# Basics of unix administration

Krzysztof Kluza

10.01.2022

## User accounts

Most of the relevant user information is stored in the following files:

- `/etc/passwd` – basic information about user accounts,
- `/etc/group` – basic information about user groups,
- `/etc/shadow` – extended user account information (e.g. expiration dates) and encrypted password (in shadow system),
- `/etc/gshadow` – enhanced information on user groups (shadow).

As root, you can check the contents of the above files. You can also view their manual, especially focusing on the meaning of the fields in these files. In particular, you can create a user account by adding the appropriate entries in the above files or using commands: `adduser` or `useradd`. You can modify an account in many ways:

- `chfn` changes the GECOS information about a user (name, surname, etc.),
- `chsh` changes shell,
- `usermod` modifies any parameters of the account,
- `groupmod` as above for the group,
- `passwd` creates a new password, and in the shadow passwords system changes the expiration dates of the account.

## Modifying and blocking accounts

1. Using `id` and `groups`, check the information about user accounts and other accounts. Using `ls -ln` one can see the numeric values of the file owners.
2. Using `chfn` or `usermod`, modify the parameters of previously created accounts, in particular the personal data of the user or the account name.
3. Block a selected user account using `passwd` (do not change a password, only block it). Note the changes in the corresponding files.
4. Block a selected user account by changing the shell into `/bin/false` with the command `chsh` and observe changes in the corresponding files.

## Scheduling tasks

Crontab syntax

```
* * * * * command
- - - - -
| | | | +----- day of the week (0 - 7) (Sun=0, Mon=1, Tue=2,..., Sun=7)
| | | +------- month (1 - 12)
| | +--------- day of the month (1 - 31)
| +----------- hour (0 - 23)
+------------- minute (0 - 59)
```

Numeric values can be specified in various formats:

| | |
|---|---|
| `1-3` | i.e. 1, 2, 3 (range) |
| `0-10/2` | i.e. 0, 2, 4, 6, 8 i 10 (every second value from 0 to 10) |
| `1,2,5` | i.e. consecutive values 1, 2, 5 |
| `*/2` | every 2nd value (e.g. in the first column it is: 0, 2, 4, 6, ... 56, 58) |
| `1-3,5,6` | i.e. 1, 2, 3, 5 and 6 |

A useful tool for checking `crontab` entries: `https://crontab.guru/`.

Using `cron`:

1. Set the appropriate cron entry in the `crontab` table, which will check the logged in users in the system every 2 minutes today in the current hour (when we have classes).

2. Using `mail`, after 2 minutes, check if the command has been executed correctly.

3. Add to the `crontab` the appropriate entries for the following commands:

   `ls` – every hour on even days,

   `quota` – once a day on odd days,

   `ps` – every half hour of every first day of the month,

   `printenv` – every Monday at 6:00 a.m.,

   `who` – every four hours, but only from Monday to Friday,

   `finger` – every day at 6:20, 8:20, 12:20 i 18:20,

   `du` – at 12:00, 13:00, 14:00, 15:00; on the 1st, 13th, 20th of every month.

4. How to run a script daily using `cron` but without modifying the `crontab` table?