

# Install and configure Cacti on CentOS

Here we use root login for this installation and configuration. Before doing anything else, we should disable SELinux.

## Step 1 – Disable SELinux

Open and edit SELinux configuration file.

```
vim /etc/sysconfig/selinux
```

Change SELINUX=enforcing to SELINUX=disabled. Save and exit. Reboot system

```
reboot
```

## Step 2 – Enable Repos

Head over to the [Fedora page](#) and copy the latest download link of the latest file.

Setup the EPEL repository:

```
rpm -Uvh https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm
```

Similarly, copy the link of [Remi](#) and install the Remi repository.

```
wget http://rpms.famillecollet.com/enterprise/remi-release-7.rpm
```

Install repository.

```
rpm -Uvh remi-release-7.rpm
```

List repositories.

```
yum repolist
```

## Step 3 – Install Apache

Let's install Apache and start the service.

```
yum install httpd httpd-devel
```

```
systemctl start httpd
```

## Step 4 – Install SNMP and RRDTool

To install SNMP and RRD Tool, enter the following command:

```
yum install net-snmp net-snmp-utils net-snmp-libs rrdtool
```

Accept typing “Y” to confirm installation.

Start SNMP.

```
systemctl start snmpd
```

## Step 5 – Install MariaDB Server

Use the following command to install the MariaDB server and start the service.

```
yum install mariadb-server
```

```
systemctl start mariadb
```

Complete MariaDB Secure Installation.

```
mysql_secure_installation
```

```
[root@localhost ~]# mysql_secure_installation
```

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB  
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current  
password for the root user. If you've just installed MariaDB, and  
you haven't set the root password yet, the password will be blank,  
so you should just press enter here.

Enter current password for root (enter for none):  
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB  
root user without the proper authorisation.

Set root password? [Y/n] Y  
New password:  
Re-enter new password:  
Password updated successfully!  
Reloading privilege tables..  
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone  
to log into MariaDB without having to have a user account created for  
them. This is intended only for testing, and to make the installation  
go a bit smoother. You should remove them before moving into a  
production environment.

Remove anonymous users? [Y/n] Y  
... Success!

Normally, root should only be allowed to connect from 'localhost'. This  
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] Y  
... Success!

By default, MariaDB comes with a database named 'test' that anyone can  
access. This is also intended only for testing, and should be removed  
before moving into a production environment.

Remove test database and access to it? [Y/n] Y  
- Dropping test database...  
... Success!  
- Removing privileges on test database...  
... Success!

Reloading the privilege tables will ensure that all changes made so far  
will take effect immediately.

Reload privilege tables now? [Y/n] Y  
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MariaDB  
installation should now be secure.

Thanks for using MariaDB!

## Step 6 – Install PHP and needed packages

Fire up the following command to install a bunch of PHP related packages.

```
yum --enablerepo=remi install php-mysql php-pear php-common php-gd php-devel php php-mbstring php-cli php-intl php-snmp
```

## Step 7 – Create a Cacti Database

Now you can log in to the database server with the previously configured password.

```
mysql -u root -p
[root@localhost ~]# mysql -u root -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 10
Server version: 5.5.56-MariaDB MariaDB Server

Copyright (c) 2000, 2017, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]>
```

Create a Database and user.

```
MariaDB [(none)]> create database fosslinuxcacti;
```

```
MariaDB [(none)]> CREATE USER 'fosslinuxuser'@'localhost'
IDENTIFIED BY 'C345gDvfr@#231';
```

Grant permission and flush privileges.

```
MariaDB [(none)]> grant all privileges on fosslinuxcacti.* to
fosslinuxuser@localhost ;
```

```
MariaDB [(none)]> FLUSH PRIVILEGES;
```

```
MariaDB [(none)]> create database fosslinuxcacti;
Query OK, 1 row affected (0.00 sec)

MariaDB [(none)]> CREATE USER 'fosslinuxuser'@'localhost' IDENTIFIED BY 'C345gDvfr@#231';
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> grant all privileges on fosslinuxcacti.* to fosslinuxuser@localhost ;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> FLUSH PRIVILEGES;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]>
```

Create database user should have access to the mysql.time\_zone\_name Table. So we should import the mysql\_test\_data\_timezone.sql to MySQL database first.

```
mysql -u root -p mysql <
/usr/share/mysql/mysql_test_data_timezone.sql
Login to MariaDB.
```

```
mysql -u root -p
Grant access to the cacti database user and flush privileges.
```


```
MariaDB [(none)]> GRANT SELECT ON mysql.time_zone_name TO
fossilinuxuser@localhost;
```

```
MariaDB [(none)]> flush privileges;
```

```
MariaDB [(none)]> GRANT SELECT ON mysql.time_zone_name TO fossilinuxuser@localhost;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> flush privileges;
Query OK, 0 rows affected (0.00 sec)

MariaDB [(none)]> █
```



## Step 8 – Optimize Database

We need to modify database parameters for better performance. Use the following command.

```
vim /etc/my.cnf.d/server.cnf
Add the following lines to the [mysqld] section.
```

```
collation-server = utf8_general_ci

init-connect='SET NAMES utf8'

character-set-server = utf8

max_heap_table_size = 128M

max_allowed_packet = 16777216

tmp_table_size = 64M

join_buffer_size = 128M

innodb_file_per_table = on
```

```
innodb_buffer_pool_size = 1024M

innodb_doublewrite = off

innodb_additional_mem_pool_size = 80M

innodb_lock_wait_timeout = 50

innodb_flush_log_at_trx_commit = 2
```

```
#
# These groups are read by MariaDB server.
# Use it for options that only the server (but not clients) should see
#
# See the examples of server my.cnf files in /usr/share/mysql/
#

# this is read by the standalone daemon and embedded servers
[server]


# this is only for the mysqld standalone daemon
[mysqld]
collation-server = utf8_general_ci
init-connect='SET NAMES utf8'
character-set-server = utf8
max_heap_table_size = 128M
max_allowed_packet = 16777216
tmp_table_size = 64M
join_buffer_size = 64M
innodb_file_per_table = on
innodb_buffer_pool_size = 512M
innodb_doublewrite = off
innodb_additional_mem_pool_size = 80M
innodb_lock_wait_timeout = 50
innodb_flush_log_at_trx_commit = 2

# this is only for embedded server
[embedded]

# This group is only read by MariaDB-5.5 servers.
# If you use the same .cnf file for MariaDB of different versions,
# use this group for options that older servers don't understand
[mysqld-5.5]

# These two groups are only read by MariaDB servers, not by MySQL.
# If you use the same .cnf file for MySQL and MariaDB,
# you can put MariaDB-only options here
[mariadb]

[mariadb-5.5]
```



When done, restart the MariaDB service using the restart command:

```
systemctl restart mariadb.service
```

## Step 9 – Install and configure Cacti

Time to install Cacti now using YUM.

```
yum -y install cacti
```

Import default cacti database file to the created database.

```
cd /usr/share/doc/cacti-1.1.38
```

Import the SQL file.

```
mysql -u root -p fosslinuxcacti < cacti.sql
```

Edit the Cacti configuration file which includes a database, password details etc.

```
vim /usr/share/cacti/include/config.php
```

Modify database details.

```
$database_type      = 'mysql';  
  
$database_default  = 'fosslinuxcacti';  
  
$database_hostname = 'localhost';  
  
$database_username = 'fosslinuxuser';  
  
$database_password = 'C345gDvfr@#231';  
  
$database_port     = '3306';  
  
$database_ssl      = false;
```

```

/*
+-----+
| Copyright (C) 2004-2018 The Cacti Group
|
| This program is free software; you can redistribute it and/or
| modify it under the terms of the GNU General Public License
| as published by the Free Software Foundation; either version 2
| of the License, or (at your option) any later version.
|
| This program is distributed in the hope that it will be useful,
| but WITHOUT ANY WARRANTY; without even the implied warranty of
| MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
| GNU General Public License for more details.
+-----+
| Cacti: The Complete RRDtool-based Graphing Solution
+-----+
| This code is designed, written, and maintained by the Cacti Group. See
| about.php and/or the AUTHORS file for specific developer information.
+-----+
| http://www.cacti.net/
+-----+
*/

/* make sure these values reflect your actual database/host/user/password */

$database_type      = 'mysql';
$database_default    = 'fossilinuxcacti';
$database_hostname   = 'localhost';
$database_username   = 'fossilinuxuser';
$database_password   = 'C345gDvfr@#23!';
$database_port       = '3306';
$database_ssl        = false;

/* when the cacti server is a remote poller, then these entries point to
 * the main cacti server. otherwise, these variables have no use.
 * and must remain commented out. */

#$rdatabase_type     = 'mysql';
#$rdatabase_default  = 'cacti';
#$rdatabase_hostname = 'localhost';
#$rdatabase_username = 'cactiuser';
#$rdatabase_password = 'cactiuser';
#$rdatabase_port     = '3306';
#$rdatabase_ssl      = false;

```



## Step 10 – Set Cron for Cacti

Open cacti cron file.

```
vim /etc/cron.d/cacti
```

Uncomment the following line.

```
*/5 * * * *    apache /usr/bin/php /usr/share/cacti/poller.php
> /dev/null 2>&1
```



```
/5 * * * * apache /usr/bin/php /usr/share/cacti/poller.php > /dev/null 2>&1
~
~
```

Save and exit the file.

## Step 11 – Configure Apache for Cacti

This will help us to do a remote installation. Edit the cacti config file.

```
vim /etc/httpd/conf.d/cacti.conf
```

Change “Require host localhost” to “Require all granted” and “Allow from localhost” to “Allow from all.”

```
#
# Cacti: An rrd based graphing tool
#
# For security reasons, the Cacti web interface is accessible only to
# localhost in the default configuration. If you want to allow other clients
# to access your Cacti installation, change the httpd ACLs below.
# For example:
# On httpd 2.4, change "Require host localhost" to "Require all granted".
# On httpd 2.2, change "Allow from localhost" to "Allow from all".

Alias /cacti /usr/share/cacti

<Directory /usr/share/cacti/>
    <IfModule mod_authz_core.c>
        # httpd 2.4
        Require all granted
    </IfModule>
    <IfModule !mod_authz_core.c>
        # httpd 2.2
        Order deny,allow
        Deny from all
        Allow from all
    </IfModule>
</Directory>

<Directory /usr/share/cacti/install>
    # mod_security overrides.
    # Uncomment these if you use mod_security.
    # allow POST of application/x-www-form-urlencoded during install
    #SecRuleRemoveById 960010
    # permit the specification of the rrdtool paths during install
    #SecRuleRemoveById 900011
</Directory>
```

Change Time Zone.

```
vim /etc/php.ini
```

```
date.timezone = your time Zone
```

E.g.:- date.timezone = Australia/Sydney

Restart apache, MariaDB, and SNMP.

```
systemctl restart httpd.service
```

```
systemctl restart mariadb.service
```

```
systemctl restart snmpd.service
```

## Step 12 – Configure the Firewall

Use these commands:

```
firewall-cmd --permanent --zone=public --add-service=http
```

```
firewall-cmd --reload
```

## Step 13 – Start Cacti installation

Open a web browser and use the following URL to access Cacti web interface.

```
http://Your-Server-IP/cacti
```

You will see the Licence Agreement page Then accept it and click Begin.

## Cacti Installation Wizard

### Cacti Version 1.1.38 - License Agreement

Thanks for taking the time to download and install Cacti, the complete graphing solution for your network. Before you can start making cool graphs, there are a few pieces of data that Cacti needs to know.

Make sure you have read and followed the required steps needed to install Cacti before continuing. Install information can be found for [Unix](#) and [Win32](#)-based operating systems.

Cacti is licensed under the GNU General Public License, you must agree to its provisions before continuing:

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

☒ Accept GPL License Agreement

Begin



#### Pre-Installation Checks

##### Location checks

Your Cacti configuration has the relative correct path (url\_path) in config.php.

##### MySQL TimeZone Support

Your Cacti database account has access to the MySQL TimeZone database and that database is populated with global TimeZone information.

##### MySQL Temporary Table Access

Your Database user account can create temporary tables.

##### PHP TimeZone Support

Your Web Servers PHP is properly setup with a TimeZone.

##### Required PHP Module Support

Cacti requires several PHP Modules to be installed to work properly. If any of these are not installed, you will be unable to continue the installation until corrected. In addition, for optimal system performance Cacti should be run with certain MySQL system variables set. Please follow the MySQL recommendations at your discretion. Always seek the MySQL documentation if you have any questions. The following PHP extensions are mandatory, and MUST be installed before continuing your Cacti install.

| Required PHP Modules | Required | Installed |
|----------------------|----------|-----------|
| PHP Version          | 5.4.0+   | 5.4.43    |
| session              | Yes      | Yes       |
| sockets              | Yes      | Yes       |
| PDO                  | Yes      | Yes       |
| pdo_mysql            | Yes      | Yes       |
| xml                  | Yes      | Yes       |
| soap                 | Yes      | Yes       |
| intl                 | Yes      | Yes       |
| json                 | Yes      | Yes       |
| openssl              | Yes      | Yes       |
| gd                   | Yes      | Yes       |
| zlib                 | Yes      | Yes       |
| posix                | Yes      | Yes       |

##### Optional PHP Module Support

The following PHP extensions are recommended, and should be installed before continuing your Cacti install.

| Optional Module | Optional | Installed |
|-----------------|----------|-----------|
| snmp            | Yes      | Yes       |
| gmp             | Yes      | Yes       |
| TrueType Text   | Yes      | Yes       |
| TrueType Box    | Yes      | Yes       |

These MySQL performance tuning settings will help your Cacti system perform better without issues for a longer time.

| Recommended MySQL System Variable Settings  | Current Value   | Recommended Value | Comments  |
|---|-----------------|-------------------|---|
| MySQL Tuning (showing only 1) Documentation (Note: Many changes below require a database restart) |                 |                   |   |
| version   | 5.5.56-MariaDB  | >= 5.6            | MySQL 5.6+ and MariaDB 10.0+ are great releases, and are very good versions to choose. Make sure you run the very latest release though which there is a long standing low level networking issue that was causing some many issues with reliability.   |
| collation_server  | utf8_general_ci | utf8_general_ci   | When using Cacti with languages other than English, it is important to use the utf8_general_ci collation type as some characters take more than a single byte. If you are first just now installing Cacti, stop, make the changes and start over again. If your Cacti has been running and is in production, see the internet for instructions on converting your databases and tables if you plan on supporting other languages.   |
| character_set_client  | utf8            | utf8              | When using Cacti with languages other than English, it is important to use the utf8 character set as some characters take more than a single byte. If you are first just now installing Cacti, stop, make the changes and start over again. If your Cacti has been running and is in production, see the internet for instructions on converting your databases and tables if you plan on supporting other languages.   |
| max_connections   | 251             | >= 300            | Depending on the number of logins and size of your data collector, MariaDB will need many connections. The calculation for mine is: total_connections = total_processors * (total_threads * script_servers + 1), then you must leave headroom for user connections, which will change depending on the number of concurrent login accounts.   |
| max_heap_table_size   | 128M            | >= 80M            | If using the Cacti Performance Booster and choosing a memory storage engine, you have to be careful to flush your Performance Booster buffer before the system runs out of memory table space. This is done two ways, first reducing the size of your output columns to just the right size. This column is in the tables_poller_output, and poller_output_boost. The second thing you can do is allocate more memory to memory tables. We have arbitrarily chosen a recommended value of 10% of system memory, but if you are using SSD disk drives, or have a smaller system, you may ignore this recommendation or choose a different storage engine. You may see the expected consumption of the Performance Booster tables under Console -> System Utilities -> View Boost Status. |
| max_allowed_packet  | 16777216        | >= 16777216       | With Remote polling capabilities, large amounts of data will be spooled from the main server to the remote pollers. Therefore, keep this value at or above 10M.   |
| tmp_table_size  | 64M             | >= 64M            | When executing subqueries, having a larger temporary table size, keep those temporary tables in memory.   |
| join_buffer_size  | 64M             | >= 64M            | When performing joins, if they are below this size, they will be kept in memory and never written to a temporary file.  |
| innodb_file_per_table   | ON              | ON                | When using InnoDB storage it is important to keep your table spaces separate. This makes managing the tables simpler for long time users of MariaDB. If you are running with this currently off, you can migrate to the per file storage by enabling the feature, and then running an alter statement on all InnoDB tables.   |
| innodb_buffer_pool_size   | 512M            | >= 449M           | InnoDB will hold as much tables and indexes in system memory as is possible. Therefore, you should make the innodb_buffer_pool large enough to hold as much of the tables and index in memory. Checking the size of the /var/lib/mysql/data directory will help in determining this value. We are recommending 20% of your systems total memory, but your requirements will vary depending on your systems size.  |
| innodb_doublewrite  | OFF             | OFF               | With modern SSD type storage, this operation actually degrades the disk more rapidly and adds a 50% overhead on all write operations.   |
| innodb_additional_mem_pool_size   | 80M             | >= 80M            | This is where metadata is stored. If you had a lot of tables, it would be useful to increase this.  |
| innodb_lock_wait_timeout  | 50              | >= 30             | Rogue queries should not be for the database to go offline to others. Kill these queries before they kill your system.  |
| innodb_flush_log_at_trx_commit  | 2               | 2                 | Setting this value to 2 means that you will flush all transactions every second rather than at commit. This allows MariaDB to perform writing less often.   |

Previous Next



Next window is Installation Type. It will show Database connection details. Click Next to continue.

## Cacti Installation Wizard

### Installation Type

Please select the type of installation

Installation options:

- **New Primary Server** - Choose this for the Primary site.
- **New Remote Poller** - Remote Pollers are used to access networks that are not readily accessible to the Primary site.

New Primary Server

The following information has been determined from Cacti's configuration file. If it is not correct, please edit "include/config.php" before continuing.

#### Local Cacti database connection information

Database: fosslinuxcacti  
Database User: fosslinuxuser  
Database Hostname: localhost  
Port: 3306  
Server Operating System Type: unix

Previous

Next

## Cacti Installation Wizard

### Critical Binary Locations and Versions

Make sure all of these values are correct before continuing.

**RRDtool Binary Path:** The path to the rrdtool binary.



**PHP Binary Path:** The path to your PHP binary file (may require a php recompile to get this file).



**snmpwalk Binary Path:** The path to your snmpwalk binary.



**snmpget Binary Path:** The path to your snmpget binary.



**snmpbulkwalk Binary Path:** The path to your snmpbulkwalk binary.



**snmpgetnext Binary Path:** The path to your snmpgetnext binary.



**snmptrap Binary Path:** The path to your snmptrap binary.



**Spine Binary File Location:** The path to Spine binary.



**Cacti Log Path:** The path to your Cacti log file.



**Theme:** Please select one of the available Themes to skin your Cacti with.

**RRDtool Version:** The version of RRDtool that you have installed.

Verify Directory Permissions and continue.

Not secure | 10.94.10.206/cacti/install/index.php

### Cacti Installation Wizard

#### Directory Permission Checks

Please ensure the directory permissions below are correct before proceeding. During the install, these directories need to be owned by the Web Server user. These permission changes are required to allow the Installer to install Device Template packages which include XML and script files that will be placed in these directories. If you choose not to install the packages, there is an 'install\_package.php' cli script that can be used from the command line after the install is complete.

After the install is complete, you can make some of these directories read only to increase security.

**Required Writable at Install Time Only**

- /usr/share/cacti/resource/snmp\_queries is **Writable**
- /usr/share/cacti/resource/script\_server is **Writable**
- /usr/share/cacti/resource/script\_queries is **Writable**
- /usr/share/cacti/scripts is **Writable**

**Required Writable after Install Complete**

- /usr/share/cacti/log is **Writable**
- /usr/share/cacti/cache/boost is **Writable**
- /usr/share/cacti/cache/mibcache is **Writable**
- /usr/share/cacti/cache/realtime is **Writable**
- /usr/share/cacti/cache/spikekill is **Writable**

All folders are writable

**NOTE:** If you are installing packages, once the packages are installed, you should change the scripts directory back to read only as this presents some exposure to the web site.

Previous Next

FOSS Linux

In the Template Setup window, you can select all templates, and click finish to the complete installation.

Not secure | 10.94.10.206/cacti/install/index.php

### Cacti Installation Wizard

#### Template Setup

Please select the Device Templates that you wish to use after the Install. If your Operating System is Windows, you need to ensure that you select the 'Windows Device' Template. If your Operating System is Linux/UNIX, make sure you select the 'Local Linux Machine' Device Template. Device Templates allow you to monitor and graph a vast assortment of data within Cacti. After you select the desired Device Templates, press 'Finish' and the installation will complete. Please be patient on this step, as the importation of the Device Templates can take a few minutes.

| Name                | Description                     | Author          | Homepage             |                                     |
|---------------------|---------------------------------|-----------------|----------------------|-------------------------------------|
| Cisco Router        | The Cisco Router Device Package | The Cacti Group | http://www.cacti.net | <input checked="" type="checkbox"/> |
| Generic SNMP Device | The Generic SNMP Device Package | The Cacti Group | http://www.cacti.net | <input checked="" type="checkbox"/> |
| Local Linux Machine | The Local Linux Device Package  | The Cacti Group | http://www.cacti.net | <input checked="" type="checkbox"/> |
| Net-SNMP Device     | The Net-SNMP Device Package     | The Cacti Group | http://www.cacti.net | <input checked="" type="checkbox"/> |
| Windows Device      | The Windows Device Template     | The Cacti Group | http://www.cacti.net | <input checked="" type="checkbox"/> |

**NOTE:** Press 'Finish' to complete the installation process after selecting your Device Templates.

Previous Finish

FOSS Linux

After the installation, it will redirect to Login Page.

Not secure | 10.94.10.206/cacti/index.php

### User Login

Enter your Username and Password below

Username

Password

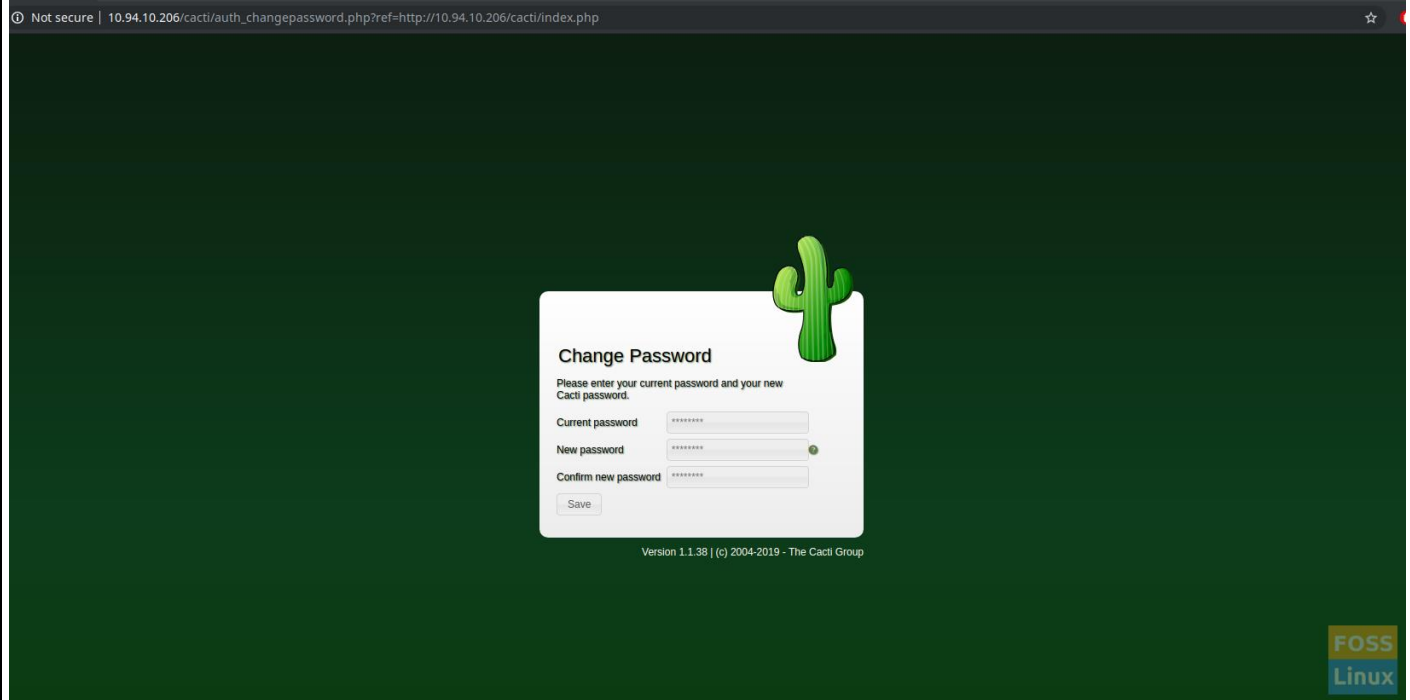
☐ Keep me signed in

Login

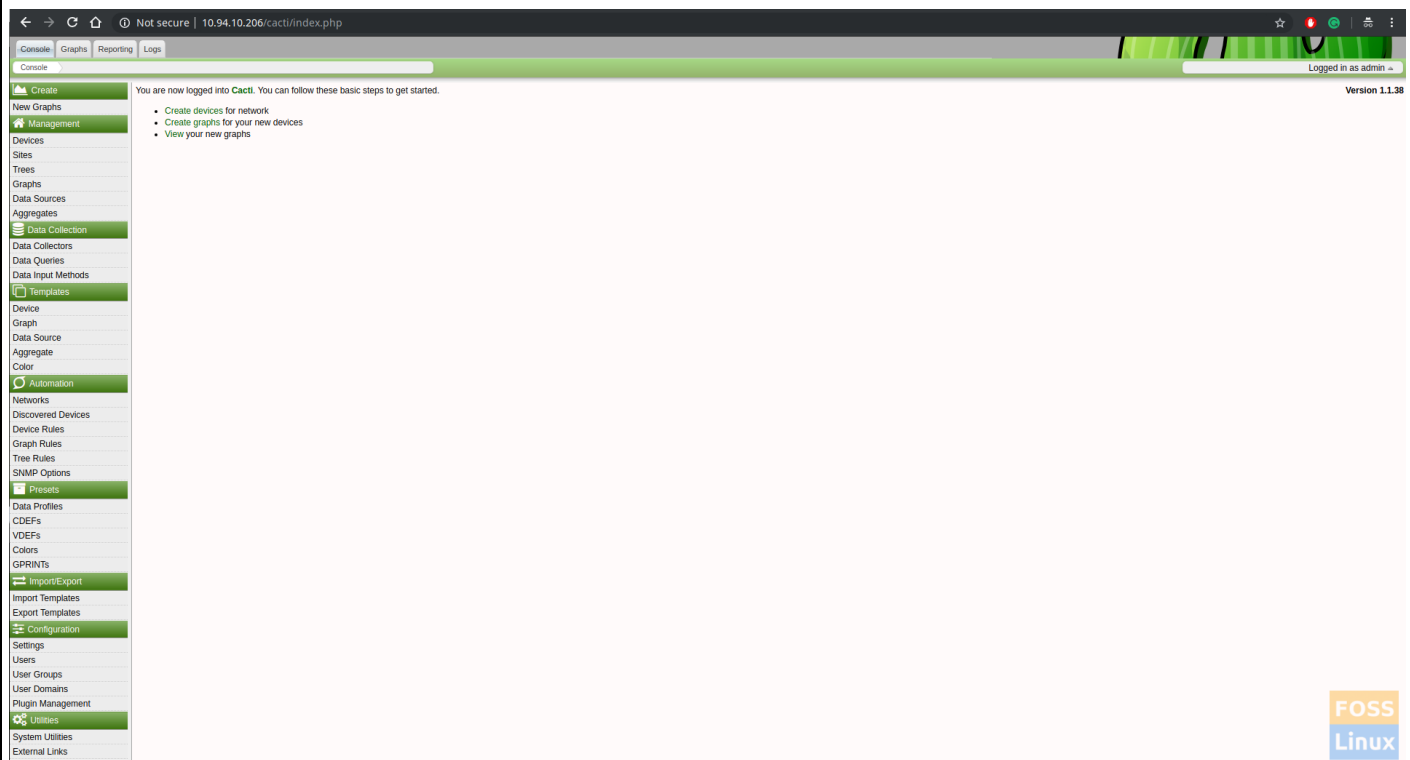
Version 1.1.38 | (c) 2004-2019 - The Cacti Group

FOSS Linux

Using default user name “admin” and default password “admin” you can log in to Cacti server. You should be asked to change the password after that.



Change PasswordChange password and click save. Then you should see the Cacti Dashboard.



You can add new devices from Managemnt-> Devices Then click plus mark “+” on top of the right-hand corner.

## Graphs

That's it! You have successfully configured Cacti on CentOS 7!

<https://www.fosslinox.com/7527/how-to-install-and-configure-cacti-on-centos-7.htm>