Open Atrium 2.45 Installation Instruction
Installing and preparing the Fedora 22 Server operating system to install Open Atrium
(This document should be valid also for Atrium 2.5x or above on Fedora 23 Server or above)

This guide was initially written for **Fedora 22 Server** and OA 2.45.

Operating System used: install Fedora 22 Server (64 bit). The package manager is now **dnf** (**yum** was used up to Fedora 21 and was then superseded by dnf)

```
# dnf install php-mbstring php-xml php-gd php php-pdo php-mysql
```

```
# dnf install gallery2 geos geos-php gallery2-ffmpeg php-tidy php-tcpdf php-simplepie php-symfony-common php-symfony-yaml php-php-
gettext php-pecl-zip php-pecl-apcu php-pear-console-color2 php-pear-Console-Table php-opcache php-mcrypt php-imap php-drush-drush php-
channel-drush php-bcmath php-IDNA_Convert php-tcpdf-dejavu-sans-fonts
```

Go to Web Server Document Root **/var/www/html** and, there, download with *wget* the current Open Atrium archive:

```
# wget http://ftp.drupal.org/files/projects/openatrium-7.x-2.45-core.tar.gz
```

Check its Hash is OK then decompress it:

```
# tar xzf openatrium-7.x-2.45-core.tar.gz
```

Execute a recursive Change Owner (Ownership) on extracted archive:

```
# chown -R apache:apache /var/www/html/openatrium-7.x-2.45/
```

Move whole extracted archive's content on **/var/www/html** and delete both the decompressed folder (**/var/www/html/openatrium-7.x-2.45**) and the downloaded archive file (**openatrium-7.x-2.45**):

```
# mv openatrium-7.x-2.45/* .; rm -fr openatrium-7.x-2.45 openatrium-7.x-2.45-core.tar.gz
```

In the end **/var/www/html** folder should contain a structure similar to the one below:

```
drwxr-xr-x.  9 root   root   4.0K Apr 28 11:26 .
drwxr-xr-x.  6 root   root   4.0K Dec 17 11:07 ..
-rw-r--r--.  1 apache apache 6.5K Apr 20 19:45 authorize.php
-rw-r--r--.  1 apache apache  95K Apr 20 19:45 CHANGELOG.txt
-rw-r--r--.  1 apache apache 1.5K Apr 20 19:45 COPYRIGHT.txt
-rw-r--r--.  1 apache apache  720 Apr 20 19:45 cron.php
drwxr-xr-x.  4 apache apache 4.0K Apr 20 19:45 includes
-rw-r--r--.  1 apache apache  529 Apr 20 19:45 index.php
-rw-r--r--.  1 apache apache 1.7K Apr 20 19:45 INSTALL.mysql.txt
-rw-r--r--.  1 apache apache 1.9K Apr 20 19:45 INSTALL.pgsql.txt
```

```
-rw-r--r--.  1 apache apache  703 Apr 20 19:45 install.php
-rw-r--r--.  1 apache apache 1.3K Apr 20 19:45 INSTALL.sqlite.txt
-rw-r--r--.  1 apache apache  18K Apr 20 19:45 INSTALL.txt
-rw-r--r--.  1 apache apache  18K Apr 20 19:45 LICENSE.txt
-rw-r--r--.  1 apache apache 8.0K Apr 20 19:45 MAINTAINERS.txt
drwxr-xr-x.  4 apache apache 4.0K Apr 20 19:45 misc
drwxr-xr-x. 42 apache apache 4.0K Apr 20 19:45 modules
-rw-r--r--.  1 apache apache  772 Apr 20 19:45 PATCHES.txt
drwxr-xr-x.  6 apache apache 4.0K Apr 20 19:45 profiles
-rw-r--r--.  1 apache apache 5.3K Apr 20 19:45 README.txt
-rw-r--r--.  1 apache apache 1.5K Apr 20 19:45 robots.txt
drwxr-xr-x.  2 apache apache 4.0K Apr 20 19:45 scripts
drwxr-xr-x.  4 apache apache 4.0K Apr 20 19:45 sites
drwxr-xr-x.  7 apache apache 4.0K Apr 20 19:45 themes
-rw-r--r--.  1 apache apache  20K Apr 20 19:45 update.php
-rw-r--r--.  1 apache apache 9.5K Apr 20 19:45 UPGRADE.txt
-rw-r--r--.  1 apache apache 2.2K Apr 20 19:45 web.config
-rw-r--r--.  1 apache apache  417 Apr 20 19:45 xmlrpc.php
```

Prepare the settings.php configuration file

Copy the default.settings.php file available into /var/www/html/sites/default/ renaming it into the required settings.php

```
# cd /var/www/html/sites/default/
# cp -p default.settings.php settings.php
```

Adapt permissions on /var/www/html/sites/default before the start of Atrium installation's script:

Modify (**a+w** or **go+w**) the write permission on the file "settings.php" (/var/www/html/sites/default/settings.php):

```
# ls -lah
total 56K
drwxr-xr-x. 2 apache apache 4.0K Apr 28 13:42 .
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 ..
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 default.settings.php
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 settings.php
# chmod a+w /var/www/html/sites/default/settings.php
# ls -lah
total 56K
drwxr-xr-x. 2 apache apache 4.0K Apr 28 13:42 .
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 ..
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 default.settings.php
-rw-rw-rw-. 1 apache apache  24K Apr 20 19:45 settings.php
```

Modify (**a+w** or **go+w**) the write permission on the folder "default" (/var/www/html/sites/default):

```
# ls -lah
total 24K
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 .
drwxr-xr-x. 9 root   root   4.0K Apr 28 13:57 ..
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 all
drwxr-xr-x. 2 apache apache 4.0K Apr 28 13:42 default
-rw-r--r--. 1 apache apache 2.4K Apr 20 19:45 example.sites.php
-rw-r--r--. 1 apache apache  904 Apr 20 19:45 README.txt
# chmod a+w default
# ls -lah
total 24K
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 .
drwxr-xr-x. 9 root   root   4.0K Apr 28 13:57 ..
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 all
drwxrwxrwx. 2 apache apache 4.0K Apr 28 13:42 default
-rw-r--r--. 1 apache apache 2.4K Apr 20 19:45 example.sites.php
-rw-r--r--. 1 apache apache  904 Apr 20 19:45 README.txt
```

Permissions after installation
Once the installation script has run, Drupal sets the permissions automatically back to 444 (Read Only) for **settings.php** file, and back to 555 (Read+eXecute) for the default folder (**g** and **o**): that's correct and should not be changed...as changing these opens up a security risk.

Prepare sub-folder "files" (/var/www/html/sites/default/files):

```
# mkdir files
# ls -lah
total 60K
drwxrwxrwx. 3 apache apache 4.0K Apr 28 15:25 .
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 ..
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 default.settings.php
drwxr-xr-x. 2 root   root   4.0K Apr 28 15:25 files
-rw-rw-rw-. 1 apache apache  24K Apr 20 19:45 settings.php
# chown -R apache:apache files/
# ls -lah
total 60K
drwxrwxrwx. 3 apache apache 4.0K Apr 28 15:25 .
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 ..
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 default.settings.php
drwxr-xr-x. 2 apache apache 4.0K Apr 28 15:25 files
-rw-rw-rw-. 1 apache apache  24K Apr 20 19:45 settings.php
# chmod a+w files/
# ls -lah
total 60K
drwxrwxrwx. 3 apache apache 4.0K Apr 28 15:25 .
drwxr-xr-x. 4 apache apache 4.0K Apr 20 19:45 ..
-rw-r--r--. 1 apache apache  24K Apr 20 19:45 default.settings.php
drwxrwxrwx. 2 apache apache 4.0K Apr 28 15:25 files
-rw-rw-rw-. 1 apache apache  24K Apr 20 19:45 settings.php
```

SELinux configuration (SELinux is left Enabled)

Verify SELinux (*Drupal wants to have /var/www/html/sites/ and /var/www/html/sites/default/settings.php writable. When dealing with SELinux we need to update context of the /var/www/html/sites/ to read and write*) once that **files** sub-folder was manually created yet:

```
# chcon -R -t httpd_sys_content_t /var/www/html/
# chcon -R -t httpd_sys_content_rw_t /var/www/html/sites/
# setsebool allow_httpd_anon_write=1
# setsebool -P httpd_can_network_connect=1
# chcon -t public_content_rw_t /var/www/html/sites/default/ /var/www/html/sites/default/settings.php
/var/www/html/sites/default/files/
```

Then, after the installation is completed, remember *to reset* previous SELinux level:

After the installation finishes, revert previous settings to benefit from SELinux:

1. Disable allow_httpd_anon_write boolean:
   ```
   setsebool allow_httpd_anon_write=0
   ```
2. Reset security context:
   ```
   chcon -R -t httpd_sys_content_t /var/www/html/sites/default
   ```

WARNING: folder /var/www/html/sites/default/files SELinux attribute configuration:

```
# ls -laZ /var/www/html/sites/default/
dr-xr-xr-x. apache apache unconfined_u:object_r:public_content_rw_t:s0 .
drwxr-xr-x. apache apache unconfined_u:object_r:httpd_sys_rw_content_t:s0 ..
-rw-r--r--. apache apache unconfined_u:object_r:httpd_sys_rw_content_t:s0 default.settings.php
drwxrwxrwx. apache apache unconfined_u:object_r:public_content_rw_t:s0 files
-r--r--r--. apache apache unconfined_u:object_r:public_content_rw_t:s0 settings.php
# chcon -R -t httpd_sys_rw_content_t /var/www/html/sites/default/files/
# ls -laZ /var/www/html/sites/default/
dr-xr-xr-x. apache apache unconfined_u:object_r:public_content_rw_t:s0 .
drwxr-xr-x. apache apache unconfined_u:object_r:httpd_sys_rw_content_t:s0 ..
-rw-r--r--. apache apache unconfined_u:object_r:httpd_sys_rw_content_t:s0 default.settings.php
drwxrwxrwx. apache apache unconfined_u:object_r:httpd_sys_rw_content_t:s0 files
-r--r--r--. apache apache unconfined_u:object_r:public_content_rw_t:s0 settings.php
```

MariaDB configuration

Enable and Start the mysqld.service (MariaDB has been installed on initial steps):

```
# systemctl enable mysqld
# systemctl start mysqld
```

Securing MariaDB and setting MariaDB root's credential

MariaDB root's Password: yourmariadbrootpassword

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: yourmariadbrootpassword
Re-enter new password: yourmariadbrootpassword
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!

MariaDB Database creation for OA

```
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 13
Server version: 10.0.17-MariaDB MariaDB Server

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [(none)]> CREATE DATABASE your_database_name_for_oa_here CHARACTER SET utf8 COLLATE utf8_general_ci;
Query OK, 1 row affected (0.00 sec)
MariaDB [(none)]> CREATE USER 'your_database_user_for_oa_here'@'localhost' IDENTIFIED BY 'your_database_user_password_for_oa_here';
Query OK, 0 rows affected (0.00 sec)
MariaDB [(none)]> SELECT User FROM mysql.user;
+-------------------------------+
| User                          |
+-------------------------------+
| root                          |
| root                          |
| root                          |
| your_database_user_for_oa_here |
+-------------------------------+
4 rows in set (0.00 sec)
MariaDB [(none)]> show grants for 'root'@'localhost';
+-----------------------------------------------------------------------------------------------------------------------+
| Grants for root@localhost                                                                                             |
+-----------------------------------------------------------------------------------------------------------------------+
| GRANT ALL PRIVILEGES ON *.* TO 'root'@'localhost' IDENTIFIED BY PASSWORD '*hashedpasswordherewillbegenerated' WITH GRANT OPTION |
| GRANT PROXY ON ''@'%' TO 'root'@'localhost' WITH GRANT OPTION                                                         |
+-----------------------------------------------------------------------------------------------------------------------+
2 rows in set (0.00 sec)
MariaDB [(none)]> show grants for 'your_database_user_for_oa_here'@'localhost';
+-----------------------------------------------------------------------------------------------------------------------+
| Grants for your_database_user_for_oa_here@localhost                                                                   |
+-----------------------------------------------------------------------------------------------------------------------+
| GRANT USAGE ON *.* TO 'your_database_user_for_oa_here'@'localhost' IDENTIFIED BY PASSWORD '*hashedpasswordherewillbegenerated' |
+---------------------------------------------------------------------------                        -----------+
1 row in set (0.00 sec)
MariaDB [(none)]> GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, DROP, INDEX, ALTER, CREATE TEMPORARY TABLES ON your_database_name_for_oa_here.* TO
'your_database_user_for_oa_here'@'localhost' IDENTIFIED BY 'your_database_user_password_for_oa_here';
Query OK, 0 rows affected (0.00 sec)
MariaDB [(none)]>

Then exit.
```

Apache (httpd) configuration

Enable and Start the **httpd.service** service:

systemctl enable httpd.service
systemctl start httpd.service

Firewall (firewalld) configuration

firewall-cmd --zone=FedoraServer --list-all

```
FedoraServer (default, active)
  interfaces: eno1
  sources:
  services: cockpit dhcpv6-client ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

firewall-cmd --zone=FedoraServer --permanent --add-service=http
firewall-cmd --zone=FedoraServer --permanent --remove-service=dhcpv6-client
firewall-cmd --reload

firewall-cmd --zone=FedoraServer --list-all

```
FedoraServer (default, active)
  interfaces: eno1
  sources:
  services: cockpit http ssh
  ports:
  masquerade: no
  forward-ports:
  icmp-blocks:
  rich rules:
```

PHP Settings (/etc/php.ini) configuration

```
realpath_cache_size = 1M
max_execution_time = 150
max_input_time = 120
memory_limit = 512M
```

```
error_log = /var/log/php_errors.log
post_max_size = 50M
upload_max_filesize = 50M
date.timezone = "Europe/Rome"
date.default_latitude = 45.6663
date.default_longitude = 12.2421
apc.rfc1867 = 1
```

Adapt above values to your Country of origin.

MariaDB (/etc/my.cnf.d/server.conf) configuration

```
bind-address=0.0.0.0 (to avoid the Non-Fatal ERROR related to the disabled Ipv6, if it was disabled as above)
max_allowed_packet=64M
innodb_log_file_size=128M
innodb_buffer_pool_size=512MB
```

At this point you could visit the URL where is it supposed the web server is running on (I assume DNS and resolv.conf works correctly on localhost) http://youripaddresshere/settings.php to start the Open Atrium 2 installation procedure using values configured in the previous steps.

If you need to **drop** the database and its user:

MariaDB (for re-install from beginning with a very new"fresh site")

Drop Database:

DROP DATABASE *databasename*;

Drop user:

DROP USER *username*;

Where databasename and username are `your_database_name_for_oa_here` and `your_database_user_for_oa_here`.