

Home security with Raspberry Pi, not only with Zoneminder

As you probably already know you can have a cheap monitoring system with cameras and email notifications using Raspberry Pi and Zoneminder. Although I have managed to use Zoneminder on a RPI 1 B+ with two USB cameras (I had used ZM for a few years before) I still have the feeling that Zoneminder is just a bit too «heavy» for SoC computers - enterprise quality database and enterprise web server for quite a dumb task? So I have begun to downsize my thinking, which eventually will lead to Arduino, better reliability (no SD card!) and power consumption at levels enabling long operation without access to power grid.

But today let us start with **Zoneminder**. There are many tutorials which can easily be found with Google. Let me add mine – the steps below lead to a working Zoneminder on RPI1 with Raspbian Jessie (I will omit 'sudo' at the beginning of commands, you'd rather add it for most of them)¹:

1. change the default password for 'pi' user with `passwd`
2. if you don't like cables or another keyboard on your desk, enable SSH by running `raspi-config` and navigating to *Interfacing Options / ssh / enable* and put RPI aside where it doesn't hamper you.
3. the you maybe don't need X server that consumes resources – use the same `raspi-config` and *Boot Options / B1 Desktop / CLI / B1 Console*. While being there - check the time zone!
4. `apt-get update`
5. it works without `apt-get dist-upgrade` but you know ...
6. `apt-get install php5 mysql-server php5-mysql` (works without explicit `php-pear` here; you will be asked to submit a password to the database, not needed further in this tutorial)
7. `apt-get install zoneminder`
8. `adduser www-data video` (you can see the effect in `/etc/group` – look for a line beginning with `video:x:`)
9. `ln -s /etc/zm/apache.conf /etc/apache2/conf-enabled/zoneminder.conf`
10. `a2enmod cgi` (without it you could get errors like `socket_sendto(/tmp/zm/zms-704658s.sock) failed: No such file or /directory//usr/share/zoneminder/includes/functions.php 2344`)
11. `a2enconf javascript-common` (it did not need to run it on Raspbian but before I had troubles with buttons on web console while running Ubuntu on a PC, just in case)
12. there might still be a bug in your version of ZM – go to `/usr/share/perl5/ZoneMinder/Memory.pm` and around line 130 change `$arch = int(3.2*length(~0));` to `$arch = 32;`
13. `echo "kernel.shmmax = 134217728" >> /etc/sysctl.conf`
`echo "kernel.shmall = 2097152" >> /etc/sysctl.conf`
14. `/etc/init.d/apache2 restart` (or `service apache2 restart`; you could now be able to see Zoneminder at `http://localhost/zm`)
15. `apt-get install postfix` (you will be asked to chose for example 'local only')
16. go to `/etc/postfix` and create file `sasl_passwd` and put one line into it :
`[smtp.gmail.com]:587 john.smith:pass1234`
Replace user name and password with your credentials; you have noticed that is a line for a Google Mail account. This account is used to send alarm notifications (sent-from). The address of the recipient shall be configured soon in Zoneminder.
17. `postmap hash:/etc/postfix/sasl_passwd`
18. `rm /etc/postfix/sasl_passwd`

¹ See <https://wiki.zoneminder.com/Raspbian> for an explanation of commands

19. replace the content of `/etc/postfix/main.cf` with following lines (you might want to adjust hostname):

```
smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no
append_dot_mydomain = no
readme_directory = no
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache
smtp_tls_security_level = may
smtp_use_tls = yes
smtp_tls_CAfile = /etc/ssl/certs/ca-certificates.crt
myhostname = raspberrypi
myorigin = $myhostname
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
mydestination = raspberrypi, localhost.localdomain, localhost
relayhost = [smtp.gmail.com]:587
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
smtp_sasl_auth_enable = yes
smtp_sasl_password_maps = hash:/etc/postfix/sasl_passwd
smtp_sasl_security_options =
smtp_sasl_tls_security_options = noanonymous
```

20. `/etc/init.d/postfix restart`

21. you might test the configuration of Postfix by `sendmail some.name@some.address <ENTER>`
`test content <ENTER> . <ENTER>`

22. visit your local Zoneminder console `http://localhost/zm` and configure a monitor – your camera might most probably be like mine:

Monitor - Monitor-1 (1) Probe Presets

General	Source	Timestamp	Buffers	Misc
Device Path	/dev/video0			
Capture Method	Video For Linux version 2			
Device Channel	0			
Device Format	PAL			
Capture Palette	*YUYV			
Target Colorspace	24 bit colour			
Capture Width (pixels)	320			
Capture Height (pixels)	240			
Preserve Aspect Ratio	<input type="checkbox"/>			
Orientation	Normal			
Deinterlacing	Disabled			

Fig. 1. Monitor configuration

23. Maximum FPS in the monitor's *General* tab, reduce number of frame in the *Buffers* tab too
24. Navigate to Options – *Email* tab. Mark OPT_EMAIL, write the email address alarms should be sent to. EMAIL_HOST will be *localhost*, the Postfix we configured.
25. if you want pictures to be attached to emails, supplement MESSAGE_BODY with %EI1% and %EIM%, or maybe %ET% for the time of events.
26. Configure a filter: → Filters → Total score, grater than 20, Email details of all matches → Save "Alarm", run filter in background.

Event Filter Close

Use Filter Alarm* [background]

Total Score greater than 20 + -

Sort by Date/Time Asc Limit to first results only

Archive all matches

Email details of all matches

Execute command on all matches

Delete all matches

Submit Execute Save Delete Reset

Fig. 2. Filter configuration

Without Zoneminder

Maybe I am not the only one who likes simplicity, and image-based motion recognition might cause troubles. So I decided to build an alternative with two cameras connected via USB, two PIR sensors (Fig. 1), a RFID reader (Fig. 2) to disarm the system easily, and a simple switch to arm it even easier, plus two LEDs for status reporting. This should run like a charm on low-clocked boards and it is way easier to activate/deactivate than using a browser while leaving your apartment.



Fig. 3. RDM630 RFID reader



Fig. 4. HC-SR501 movement sensor

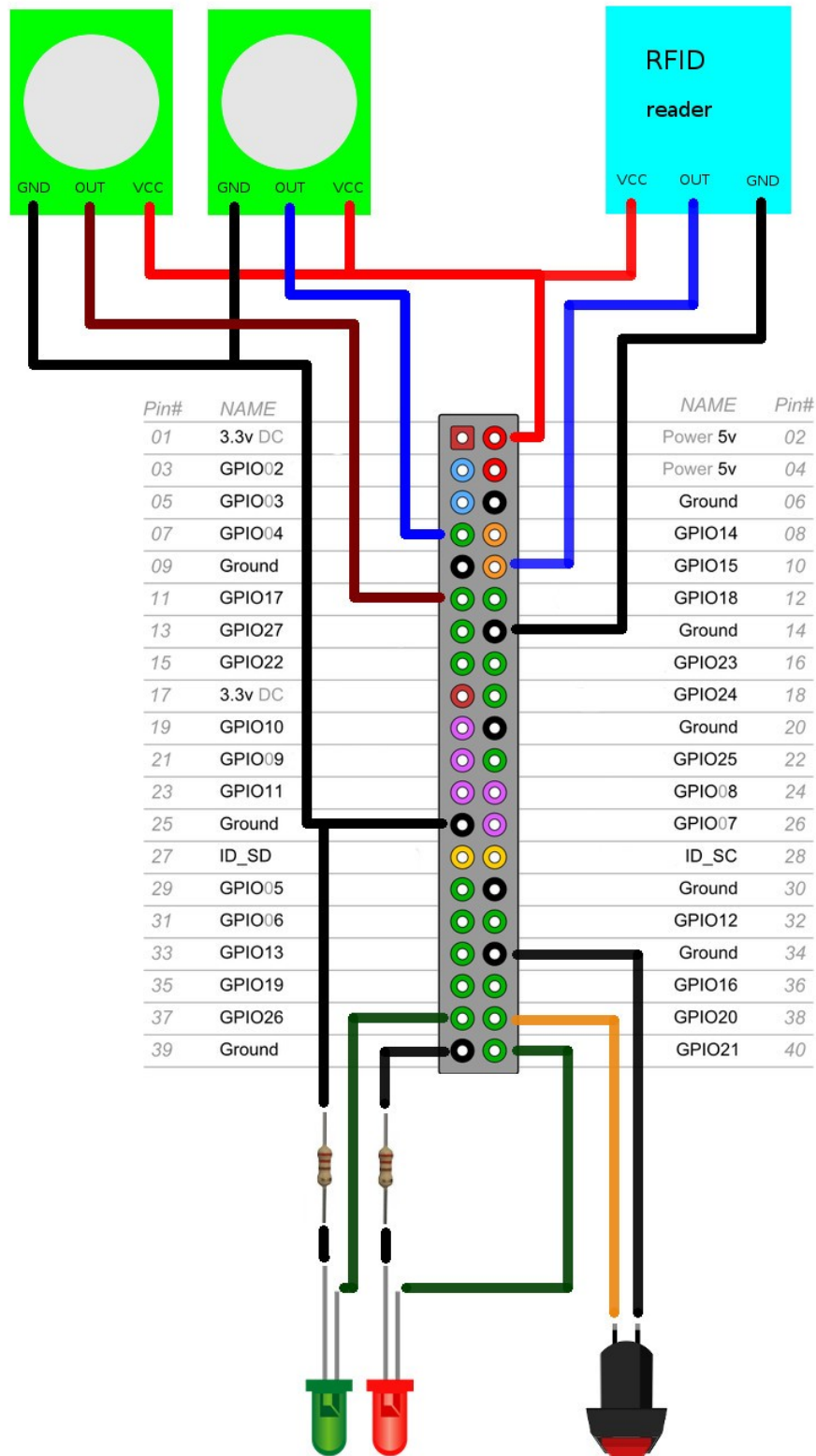


Fig. 5. Pinout of the security system (don't forget to plugin USB cameras!)

The only box I had to put it all together except RPI was not very satisfactory for my wife but not too bad I guess (see fig. 6.), and I managed to wire all this stuff using one single UTP cable (6 signal cables + VCC + GND), use of an RGB led does not change the wiring/schema.

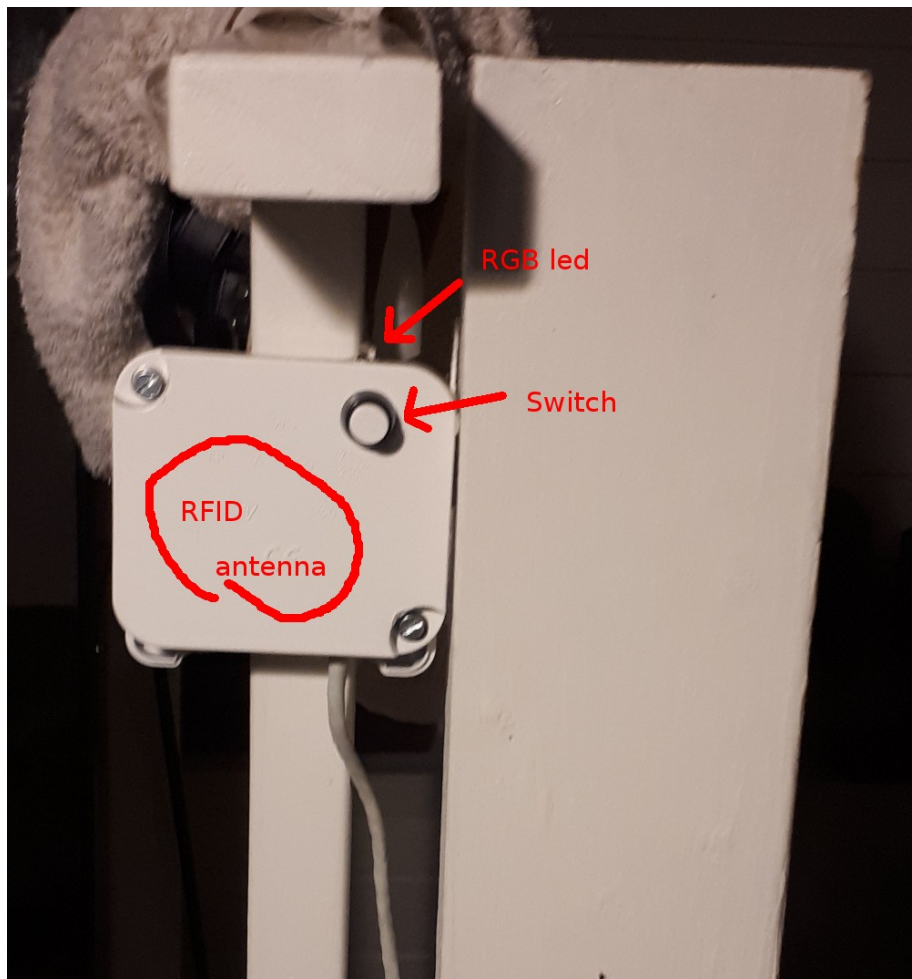


Fig. 6. My control panel

All this takes less than hundred lines in Python to run. There are some prerequisites:

1. install Postfix, see 15-20 above
2. `apt-get install python-setuptools`
3. `easy_install pip`
4. `pip install pyserial`
5. `apt-get install streamer`
6. `apt-get install mailutils`
7. disable serial being used by console logging. I found a few different ways:
 - a) raspi-config → Interfacing Options → Serial → Login shell NOT accessible over serial
 - b) removing `console=serial0,115200` from file `/boot/cmdline.txt`
 - c) `systemctl stop serial-getty@ttyAMA0.service`
`systemctl disable serial-getty@ttyAMA0.service`

A few comments to the source code, or just hints for you to write your own:

- LEDs and PIR sensors are configured by standard `GPIO.setup GPIO.OUT` and `GPIO.IN` respectively
- for that wiring of switch you need `GPIO.setup (?, GPIO.IN, pull_up_down=GPIO.PUD_UP)`
- the RFID reader is connected to GPIO15 which is board's RX, this can be read with `ser = serial.Serial('/dev/ttyAMA0', 9600, timeout=0.1)` and `ser.read(12)`
 This works on Raspbian Jessie on RPI 1, but it might be changed to `/dev/serial0` with other distributions.
- maybe there should be a resistor for the switch too
- I use streamer to dump images from USB cameras:

```
streamer -c /dev/video0 -s 640x480 -o camdmp.jpeg  
and streamer -c /dev/video1 -s 640x480 -o camdmp2.jpeg for the second camera  
- write some alarming wake-up text into alarmmsg.txt file and send email with:  
mail -s "Alarm" -t john.smith@gmail.com -A camdmp.jpeg -A camdmp2.jpeg < alarmmsg.txt
```

Have fun!

Bogusz, Moss, Norway, 10-09-2017