Contents

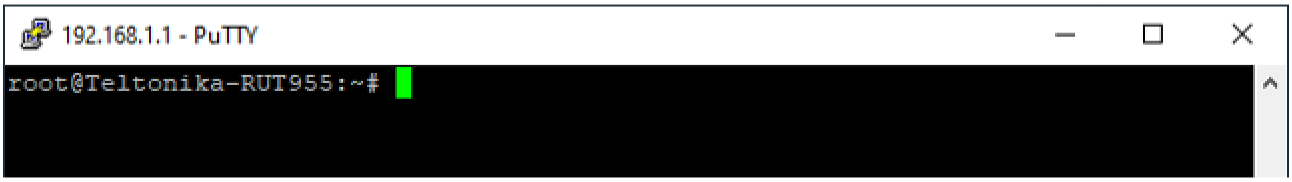
Basic scripting with ASH including Examples 2

1. Connect to the router via SSH 2
2. Keeping folder data after firmware upgrade 2
3. Writing scripts 3
4. Making the script executable. 4
5. Running scripts automatically using crontab 5

Basic Scripting

Basic scripting with ASH including Examples

1. Connect to the router via SSH

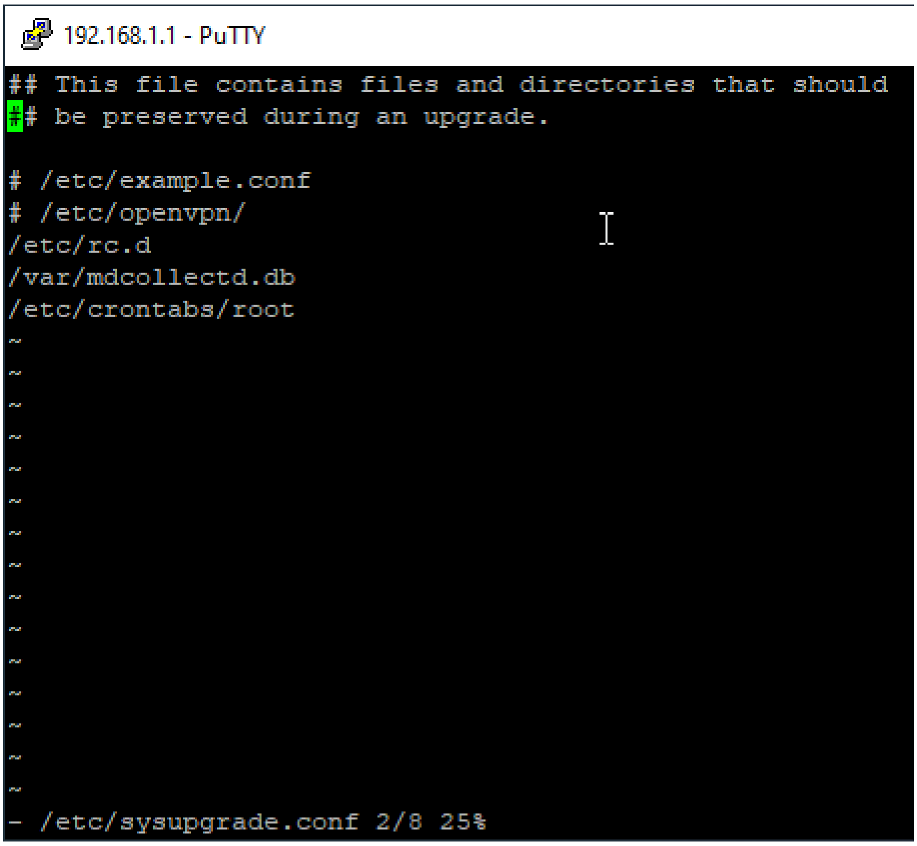


```
root@Teltonika-RUT955:~#
```

2. Keeping folder data after firmware upgrade

Before we start scripting we need to make sure that our scripts won‘t get lost after the firmware upgrade. To do that we need to exclude a specific folder, we can do it by editing sysupgrade.conf file.

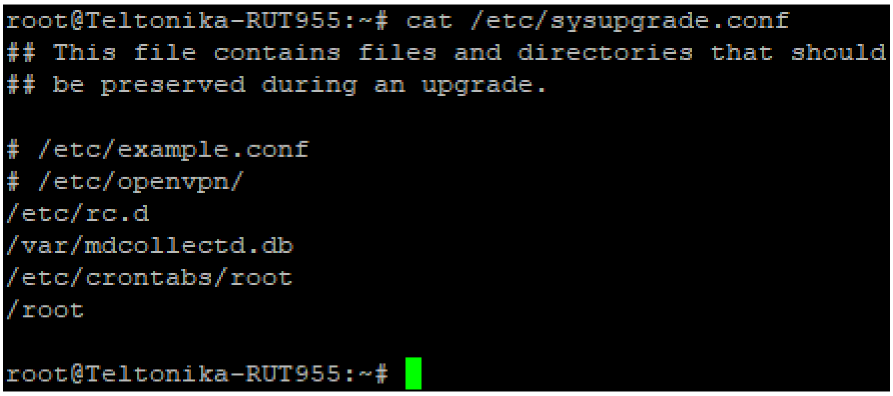
1. Open sysupgrade.conf with vi editor:
   <vi /etc/sysupgrade.conf>



```
## This file contains files and directories that should
## be preserved during an upgrade.

# /etc/example.conf
# /etc/openvpn/
/etc/rc.d
/var/mdcollectd.db
/etc/crontabs/root
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
- /etc/sysupgrade.conf 2/8 25%
```

2. Press <Insert> on your keyboard to enter editing mode
3. Add another entry at the botton of the list with the folder you will be keeping your scripts. (In my case I‘m using /root)
4. Press <ESC> on your keyboard to exit editing mode
5. Press <:wq> and enter to save and quit



```
root@Teltonika-RUT955:~# cat /etc/sysupgrade.conf
## This file contains files and directories that should
## be preserved during an upgrade.

# /etc/example.conf
# /etc/openvpn/
/etc/rc.d
/var/mdcollectd.db
/etc/crontabs/root
/root

root@Teltonika-RUT955:~#
```

3. Writing scripts

We can write scripts with a VI editor. You can use this Hello World as a template for easier start

```
#!/bin/bash
echo "Hello World"
```

For this example we are using a simple script which changes WiFi SSID to RUT955_1 and adds +1 every iteration. For example. Our SSID is RUT955_1, after script runs it changes to RUT955_2, when it runs again it changes to RUT955_3 and so on.

```
#!/bin/sh

#Variable with location of variable file

varfile="/root/var"

#Variable with current iteration number

var=`cat "$varfile"`

#UCI command that changes WiFi SSID to RUT955_[Whatever is in variable file]

uci set wireless.@wifi-iface[0].ssid='RUT955_'$var

#Commits UCI changes

uci commit wireless

#Reloads luci WebUI

luci-reload

#Updates variable file with current iteration number + 1

echo "$((var + 1))">/root/var
```

For this script to run, you need to also have a file called var in /root/ directory, this file has to contain a character <1>. This is the number from which the script is going to grab current iteration.

4. Making the script executable.

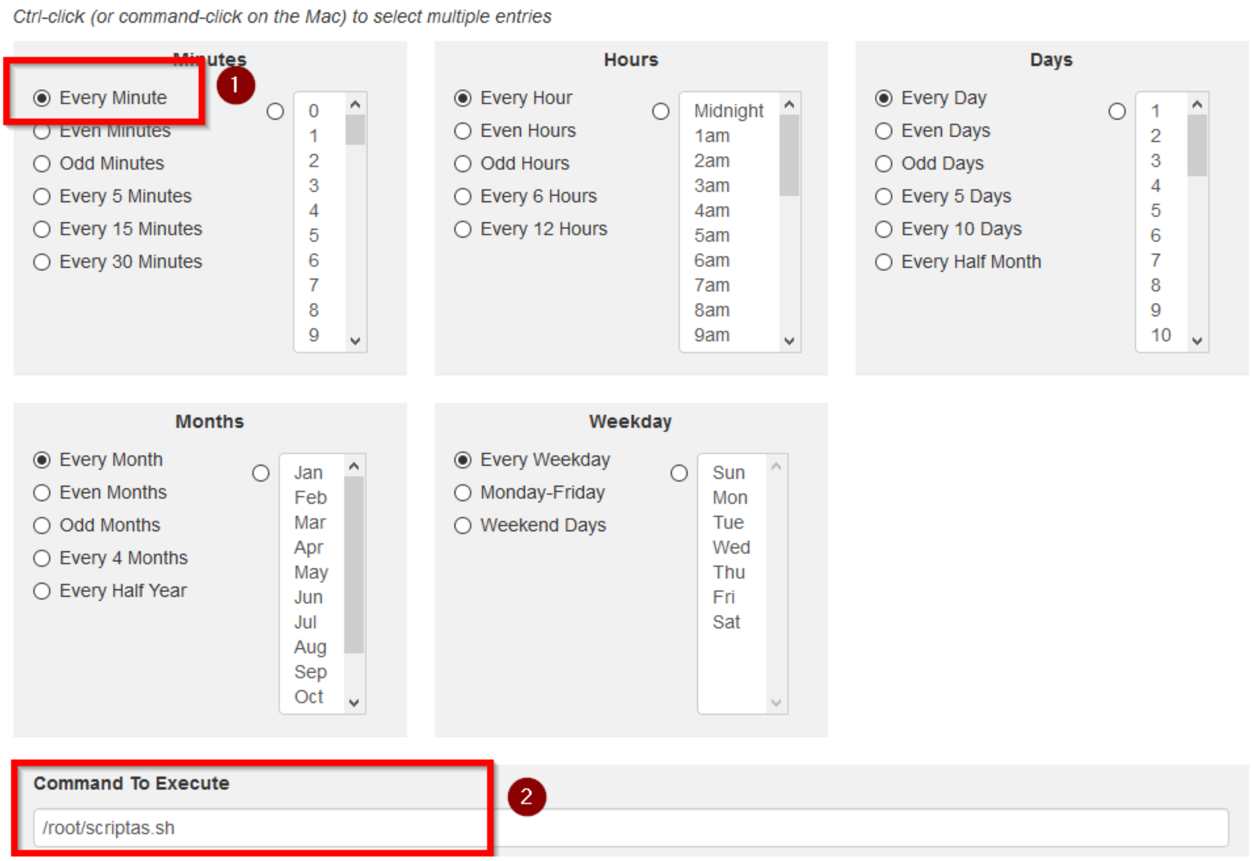
Just writing a script is not enough, because system won‘t allow executing the file, even though it has extension .sh. To make it executable you can use <chmod>.

Syntax for the command is <chmod +x path_to_script>

After you run this command, script should be executable

5. Running scripts automatically using crontab

Running scripts once is fine, but often we want script to run at specific intervals continuosly, we can use crontab to accomplish just that. It could look a bit confusing, but there are simple generators online which you can use to generate crontab commands, I use this one: https://crontab-generator.org/



Ctrl-click (or command-click on the Mac) to select multiple entries

Minutes: ● Every Minute

Hours: ● Every Hour

Days: ● Every Day

Months: ● Every Month

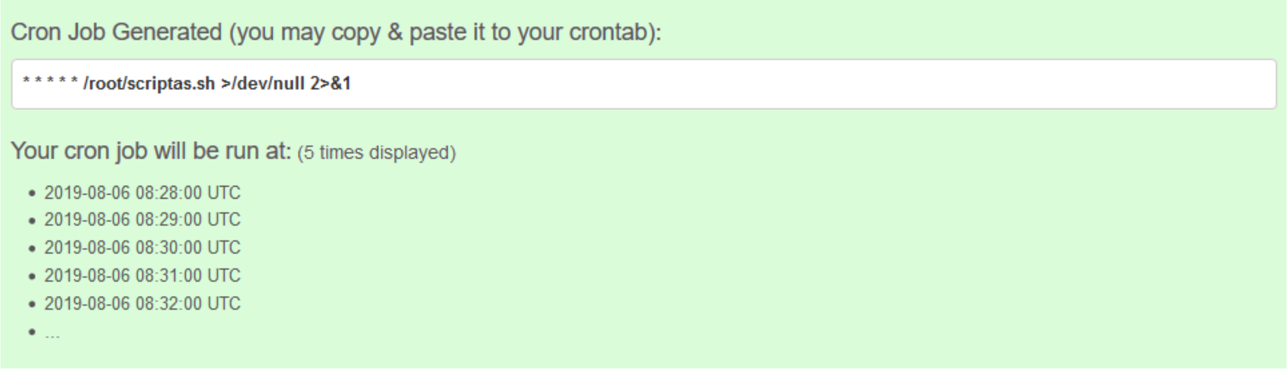
Weekday: ● Every Weekday

Command To Execute

/root/scriptas.sh

With this configuration, the script will run every 1 minute.

You can copy the generated command from here:



Cron Job Generated (you may copy & paste it to your crontab):

```
* * * * * /root/scriptas.sh >/dev/null 2>&1
```

Your cron job will be run at: (5 times displayed)

- 2019-08-06 08:28:00 UTC
- 2019-08-06 08:29:00 UTC
- 2019-08-06 08:30:00 UTC
- 2019-08-06 08:31:00 UTC
- 2019-08-06 08:32:00 UTC
- ...

In the terminal:

- Run the command: <crontab -e>
- Press Insert on your keyboard to enter edit mode
- Paste the command from crontab generator
- Press ESC on your keyboard to exit editor mode
- Press :wq to save and quit

Now the script should run every minute and change SSID by one increment.