

How to use HotPlug for SES and Reset buttons

On the WRT54G, GL, and GS models there are both a Reset and a Secure Easy Setup (SES) button. The SES button is behind the Cisco/Linksys logo on the left front panel while the reset button is recessed into the back side between the WAN jack and the nearest antenna. Actually, WRT54G versions 1.0 up to version 2.2 are excluded because they do not have the SES button.

What we're doing is having shell scripts executed whenever a given event occurs, such as pressing the SES or Reset buttons.

To do this we first remove or comment out the following lines from the file `"/etc/hotplug2-init.rules`

```
SUBSYSTEM ~~ button { exec kill -USR1 1 ; }
```

These lines are not needed and may interfere with our scripts.

Next, create the directory `"/etc/hotplug.d/button"`. This is where our button scripts will go.

Now, `cd /etc/hotplug.d/button` and edit the file that we'll call `"00-button-log"`. This will be a sample script showing off some of the variables that can be used by our button scripts. Put the following into said file:

```
#!/bin/sh logger "($0 $1) $BUTTON-- $ACTION -- $SEEN"
```

After saving the file be sure to make it executable with the command:

chmod +x 00-button-log.

1. The first line simply tells Linux to have the shell (`/bin/sh`) interpret the lines that follow.
2. `"logger"` is a program that stores its arguments into the system log. Use the program `"log read`
`"` to see its contents.

3. `$BUTTON` is replaced by the button name -- either 'ses' or 'reset'. You should test this variable to determine which button was pressed.

4. `$ACTION` is replaced by the string 'pressed' or 'released'.

5. `$SEEN` is replaced by the number of seconds since the last event. When used with

WRT54G: How to use HotPlug for SES and Reset buttons

Written by Rusty Haddock/AE5AE - Last Updated Friday, 20 March 2015 00:51

\$ACTION == 'released' it will tell you how long the button was pressed. This is useful for detecting short presses, long presses, and *REALLY LONG* presses.

Be aware that the buttons are mechanical devices activating an electrical circuit and thus are subject to bounce. This means that multiple 'pressed' or 'released' events are possible with just a single press or release. Be prepared to deal with this in your scripts.

Save the file and we'll see what happens when a button is pressed. We do that by running **logread -f**

". The -f allows us to keep reading the tail end of the file displaying all new log records. Press CTRL-C to stop.

```
root@AE5AE-Base6:/etc/hotplug.d/button# logread -f
Apr  3 14:35:09 AE5AE-Base6
user.notice root: (/sbin/hotplug-call - button - - ) ses-- pressed -- 185
Apr  3 14:35:09
AE5AE-Base6 user.notice root: (/sbin/hotplug-call - button - - ) ses-- pressed -- 0
Apr  3
14:35:12 AE5AE-Base6 user.notice root: (/sbin/hotplug-call - button - - ) ses-- released -- 3
```

See the multiple presses? And sometimes we get lucky and see only one release. Notice that I held the SES button for about 3-seconds before releasing it.

Let's move on to a new file. We'll call it **"20-toggle-flash-white"**. Here are the contents:

```
#!/bin/sh
if [ "$BUTTON" = "ses" ] && [ "$ACTION" = "released" ] && [ $SEEN -gt 0 ] && [
$SEEN -lt 5 ] then
    flash=$(cat /proc/diag/led/ses_white)
    logger "ses_white was $flash"
    if [ "$flash" = "f" ] then
        echo 0 > /proc/diag/led/ses_white
    else
        echo f >
/proc/diag/led/ses_white
    fi
fi
```

Don't forget to run:

chmod +x 20-toggle-flash-white

This script will toggle the white LED behind the SES button between flashing and a steady off. Using **echo** to write an 'f' into the ses_white file will cause that particular LED to flash on its own *ad infinitum*. That capability is in the device driver for the LEDs so there's no need to write a program that keeps the LED flashing. Writing a zero to that file will turn off the LED. You can read the file to see what the LED's current state is; that's what the **cat /proc/diag/led/ses_white** does.

WRT54G: How to use HotPlug for SES and Reset buttons

Written by Rusty Haddock/AE5AE - Last Updated Friday, 20 March 2015 00:51

If your router doesn't have the white LED then replace 'ses_white' with 'dmz'. This will cause the DMZ LED flash.

Make sure to test the shell variable \$SEEN for being greater than zero otherwise the bounces will trigger the flashing.

When you do a "**logread -f**" to see the logged messages, notice that the previous script is outputting messages as well. Yup, all scripts in the 'button' directory will be run on each button event. It's best to 'rm' the files you don't need.