

# Installation

DomotiYii is the webbased client for DomotiGa and is using the [Yii](#) PHP Framework.

This Wiki page describes the steps to install DomotiYii on your system.

The installation consists of the following steps:

- [\[\[Installation#Requirements|Requirements\]\]](#)
- Web Server - [\[\[Installation#Apache|Apache\]\]](#), [\[\[Installation#Lighttpd|Lighttpd\]\]](#) OR [\[\[Installation#nginx|nginx\]\]](#)
- [\[\[Installation#domotiyii|DomotiYii\]\]](#)

## Requirements

DomotiYii has the following requirements:

- DomotiGa 1.0.014 or higher
- JSON-RPC server enabled
- Web Server (installation steps are described below)

Check in DomotiGa if the module is running and which port is configured (9090)

Check in the file `/var/www/domotiyii/protected/config/main.php`

if the port setting for JSON server are the same: 9090

## Web Server

DomotiYii requires a web server to run, the following three web servers have been successfully tested with DomotiYii:

- [\[\[Installation#Apache HTTP Server|Apache\]\]](#)
- [\[\[Installation#Lighttpd|Lighttpd\]\]](#)
- [\[\[Installation#nginx|nginx\]\]](#)

The choice between the three web servers is trivial, because they deliver all the same functionality. Lighttpd and nginx have a lower memory footprint, thus those are recommended for systems with 512MB or less memory.

The installation and configuration below are based on Ubuntu

## Apache HTTP Server



The Apache HTTP Server, commonly referred to as Apache, is a web server application notable for playing a key role in the initial growth of the World Wide Web. In 2009, it became the first web server software to serve more than 100 million websites. Apache is developed and maintained by an open community of developers under the auspices of the Apache Software Foundation. Released under the Apache License, Apache is open-source software.

1. Apache and the required PHP software can be installed with the following command:

## Ubuntu 12.04

```
$ sudo apt-get install apache2 php5 libapache2-mod-php5 php5-mysql
```

## Ubuntu 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10

```
$ sudo apt-get install apache2 php5 libapache2-mod-php5 php5-mysql php5-json
```

## Ubuntu 16.04 / 16.10

```
$ sudo apt-get install apache2 php7.0 libapache2-mod-php7.0 php7.0-mysql
```

*Ubuntu 16.04 and higher uses PHP 7.0 as default version, the previous Ubuntu versions used PHP 5*

2. Enable the required Apache *rewrite* module as follows:

```
$ sudo a2enmod rewrite
```

3. Make Apache more secure, modify the file `/etc/apache2/sites-enabled/000-default` and change or replace the following (it is directly below the *DocumentRoot /var/www* directive).

```
DocumentRoot /var/www
```

```
<Directory /var/www/>  
    Options None  
    AllowOverride None  
    Order allow,deny  
    allow from all  
</Directory>
```

And allow `.htaccess` file to overwrite settings by modify the file `/etc/apache2/sites-enabled/000-default` and add the following:

```
<Directory /var/www/domotiyii/>  
    AllowOverride All  
</Directory>
```

4. Restart Apache to use the *rewrite* module:

```
$ sudo service apache2 restart
```

## Reducing memory usage

Apache is not the most logical choice for [\[\[DomotiGa:Hardware\\_RaspberryPi|Rapsberry Pi\]\]](#) and other mini PC platforms, [\[\[Installation#Lighttpd\]\]](#) or [\[\[Installation#nginx\]\]](#) seems like a more natural choice as they are lighter on memory / CPU. footprint.

However, if you are like me and feel that you have more experience with apache, I would highly recommend that you change the

default settings to lower the memory consumption on your Raspberry Pi.

After installing Apache, see [\[\[DomotiGa:Install\\_Optional\]\]](#):

```
$ sudo vi /etc/apache2/apache2.conf
```

Change to the following values:

```
<IfModule mpm_prefork_module>
  StartServers      1
  MinSpareServers   1
  MaxSpareServers   3
  MaxClients        15
```

## Apache troubleshooting

Not Found

The requested URL /domotiyii/devices/index was not found on this server.  
Apache/2.4.7 (Ubuntu) Server at localhost Port 80

If you get these errors, it means the `mod_rewrite` module is not working.  
This can be caused by several mistakes.

- You forgot to enable it:

```
$ sudo a2enmod rewrite
```

- You forgot to add these lines to the Apache config file for this site, or directory is not matching installed directory:

```
<Directory /var/www/domotiyii/>
  AllowOverride All
</Directory>
```

- You didn't copy the `.htaccess` file over:

```
prompt:/var/www/html/domotiyii$ ls -lia .htaccess
ls: cannot access .htaccess: No such file or directory
```

Create it:

```
$ sudo vi .htaccess
Options +FollowSymLinks
IndexIgnore */*
<IfModule mod_rewrite.c>
RewriteEngine on
```

```
# if a directory or a file exists, use it directly
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
```

```
# otherwise forward it to index.php
RewriteRule . index.php
</IfModule>
```

## Lighttpd



Lighttpd (pronounced "lighty") is an open-source web server more optimized for speed-critical environments than common products while remaining standards-compliant, secure and flexible. It was originally written by Jan Kneschke as a proof-of-concept of the [c10k problem](#) - how to handle 10,000 connections in parallel on one server, but has gained worldwide popularity.

1. Lighttpd and the required PHP software can be installed with the following command:

### Ubuntu 12.04

```
$ sudo apt-get install lighttpd php5-cgi php5-mysql
```

### Ubuntu 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10

```
$ sudo apt-get install alighttpd php5-cgi php5-mysql php5-json
```

### Ubuntu 16.04 / 16.10

```
$ sudo apt-get install lighttpd php7.0-cgi php7.0-mysql php7.0-json
```

2. The default PHP CGI configuration needs to be changed, to make the *pathinfo* compliant to the official specification. Modify the file */etc/php5/cgi/php.ini* and locate the following line and remove the ";" character from it:

```
;cgi.fix_pathinfo=1
```

If this line doesn't exist, add the following line to the end of the file:

```
cgi.fix_pathinfo=1
```

2. Enable the required Lighttpd *fastcgi* module as follows:

```
$ sudo lighttpd-enable-mod fastcgi
```

3. Tell Lighttpd where to find php binary, add the following to the end of the file */etc/lighttpd/lighttpd.conf*:

### Ubuntu 12.04 / 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10

```
fastcgi.server = ( ".php" => ((
    "bin-path" => "/usr/bin/php5-cgi",
    "socket" => "/tmp/php.socket"
)))
```

#### Ubuntu 16.04 / 16.10

```
fastcgi.server = ( ".php" => ((
    "bin-path" => "/usr/bin/php-cgi7.0",
    "socket" => "/tmp/php.socket"
)))
```

4. The Lighttpd *rewrite* module needs to be enabled, this can only be done by modifying the */etc/lighttpd/lighttpd.conf*. Remove in the following section the *"#"* from the line *"mod\_rewrite"*:

```
server.modules = (
    "mod_access",
    "mod_alias",
    "mod_compress",
    "mod_redirect",
    # "mod_rewrite",
)
```

The section should then look as follows:

```
server.modules = (
    "mod_access",
    "mod_alias",
    "mod_compress",
    "mod_redirect",
    "mod_rewrite",
)
```

5. Add the following rewrite rule to *lighttpd.conf*:

**If you want to access DomotiYii via <http://servername/domotiyii>**

```
url.rewrite-if-not-file = (
    "domotiyii/.*?(.*)" => "domotiyii/index.php?$1",
    "domotiyii/." => "domotiyii/index.php"
)
```

*In this case you have *index.php* under */var/www/domotiyii/domotiyii/**

Here the complete *lighttpd.conf*:

```
server.modules = (
    "mod_access",
    "mod_alias",
    "mod_compress",
    "mod_redirect",
```

```

    "mod_rewrite",
)

# Ubuntu 15.10 and lower. Uncomment this if you run this version
#fastcgi.server = ( ".php" => ((
#     "bin-path" => "/usr/bin/php5-cgi",
#     "socket" => "/tmp/php.socket"
#     )))

# Ubuntu 16.04 and up. Comment this, if you run a lower version
fastcgi.server = ( ".php" => ((
    "bin-path" => "/usr/bin/php-cgi7.0",
    "socket" => "/tmp/php.socket"
    )))

url.rewrite-if-not-file = (
    "domotiyii/.*?(.*)" => "domotiyii/index.php?$1",
    "domotiyii/." => "domotiyii/index.php"
)

server.document-root    = "/var/www/domotiyii/"
server.upload-dirs      = ( "/var/cache/lighttpd/uploads" )
server.errorlog         = "/var/log/lighttpd/error.log"
server.pid-file         = "/var/run/lighttpd.pid"
server.username         = "www-data"
server.groupname        = "www-data"
server.port             = 80

index-file.names        = ( "index.php" )
url.access-deny         = ( "~", ".inc" )
static-file.exclude-extensions = ( ".php", ".pl", ".fcgi" )

compress.cache-dir     = "/var/cache/lighttpd/compress/"
compress.filetype      = ( "application/javascript", "text/css", "text/html", "text/plain" )

include_shell "/usr/share/lighttpd/create-mime.assign.pl"
include_shell "/usr/share/lighttpd/include-conf-enabled.pl"

```

If you want to access DomotiYii via <http://servername/>

```

url.rewrite-if-not-file = (
    ".*?(.*)" => "index.php?$1",
    "." => "index.php"
)

```

*In this case you have index.php under /var/www/domotiyii/*

Here the complete lighttpd.conf:

```

server.modules = (
    "mod_access",
    "mod_alias",
    "mod_compress",
    "mod_redirect",
    "mod_rewrite",
)

# Ubuntu 15.10 and lower. Uncomment this if you run this version
#fastcgi.server = ( ".php" => ((

```

```

#         "bin-path" => "/usr/bin/php5-cgi",
#         "socket" => "/tmp/php.socket"
#     )))

# Ubuntu 16.04 and up. Comment this, if you run a lower version
fastcgi.server = ( ".php" => ( (
    "bin-path" => "/usr/bin/php-cgi7.0",
    "socket" => "/tmp/php.socket"
    )))

url.rewrite-if-not-file = (
    ".*\?(.*)" => "index.php?$1",
    "." => "index.php"
)

server.document-root    = "/var/www/domotiyii"
server.upload-dirs      = ( "/var/cache/lighttpd/uploads" )
server.errorlog         = "/var/log/lighttpd/error.log"
server.pid-file         = "/var/run/lighttpd.pid"
server.username         = "www-data"
server.groupname        = "www-data"
server.port             = 80

index-file.names        = ( "index.php" )
url.access-deny         = ( "~", ".inc" )
static-file.exclude-extensions = ( ".php", ".pl", ".fcgi" )

compress.cache-dir     = "/var/cache/lighttpd/compress/"
compress.filetype      = ( "application/javascript", "text/css", "text/html", "text/plain" )

include_shell "/usr/share/lighttpd/create-mime.assign.pl"
include_shell "/usr/share/lighttpd/include-conf-enabled.pl"

```

6. Restart Lighttpd to use the new configuration and module:

```
$ sudo /etc/init.d/lighttpd force-reload
```

## nginx



nginx (pronounced "engine x") is an open source reverse proxy server for HTTP, HTTPS, SMTP, POP3, and IMAP protocols, as well as a load balancer, HTTP cache, and a web server (origin server). The nginx project started with a strong focus on high concurrency, high performance and low memory usage. It is licensed under the 2-clause BSD-like license.

1. nginx and the required PHP software can be installed with the following command:

### Ubuntu 12.04

```
$ sudo apt-get install nginx php5 php5-fpm php5-mysql
```

### Ubuntu 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10

```
$ sudo apt-get install nginx php5 php5-fpm php5-mysql php5-json
```

## Ubuntu 16.04 / 16.10

```
$ sudo apt-get install nginx php7.0 php7.0-fpm php7.0-json php7.0-mysql
```

2. modify the file `/etc/nginx/sites-enabled/default` and replace the complete file with the following:

Select one of the following configurations depending on the URL you want to use.

If you want to open it via <http://servername>

```
server {  
  
    listen 80;  
    # Don't enable the next line on a Raspberry Pi  
    #listen [::]:80 default_server ipv6only=on;  
  
    server_name DomotiGa;  
    root /var/www;  
    set $yii_bootstrap "index.php";  
  
    charset utf-8;  
  
    location / {  
        index index.html index.php;  
        try_files $uri $uri /$yii_bootstrap?$args;  
  
        location ~ ^/(protected|framework|themes/w+/views) {  
            deny all;  
        }  
  
        #avoid processing of calls to unexisting static files by yii  
        location ~ \.(js|css|png|jpg|gif|swf|ico|pdf|mov|fla|zip|rar|txt)$ {  
            try_files $uri =404;  
            #For production servers you can set expires header with a long period  
            #expires 3M;  
        }  
  
        # pass the PHP scripts to FastCGI server listening on unix:/var/run/php*-fpm.sock  
        #  
        location ~ \.php {  
            include fastcgi_params;  
  
            # *****  
            # Uncomment php5 or php7.0, depending your Ubuntu version  
            # Ubuntu 15.10 and earlier use php5  
            # Ubuntu 16.04 and up use php7.0 (default)  
            # *****  
  
            # Ubuntu 15.10 and earlier  
            #fastcgi_pass unix:/var/run/php5-fpm.sock;  
  
            # Ubuntu 16.04 and up  
            fastcgi_pass unix:/run/php/php7.0-fpm.sock;  
  
            # *****  
  
            #let yii catch the calls to unexisting PHP files  
            set $fsn /$yii_bootstrap;  
            if (-f $document_root$fastcgi_script_name){  
                set $fsn $fastcgi_script_name;  
            }  
        }  
    }  
}
```







*This step can be skipped on 14.04 and higher!*

If you have installed a recent version (on Ubuntu 14.04 for example) or got an upgrade of php5-fpm to v5.4.34 or later there is already a configuration here `/etc/php5/fpm/pool.d/www.conf`

It's similar to the one below, so you can skip step 3 or if you have `/etc/php5/fpm/pool.d/phpfpm.conf` already delete it.

Create the file `/etc/php5/fpm/pool.d/phpfpm.conf` and insert the following content:

```
[phpfpm]
```

```
user = www-data
group = www-data
```

```
listen = /var/run/php5-fpm.sock
listen.owner = www-data
listen.group = www-data
```

```
pm = dynamic
pm.max_children = 5
pm.start_servers = 2
pm.min_spare_servers = 1
pm.max_spare_servers = 3
chdir = /var/www
```

```
; The following php_admin_* settings below, override php.ini settings
;php_admin_value[error_log] = /var/log/fpm-php-error.log
```

```
; The following settings are specifically used on development sites
; Should review/disable for production site
;php_admin_value[max_execution_time] = 180
;php_admin_flag[log_errors] = on
;php_admin_value[memory_limit] = 50m
;php_admin_value[error_reporting] = E_ALL
;php_admin_flag[display_errors] = on
;php_admin_flag[display_startup_errors] = on
```

See `/etc/php5/fpm/pool.d/www.conf` on the meaning of the keys above.

#### 4. Configure PHP

Edit `php.ini` file.

**Ubuntu 12.04 / 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10**

```
$ sudo vi /etc/php5/fpm/php.ini
```

**Ubuntu 16.04 / 16.10**

```
$ sudo vi /etc/php/7.0/fpm/php.ini
```

Uncomment or add the following lines:

```
extension=mysqli.so
extension=pdo_mysql.so
```

#### 5. Restart nginx:

```
$ sudo service nginx restart
```

6. Restart php-fpm:

**Ubuntu 12.04 / 13.04 / 13.10 / 14.04 / 14.10 / 15.04 / 15.10**

```
$ sudo service php5-fpm restart
```

**Ubuntu 16.04 / 16.10**

```
$ sudo service php7.0-fpm restart
```

*Note:* This failed for me the first time, only after installing DomotiYii as documented below it restarted ok.

### Troubleshooting

- nginx has its configuration files in `/etc/nginx/`
- Logs are in `/var/log/nginx/`, if not check `/etc/nginx/nginx.conf` `access_log` and `error_log` variables

Documentation can be found (apart from using your search engine) on <http://wiki.nginx.org/Main> under the [Configure](#) button and <http://nginx.org/en/docs/>.

### 502 Bad Gateway

If your browser is telling *502 Bad Gateway* and you see in the `error_log`:

```
[crit] 5728#0: *3 connect() to unix:/var/run/php5-fpm.sock failed (13: Permission denied) while connecting to upstream, client:
aaa.bbb.ccc.ddd, server: domotiga, request: "GET /domotiyii/ HTTP/1.1", upstream: "fastcgi://unix:/var/run/php5-fpm.sock:", host:
"aaa.bbb.ccc.eee"
```

Check the permissions of `/var/run/php5-fpm.sock`:

```
$ ls -la /var/run/php5-fpm.sock
0 srw-rw---- 1 root root 0 jun  9 22:59 /var/run/php5-fpm.sock
```

This is not right based on [this thread](#), you can fix this:

- by doing `sudo chown www-data:www-data /var/run/php5-fpm.sock`
- by adding the "listen.user = www-data" and "listen.group = www-data" lines in `/etc/php5/fpm/pool.d/phpfpm.conf`
- by using a TCP/IP connection instead of Unix socket,

### Notes about Performance

On small embedded devices it's better to use Unix socket instead of a TCP-IP connection for the communication between nginx and PHP-FPM.

Make sure the pm worker parameters are not set to high.

## DomotiYii

### Copy files

After you installed the Web Server, DomotiYii can be download and copied to the Servers directory. Execute the following steps for this:

```
$ mkdir ~/install
$ cd ~/install
```

```
$ git clone https://github.com/DomotiGa/DomotiYii.git domotiyii
```

### For <http://serverip>

If you have chosen to access DomotiYii via <http://serverip> do this:

```
$ sudo cp -rp ~/install/domotiyii/domotiyii/* /var/www/  
$ sudo chown -R www-data:www-data /var/www/protected/runtime  
$ sudo chown -R www-data:www-data /var/www/assets
```

If you want to develop and create new code with Gii, change these permissions too.

```
$ sudo chown -R www-data:www-data /var/www/protected/views  
$ sudo chown -R www-data:www-data /var/www/protected/controllers  
$ sudo chown -R www-data:www-data /var/www/protected/models
```

### For <http://serverip/domotiyii>

If you have chosen to access DomotiYii via <http://serverip/domotiyii> do this:

```
$ sudo cp -rp ~/install/domotiyii/domotiyii /var/www/  
$ sudo chown -R www-data:www-data /var/www/domotiyii/protected/runtime  
$ sudo chown -R www-data:www-data /var/www/domotiyii/assets
```

If you want to develop and create new code with Gii, change these permissions too.

```
$ sudo chown -R www-data:www-data /var/www/domotiyii/protected/views  
$ sudo chown -R www-data:www-data /var/www/domotiyii/protected/controllers  
$ sudo chown -R www-data:www-data /var/www/domotiyii/protected/models
```

## Set basePathDomotiGa

To display log files, DomotiYii needs to access the log files.  
Set the correct basePathDomotiGa in domotiyii/protected/config/main.php:

```
// base path to domotiga  
'basePathDomotiGa'=>'/home/<changethis>/domotiga',
```

## Tweak DomotiYii

If you have changed DomotiGa's database name, MySQL credentials, JSON-RPC port and so on, you need to edit the config file which is in 'protected/config' and called 'main.php'. Be careful with this file, **make a backup before modifying it if you don't really know Yii framework** but you can guess easily what to change.

For example, if your database name is now "my-domotiga", you have to change this:

```
'connectionString' => 'mysql:host=localhost;dbname=domotiga',
```

TO

'connectionString' => 'mysql:host=localhost;dbname=my-domotiga',

You can find the other parameters in this file too.

## Use DomotiYii

After the Server and DomotiYii are installed and configured you can start using the DomotiYii client.

Go to <http://serverip/domotiyii> or <http://serverip> depending on chosen directory.

Login credentials are: admin/admin

## RRDTool graphs

If you also want to serve the RRDToolgraphs that are in your /home/user/domotiga/rrd/graphs folder, see [\[\[DomotiGa:RRDTool#Web server configuration|RRDTool Web server configuration\]\]](#).

## Resources

- [Apache HTTP Server](#)
- [Lighttpd](#)
- [nginx](#)

TBD:

- <http://www.thegeekstuff.com/2011/03/apache-hardening/>
- <http://www.techrepublic.com/blog/10things/10-things-you-should-do-to-secure-apache>
- <http://www.yiiframework.com/wiki/153/using-yii-with-nginx-and-php-fpm/>
- <http://www.yiiframework.com/forum/index.php?/topic/6977-removing-index-php-in-url-with-other-webservers/>
- <http://phpaul.wordpress.com/2011/01/13/rewrite-a-url-with-lighttpd-for-yii-framework/>
- <http://redmine.lighttpd.net/projects/1/wiki/DebugVariables>
- <https://help.ubuntu.com/10.04/serverguide/httpd.html>

{{comments}}

{{comment\_form}}

{{count}}

Updated by: {{lastupdated\_by}}, {{lastupdated\_at}}  
Access count: {{show\_count}}

## Attached Files

### Files

apache.png	2.91 KB	02/05/2014	jessekerkhoven
lighttpd.png	7.29 KB	02/05/2014	jessekerkhoven
nginx.gif	524 Bytes	02/05/2014	jessekerkhoven