

# MySensors

What you can do with MySensors is described [here](#), and a detailed guide on how to build a MySensors gateway with a Raspberry Pi and a single radio module is [here](#).

With the d-diot platform you have two radio modules, 1 x NRF24I01 (2.4 Ghz) and 1 x RFM69 (433, 868 or 915 Mhz), connected to the GPIO of the Raspberry Pi and two independent gateways instances that runs simultaneously. This setup has been inspired by [this post](#) on the MySensors forum.

The main advantage of the dual radio setup is that you can use both type of sensors and actuators, on the basis of your needs. [Here](#) is an interesting discussion about the pros and cons of each radio type.

## d-diot images v.1.3 and below

### Source code

Build the directory tree and download source code. Due to this [bug](#) that, at the time of writing this guide, affects the signing features in the master branch, compile the development branch.

```
pi@d-diot:~ $ cd ~
pi@d-diot:~ $ mkdir MySensors
pi@d-diot:~ $ mkdir MySensors/source
pi@d-diot:~ $ mkdir MySensors/source/master
pi@d-diot:~ $ mkdir MySensors/source/dev
pi@d-diot:~ $ mkdir MySensors/nrf24
pi@d-diot:~ $ mkdir MySensors/rfm69
pi@d-diot:~ $ cd MySensors/source/dev
pi@d-diot:/home/pi/MySensors/source/dev $ git clone
https://github.com/mysensors/MySensors.git --branch development
pi@d-diot:~ $ cd ~
pi@d-diot:~ $ cd MySensors/source/master
pi@d-diot:/home/pi/MySensors/source/master $ git clone
https://github.com/mysensors/MySensors.git --branch master
```

### d-diot images below 1.3

In the d-diot images below version 1.3 the development branch is used and the compiled version of the MySensors gateway is compatible only with a Raspberry Pi 3.

```
pi@d-diot:~ $ cd /home/pi/MySensors/source/dev
pi@d-diot:/home/pi/MySensors/source/dev $ cp -a MySensors/.
/home/pi/MySensors/nrf24/
pi@d-diot:/home/pi/MySensors/source/dev $ cp -a MySensors/.
/home/pi/MySensors/rfm69/
```

## d-diot image v.1.3

In the d-diot images above or equal to version 1.3 the master branch is used. The default MySensors gateways are compiled for the Raspberry Pi 4, but the binary for the Raspberry pi 3 are stored in a separated folder.

```
pi@d-diot:~ $ cd /home/pi/MySensors/source/master
pi@d-diot:/home/pi/MySensors/source/master $ cp -a MySensors/.
/home/pi/MySensors/nrf24/
pi@d-diot:/home/pi/MySensors/source/master $ cp -a MySensors/.
/home/pi/MySensors/rfm69/
pi@d-diot:/home/pi/MySensors/source/master $ cp -a MySensors/.
/home/pi/MySensors/nrf24-pi3/
pi@d-diot:/home/pi/MySensors/source/master $ cp -a MySensors/.
/home/pi/MySensors/rfm69-pi3/
```

If you have a Pi 3, you need to switch the binaries used by the d-diot images with the following commands:

```
pi@d-diot:~ $ mv /home/pi/MySensors/nrf24/ /home/pi/MySensors/nrf24-pi4/
pi@d-diot:~ $ mv /home/pi/MySensors/nrf24-pi3/ /home/pi/MySensors/nrf24/
pi@d-diot:~ $ mv /home/pi/MySensors/rfm69/ /home/pi/MySensors/rfm69-pi4/
pi@d-diot:~ $ mv /home/pi/MySensors/rfm69-pi3/ /home/pi/MySensors/rfm69/
```

## RFM69 gateway (SPI0, port 5003)

To configure the MySensors ethernet gateway for the RFM69 (868 Mhz) radio module, run the following commands:

```
pi@d-diot:~ $ cd /home/pi/MySensors/rfm69
pi@d-diot:/home/pi/MySensors/rfm69 $ ./configure --my-gateway=ethernet --my-
port=5003 --my-transport=rfm69 --my-rfm69-frequency=868 --my-is-rfm69hw --
spi-driver=SPIDEV --spi-spidev-device=/dev/spidev0.0 --my-leds-err-pin=7 --
my-leds-rx-pin=11 --my-leds-tx-pin=13 --my-signing=software --my-signing-
debug --my-signing-request-signatures --my-signing-weak_security --my-
config-file=/etc/mysensors-rfm69.conf
```

If you have a RFM69 radio module with a different frequency (433 or 915 Mhz), change the parameter “-my-rfm69-frequency=868” accordingly.

Please note that the gateway is compiled with the flag “-my-signing-weak\_security”, so you can use both, signed and unsigned nodes. If you want also to encrypt the communication between the nodes and the gateway add “-my-rfm69-encryption-enabled”, but this requires that all node uses an encrypted connection.

A more detailed discussion on how to secure your MySensors network is [here](#), and the list of the configure parameters is [here](#).

Compile the source code:

```
pi@d-diot:/home/pi/MySensors/rfm69 $ make
```

If something goes wrong during the compilation, you can revert back with:

```
pi@d-diot:/home/pi/MySensors/rfm69 $ make cleanconfig
pi@d-diot:/home/pi/MySensors/rfm69 $ make clean
```

Rename the binary file

```
pi@d-diot:/home/pi/MySensors/rfm69 $ cd /home/pi/MySensors/rfm69/bin/
pi@d-diot:/home/pi/MySensors/rfm69/bin $ mv msgw msgw-rfm69
pi@d-diot:/home/pi/MySensors/rfm69/bin $ cd ..
```

Start the gateway once to generate the config file.

```
pi@d-diot:/home/pi/MySensors/rfm69 $ sudo ./bin/msgw-rfm69
```

Press CTRL+c to terminate the execution and edit the config file.

```
pi@d-diot:/home/pi/MySensors/rfm69 $ sudo nano /etc/mysensors-rfm69.conf
```

This is how the config file should be.

[/etc/mysensors-rfm69.conf](#)

```
# Logging
# Verbosity: debug,info,notice,warn,err
verbose=debug

# Enable logging to a file.
log_file=0
# Log file path.
log_filepath=/tmp/msgw-rfm69.log

# Enable logging to a named pipe.
# Use this option to view your gateway's log messages
# from the log_pipe_file defined bellow.
# To do so, run the following command on another terminal:
#   cat "log_pipe_file"
log_pipe=0
log_pipe_file=/tmp/msgw-rfm69.pipe

# Enable logging to syslog.
syslog=0

# EEPROM settings
eeprom_file=/etc/mysensors-rfm69.eeprom
eeprom_size=1024
```

```
# Software signing settings
# Note: The gateway must have been built with signing
#       support to use the options below.
#
# To generate a HMAC key run mysgw with: --gen-soft-hmac-key
# copy the new key in the line below and uncomment it.
soft_hmac_key=AD5FEE012A7C793950558BA97E974F5B85476584ED1E4AC77FDB9FB0D
E72A04F
# To generate a serial key run mysgw with: --gen-soft-serial-key
# copy the new key in the line below and uncomment it.
soft_serial_key=C2B57D1215706E0F91

# Encryption settings
# Note: The gateway must have been built with encryption
#       support to use the options below.
#
# To generate a AES key run mysgw with: --gen-aes-key
# copy the new key in the line below and uncomment it.
aes_key=26548DFDCF82E9DF33E7500E629D9C40
```

See the “# Software signing settings” section at the end of the config file: three default keys are present but you have to [generate your own keys](#) to avoid security issues.

Test the gateway. [Here](#) more information on how to read the output messages.

```
pi@d-diot:/home/pi/MySensors/rfm69 $ sudo ./bin/mysgw-rfm69
```

Press CTRL+c to terminate the execution.

## NRF24 gateway (SPI1, port 5004)

To configure the MySensors ethernet gateway for the NRF24I01 (2.4 Ghz) radio module, run the following commands:

```
pi@d-diot:~ $ cd /home/pi/MySensors/nrf24
pi@d-diot:/home/pi/MySensors/nrf24 $ ./configure --my-gateway=ethernet --my-
port=5004 --my-transport=rf24 --my-rf24-pa-level=RF24_PA_MAX --spi-
driver=SPIDEV --spi-spidev-device=/dev/spidev1.0 --my-rf24-irq-pin=33 --my-
rf24-ce-pin=37 --my-rf24-cs-pin=36 --my-leds-err-pin=12 --my-leds-rx-pin=16
--my-leds-tx-pin=18 --my-signing=software --my-signing-debug --my-signing-
request-signatures --my-signing-weak_security --my-config-
file=/etc/mysensors-nrf24.conf
```

Please note that the gateway is compiled with the flag “-my-signing-weak\_security”, so you can use both, signed and unsigned nodes. If you want also to encrypt the communication between the nodes

and the gateway add “-my-rf24-encryption-enabled”, but this requires that all node uses an encrypted connection.

A more detailed discussion on how to secure your MySensors network is [here](#), and the list of the configure parameters is [here](#).

Compile the source code:

```
pi@d-diot:/home/pi/MySensors/nrf24 $ make
```

If something goes wrong during the compilation, you can revert back with:

```
pi@d-diot:/home/pi/MySensors/nrf24 $ make cleanconfig
pi@d-diot:/home/pi/MySensors/nrf24 $ make clean
```

Rename the binary file

```
pi@d-diot:/home/pi/MySensors/nrf24 $ cd /home/pi/MySensors/nrf24/bin/
pi@d-diot:/home/pi/MySensors/nrf24/bin $ mv msgw msgw-nrf24
pi@d-diot:/home/pi/MySensors/nrf24/bin $ cd ..
```

Start the gateway once to generate the config file.

```
pi@d-diot:/home/pi/MySensors/nrf24 $ sudo ./bin/msgw-nrf24
```

Press CTRL+c to terminate the execution and edit the config file.

```
pi@d-diot:/home/pi/MySensors/nrf24 $ sudo nano /etc/mysensors-nrf24.conf
```

This is how the config file should be.

[/etc/mysensors-nrf24.conf](#)

```
# Logging
# Verbosity: debug,info,notice,warn,err
verbose=debug

# Enable logging to a file.
log_file=0
# Log file path.
log_filepath=/tmp/msgw-nrf24.log

# Enable logging to a named pipe.
# Use this option to view your gateway's log messages
# from the log_pipe_file defined bellow.
# To do so, run the following command on another terminal:
#   cat "log_pipe_file"
log_pipe=0
log_pipe_file=/tmp/msgw-nrf24.pipe
```

```
# Enable logging to syslog.
syslog=0

# EEPROM settings
eeprom_file=/etc/mysensors-nrf24.eeprom
eeprom_size=1024

# Software signing settings
# Note: The gateway must have been built with signing
#       support to use the options below.
#
# To generate a HMAC key run msgw with: --gen-soft-hmac-key
# copy the new key in the line below and uncomment it.
soft_hmac_key=AD5FEE012A7C793950558BA97E974F5B85476584ED1E4AC77FDB9FB0D
E72A04F
# To generate a serial key run msgw with: --gen-soft-serial-key
# copy the new key in the line below and uncomment it.
soft_serial_key=C2B57D1215706E0F91

# Encryption settings
# Note: The gateway must have been built with encryption
#       support to use the options below.
#
# To generate a AES key run msgw with: --gen-aes-key
# copy the new key in the line below and uncomment it.
aes_key=26548DFDCF82E9DF33E7500E629D9C40
```

See the “# Software signing settings” section at the end of the config file: three default keys are present but you have to [generate your own keys](#) to avoid security issues.

Test the gateway. [Here](#) more information on how to read the output messages.

```
pi@d-diot:/home/pi/MySensors/nrf24 $ sudo ./bin/msgw-nrf24
```

Press CTRL+c to terminate the execution.

## Autostart MySensors at boot

Edit the /etc/rc.local file:

```
pi@d-diot:~ $ sudo nano /etc/rc.local
```

Add the following lines before the “exit 0” string.

[/etc/rc.local](#)

```
# Launch MySensors rfm69
/home/pi/MySensors/rfm69/bin/./mysgw-rfm69 &

# Launch Mysensors nrf24
/home/pi/MySensors/nrf24/bin/./mysgw-nrf24 &
```

## Start and stop MySensors

To stop MySensors RFM69 and NRF24 gateways:

```
pi@d-diot:~ $ sudo killall mysgw-rfm69
pi@d-diot:~ $ sudo killall mysgw-nrf24
```

To start MySensors RFM69 and NRF24 gateways:

```
pi@d-diot:~ $ sudo nohup /home/pi/MySensors/rfm69/bin/./mysgw-rfm69
>/dev/null 2>&1 &
pi@d-diot:~ $ sudo nohup /home/pi/MySensors/nrf24/bin/./mysgw-nrf24
>/dev/null 2>&1 &
```

## d-diot images v.1.4 and above

Download source code from master branch.

```
pi@d-diot:~ $ cd ~
pi@d-diot:~ $ git clone https://github.com/mysensors/MySensors.git --branch
master
```

To configure the MySensors ethernet gateway for the RFM69 (868 Mhz) radio module, run the following commands:

```
pi@d-diot:~ $ cd /home/pi/MySensors
pi@d-diot:~/MySensors $ ./configure --my-gateway=ethernet --my-port=5003 --
my-transport=rfm69 --my-rfm69-frequency=868 --my-is-rfm69hw --spi-
driver=SPIDEV --spi-spidev-device=/dev/spidev0.0 --my-leds-err-pin=16 --my-
leds-rx-pin=11 --my-leds-tx-pin=13 --my-signing=software --my-signing-debug
--my-signing-request-signatures --my-signing-weak_security --my-config-
file=/etc/mysensors.conf
```

If you have a RFM69 radio module with a different frequency (433 or 915 Mhz), change the parameter “-my-rfm69-frequency=868” accordingly.

Please note that the gateway is compiled with the flag “-my-signing-weak\_security”, so you can use both, signed and unsigned nodes. If you want also to encrypt the communication between the nodes and the gateway add “-my-rfm69-encryption-enabled”, but this requires that all node uses an

encrypted connection.

A more detailed discussion on how to secure your MySensors network is [here](#), and the list of the configure parameters is [here](#).

Compile the source code:

```
pi@d-diot:~/MySensors $ make
```

If something goes wrong during the compilation, you can revert back with:

```
pi@d-diot:~/MySensors $ make cleanconfig
pi@d-diot:~/MySensors $ make clean
```

Start the gateway once to generate the config file.

```
pi@d-diot:~/MySensors $ sudo ./bin/mysgw
```

Press CTRL+c to terminate the execution and edit the config file.

```
pi@d-diot:~/MySensors $ sudo nano /etc/mysensors.conf
```

This is how the config file should be.

[/etc/mysensors.conf](#)

```
# Logging
# Verbosity: debug,info,notice,warn,err
verbose=debug

# Enable logging to a file.
log_file=0
# Log file path.
log_filepath=/tmp/mysgw.log

# Enable logging to a named pipe.
# Use this option to view your gateway's log messages
# from the log_pipe_file defined bellow.
# To do so, run the following command on another terminal:
#   cat "log_pipe_file"
log_pipe=0
log_pipe_file=/tmp/mysgw.pipe

# Enable logging to syslog.
syslog=0

# EEPROM settings
eeprom_file=/etc/mysensors.eeprom
eeprom_size=1024
```

```
# Software signing settings
# Note: The gateway must have been built with signing
#       support to use the options below.
#
# To generate a HMAC key run msgw with: --gen-soft-hmac-key
# copy the new key in the line below and uncomment it.
soft_hmac_key=AD5FEE012A7C793950558BA97E974F5B85476584ED1E4AC77FDB9FB0D
E72A04F
# To generate a serial key run msgw with: --gen-soft-serial-key
# copy the new key in the line below and uncomment it.
soft_serial_key=C2B57D1215706E0F91

# Encryption settings
# Note: The gateway must have been built with encryption
#       support to use the options below.
#
# To generate a AES key run msgw with: --gen-aes-key
# copy the new key in the line below and uncomment it.
aes_key=26548DFDCF82E9DF33E7500E629D9C40
```

See the “# Software signing settings” section at the end of the config file: three default keys are present but you have to [generate your own keys](#) to avoid security issues.

Test the gateway. [Here](#) more information on how to read the output messages.

```
pi@d-diot:~/MySensors $ sudo ./bin/msgw
```

Press CTRL+c to terminate the execution.

Install the compiled code

```
pi@d-diot:~/MySensors $ sudo make install
```

Press CTRL+c to terminate the execution.

## Autostart MySensors at boot

Enable the system service

```
pi@d-diot:~ $ sudo systemctl enable msgw.service
```

## Start and stop MySensors

To stop MySensors RFM69 gateway:

```
pi@d-diot:~ $ sudo systemctl stop msgw.service
```

To start MySensors RFM69 gateway:

```
pi@d-diot:~ $ sudo systemctl start msgw.service
```

From:  
<https://www.wiki.d-diot.com/> - **d-diot wiki**

Permanent link:  
[https://www.wiki.d-diot.com/system\\_administration/manual\\_installation/3\\_mysensors](https://www.wiki.d-diot.com/system_administration/manual_installation/3_mysensors)

Last update: **2020/07/26 14:33**

