The numerical value of the group's ID.
Value must be non-negative.
This value must be unique, unless the `-o` option is used.

`-o` Allows to assign an existing ID to a group.
The default is to use the smallest ID value equal or greater than `GID_MAX` from `/etc/login.defs` and greater than every other group. Values between `0` and lower than `GID_MIN` are typically reserved for system accounts.

`-r` This flag instructs groupadd to add a system account. The first available `gid` lower than `GID_MAX` will be automatically selected unless the `-g` option is also given on the command line.

`groupmod [-g newgid] [-n newname] group`

System administrator modifies a group settings.

`-g newgid` changes the gid of the group.
`-n newname` changes the name of the group.

`groupdel group` System administrator deletes a group to the system

System administrator(root) `gpasswd` options:

`gpasswd [options] group`

adds/changes the group's password.
Note: The group's password is only needed if a user, which is not a member of the group, wants to temporarily become one and have it as its effective group.
He will be prompted to give the group's password.
Options:

`-R` Makes the group reserved to members-only.
Result: No change of group through `sg` or `newgrp` is allowed for non-members.

The password in `/etc/gshadow` becomes `'!'`

`-A` *`user,...`* adds Group **A**dministrator(s) to a group.

`-M` *`user,...`* adds Group **M**ember(s) to a group.

`-r` *`group`* Removes the password for the group.
The group is then also reserved for members-only
Password in `/etc/gshadow` becomes `'!'`.

Group administrators `gpasswd` options:

`gpasswd [`*`options`*`]` *`group`*

Adds a new password to a group.

Options:

`-a` *`user`* Adds permanently a user to a group

`-d` *`user`* Deletes permanently a user from a group.

`-r` *`group`* Removes the password for the group.(same as with root)

`newgrp` *`group`*
or `sg` *`group`*
A user changes itself temporarily to a new group. If the user is not a permanent member, password is asked. The user will be denied access if the group password is empty and the user is not a permanent member.

`sg` *`group`* `-c` *`command`* Runs a command as participant of the given group and returns to normal after the command finishes.

`grpck` *`group`* System administrator checks a group.

The groups configuration files:

`/etc/group` Where all the users for each groups are listed.
(Note: The default (main)group of users does not contain the users names)

Format:

`groupname :` *`Password`* or `x` or `! : GID :` *`Memberlist`*
(eg. user1,user2.. Users' list are separated by comas)

`/etc/gshadow` Where the groups passwords are kept

Format:

`groupname` `:` *`Password`* `or ! :` *`AdminUsersList`* `:` *`MemberUsersList`*
(Admin and Users' list are separated by comas)

- **Converting to/from the standard(older) to shadow(newer) password system.**
  In the older times of Unix, this file was containing the whole encrypted (`DES`) password in the second field. Because of new faster computers being capable of reverse decoding the passwords, the shadow password system got introduced. If the shadow password system is installed and activated, the second field of

/etc/passwd file contains an 'x', to indicate that the user has a password and it is located in /etc/shadow file.

To convert from one password system to another the following commands are used:

pwconv	Converts all the users passwords from the older system to shadow password system. It creates the file /etc/shadow.
pwunconv	Converts all the users passwords from the shadow system to older system. The /etc/shadow is then erased.
grpconv	Similar to pwconv except that it applies it to the groups. Converts all the group passwords from the older system to shadow password system. It creates the file /etc/gshadow.
grpunconv	Similar to pwunconv except that it applies it to the groups. Converts all the group passwords from the shadow system to older system. The /etc/gshadow is then erased.

- **Checking the consistence of passwords and groups files:**
  Two tools are available for checking the consistence of the user's and groups accounts files.

pwck [*options*] Checks the user's accounts files for consistency.
(/etc/passwd and /etc/shadow)
Checks are made to verify that each entry has:
- the correct number of fields
- a unique user name
- a valid user and group identifier
- a valid primary group
- a valid home directory
- a valid login shell.

Options:

-r This causes all questions regarding changes to be answered 'no' without user intervention.

-s Sorts entries in /etc/passwd and /etc/shadow by UID.

grpck [*options*] Checks the group's accounts files consistency.
(/etc/group and /etc/gshadow)
Checks are made to verify that each entry has:
- the correct number of fields
- a unique group name
- a valid list of members and administrators.

Options:

-r This causes all questions regarding changes to be answered 'no' without user intervention.

-s Sorts entries in /etc/passwd and /etc/shadow by UID.

- **Tips & Tricks**
  - Showing all the registered users and their groups in details.

```
Grep ':[1-9][0-9]..:' /etc/passwd | cut -d: -f1) ; \
     for user in $users; do id $user ; done ;
```

  - Producing an encrypted password through `crypt()` function.

```
echo ClearTextPassword | mkpasswd -s
```

  - Disabling a User account without deleting anything:
    - by Adding a `*` or a `!` to the encrypted password in `/etc/shadow` file**.**
  - Preventing a user from login in with a shell:
    - by changing the shell of the user to `/bin/false`
    eg. `usermod -s /bin/false username`

Benutzer und Gruppenverwaltung

• Befehleliste

mit YaST: Benutzerverwaltung
Das Verzeichnis `/etc/skel`

```
useradd -m Benutzer
usermod -u uid -l new_loginname -d new_home_dir -s new_shell loginname
usermod -G group1,group2,... (add groups to existent group) loginname
passwd -S (Status), -x (Maxdays), -w (Warning), -l (lock=disable), -u (enable)
userdel -r Benutzer
groupadd -g gid  Gruppenname
groupadd -g gid  -o Gruppenname (mit -o können zwei Gruppen die gleiche gid haben!)
groupdel Gruppenname
groupmod -g gid -n Neuen-Gruppenname Gruppenname
```

Die Dateien `/etc/passwd` & `/etc/shadow`
Die dateien `/etc/group` & `/etc/gshadow`

Benutzer im System sehen: `w, who, finger`