AUTOMATING TASK WITH JOB SCHEDULING

Cron daemon and the *cron* table (crontab) are the most useful tools for scheduling regular tasks. The first, *crond*, is a daemon that runs in the background. The *cron* daemon checks the *cron* table for which commands to run at specifies times.

The crontab file location: /etc/crontab

Time representation for Use in the crontab:

FIELD TIME UNIT	REPRESENTATION
1. Minute	0-59
2. Hour	0-23
3. Day of the Month	1-31
4. Month	1-12
5. Day of the week	0-7
6. User	e.g., root or something
7. command	e.g., man grep

To edit crontab, we can run the command:

```
Kali> crontab -e
```

The first time we run this command, it will ask which editor we would like to use. The default is /bin/nano.

Cron characters:

*-- any value
5,6—List of values (5 and 6)
1-4--- Range of values (1 to 10)
*/5—Step values (every 5)

Example: -

1. Run a job at minute 30, every hour
   30 * * * * <user> <command>
2. Every day at midnight
   00 00 * * * <user> <command>
3. Every day at 06:30 am
   30 06 * * * <user> <command>
4. Every Monday in April at 06:30 am
   30 06 * 04 1 <user> <command>
5. Midnight on the first of every month
   00 00 01 * * <user> <command>
6. Midnight at every weekday
   00 00 * * 1-5 <user> <command>
8. Every 5 minutes
   */5 * * * * <user> <command>

CRONTAB SHORTCUTS

The crontab file has some built in shortcuts we can use instead of a specifying the time, day and month.

1. @yearly
2. @annually
3. @monthly
4. @weekly
5. @daily
6. @midnight
7. @noon
8. @reboot

USING RC SCRIPTS TO RUN JOBS AT STARTUP

Whenever we start Linux system, a number of scripts are run to set up the environment for us. These are known as rc scripts. After the kernel has initialized and loaded all its modules, the kernel starts a daemon known as init or init.d. This daemon then begins to run a number of scripts found in /etc/init.d/rc. These scripts include commands for starting many of the services necessary to run our Linux system as we expect.

LINUX RUNLEVELS

Linux has multiple runlevels that indicate what services should be started at bootup. For instance, runlevel 1 is single-user mode, and services such as networking are not started in runlevel 1. The rc scripts are set to run depending on what runlevel is selected:

0	Halt the system
1	Single-user/minimal mode
2-5	Multiuser modes
6	Reboot the system

ADDING SERVICES TO rc.d

We can add services for the rc.d script to run at startup using the update-rc.d command. This command enables us to add or remove services from the rc.d script.

Syntax: update-rc.d <name of the script or service> <remove|defaults|disable|enable>

Kali> update-rc.d postgresql defaults

This adds the line to the file and then we have to reboot the system for the change to take place. The service run automatically at bootup.

ADDING SERVICES TO OUR BOOTUP VIA A GUI

We can download the rudimentary GUI-based tool rcconf from the Kali repository.

```
Kali> apt-get install rcconf
```

Then,

```
Kali> rcconf
```

This will open a simple GUI and we can scroll through the available services.