

A little guide, in cookbook stile, to install AtoM into Linux RedHat or CentOS release 7 from zero.

We assume:

- IP: 192.168.1.10
- Hostname: atom.example.com
- User: admin
- Pass: My000!Pass2
- DB user 1: root
- DB pass 1: Admin!Pass2
- DB user 2: atom
- DB pass 2: atom1Pass
- URL: http://localhost/atom
- URL: http://atom.example.com/atom

[Per scaricare il file \(ODT e PDF\)&#160;clicca qui](#) .

### **1. Prepare server**

-

Install RedHat or CentOS as basic server

-

after reboot install epel repo

```
# yum install epel-release
```

-

if you use RedHat enable rhel-7-server-optional-rpms repo:

-

```
edit /etc/repo.d/ redhat.repo
```

-

find rhel-7-server-optional-rpms section and set enable "up"

...

enable = 1

...

-

install some utility

```
# yum install rpmdelta unzip bzip2 mailx ntp yum-cron sudo
```

-

some optional utility (but useful!)

```
# yum install mc htop iotop iftop bind-utils
```

-

update your server

```
# yum update -y
```

-

and reboot

```
# reboot
```

## **1.1. First tuning**

-

I prefer old iptables:

-

remove firewalld

```
# yum remove firewalld
```

-

install iptables-service

```
# yum install iptables-services
```

```
-
```

```
edit /etc/sysconfig/iptables
```

```
# vim /etc/sysconfig/iptables
```

```
-
```

```
set iptables rule (example routes)
```

```
*filter
```

```
:INPUT ACCEPT [0:0]
```

```
:FORWARD ACCEPT [0:0]
```

```
:OUTPUT ACCEPT [0:0]
```

-A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT

-A INPUT -p icmp -j ACCEPT

-A INPUT -i lo -j ACCEPT

-A INPUT -p tcp -m state --state NEW -m tcp --dport 22 -j ACCEPT

-A INPUT -p tcp -m state --state NEW -m tcp --dport 80 -j ACCEPT

-A INPUT -j REJECT --reject-with icmp-host-prohibited

-A FORWARD -j REJECT --reject-with icmp-host-prohibited

COMMIT

-

enable and start iptables

# systemctl enable iptables

# systemctl start iptables

-

set ntp:

-

edit /etc/ntp.conf

# vim /etc/ntp.conf

-

set country server (example for Italy)

...

server 0.it.pool.ntp.org iburst

server 1.it.pool.ntp.org iburst

server 3.it.pool.ntp.org iburst

server 4.it.pool.ntp.org iburst

...

-

force synchronization

# ntpdate 0.it.pool.ntp.org

-

enable and start ntp

# systemctl enable ntpd

# systemctl start ntpd

-

enforce the security. Make a one new user and lock root:

-

create sudo group

```
# groupadd sudo
```

```
-
```

```
create new user "admin"
```

```
# useradd -m -G sudo -s /bin/bash admin
```

```
-
```

```
set password for admin
```

```
# passwd admin
```

```
New password: My000!Pass2
```

```
Retype new password: My000!Pass2
```

```
-
```

```
run visudo
```



```
# visudo
```

```
-
```

and append the following line

```
...
```

```
%sudo ALL=/usr/bin/su
```

```
-
```

lock root

```
# passwd -l root
```

```
-
```

from now, only "admin" can login, only users of sudo group can became root with the command

```
# sudo su -
```

```
-
```

```
enable yum-cron
```

```
# touch /var/lock/subsys/yum-cron
```

```
-
```

```
disable SELinux (if it active AtoM not works properly or you have set security contest)
```

```
-
```

```
edit /etc/sysconfig/selinux
```

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

```
-
```

```
update the follow line
```

```
...
```

SELINUX=disabled

...

-

reboot the server

# reboot

## 1.2. Install dependencies

-

Install MariaDB (the open fork of MySQL)

-

install

# yum install mariadb mariadb-server

-

enable and run MariaDB

```
# systemctl enable mariadb
```

```
# systemctl start mariadb
```

```
-
```

secure the RDBM

NB: set "Admin!Pass2" as password for root and answer "yes" for any question

```
# mysql_secure_installation
```

```
-
```

install Apache and PHP

```
-
```

install packages

```
# yum install httpd php php-cli php-JsonSchema php-fpm php-cli php-mysql
```

```
php-gd php-imap php-ldap php-odbc php-pear php-xml php-xmlrpc  
php-magickwand php-magpierss php-mbstring php-mcrypt php-shout php-snmp  
php-soap php-tidy php-pecl-apc
```

-

```
set php.ini:
```

-

```
edit /etc/php.ini
```

```
# vi /etc/php.ini
```

-

```
set follow lines
```

```
...
```

```
post_max_size = 120M
```

...

```
cgi.fix_pathinfo=0
```

...

```
upload_max_filesize = 100M
```

...

```
date.timezone = 'Europe/Rome'
```

...

-

```
install memcached:
```

-

```
install
```

```
# yum install memcached php-pecl-memcache
```

-

```
edit /etc/sysconfig/memcached
```

```
# vim /etc/sysconfig/memcached
```

-

and set

```
PORT="11211"
```

```
USER="memcached"
```

```
MAXCONN="1024"
```

```
CACHESIZE="64"
```

```
OPTIONS=""
```

-

enable and start

```
# systemctl start memcached
```

```
# systemctl enable memcached
```

-

set php-fpm

-

```
remove /etc/php-fpm.d/www.conf
```

```
# rm -rf /etc/php-fpm.d/www.conf
```

-

```
create /etc/php-fpm.d/atom.conf
```



```
# vim /etc/php-fpm.d/atom.conf
```

```
-
```

and paste (please: set the real IP at the bottom)

```
[atom]
```

```
# The user running the application
```

```
user = apache
```

```
group = apache
```

```
# Use UNIX sockets if Nginx and PHP-FPM are running in the same machine
```

listen = /var/run/php-fpm.atom.sock

listen.owner = apache

listen.group = apache

listen.mode = 0600

# The following directives should be tweaked based in your hardware resources

pm = dynamic

pm.max\_children = 30

pm.start\_servers = 10

pm.min\_spare\_servers = 10

pm.max\_spare\_servers = 10

```
pm.max_requests = 200
```

```
chdir = /
```

```
# Some defaults for your PHP production environment
```

```
# A full list here: http://www.php.net/manual/en/ini.list.php
```

```
php_admin_value[expose_php] = off
```

```
php_admin_value[allow_url_fopen] = on
```

```
php_admin_value[memory_limit] = 512M
```

```
php_admin_value[max_execution_time] = 120
```

```
php_admin_value[post_max_size] = 72M
```

```
php_admin_value[upload_max_filesize] = 64M
```

php\_admin\_value[max\_file\_uploads] = 10

php\_admin\_value[cgi.fix\_pathinfo] = 0

php\_admin\_value[display\_errors] = off

php\_admin\_value[display\_startup\_errors] = off

php\_admin\_value[html\_errors] = off

php\_admin\_value[session.use\_only\_cookies] = 0

# APC, which is still used in PHP 5.5 for userland memory cache unless you

# are switching to something like sfMemcacheCache

php\_admin\_value[apc.enabled] = 1

php\_admin\_value[apc.shm\_size] = 64M

```
php_admin_value[apc.num_files_hint] = 5000
```

```
php_admin_value[apc.stat] = 0
```

```
# Zend OPcache
```

```
# Only in Ubuntu 14.04 (PHP 5.5).
```

```
# Don't use this in Ubuntu 12.04, it won't work.
```

```
php_admin_value[opcache.enable] = 1
```

```
php_admin_value[opcache.enable_cli] = 0
```

```
php_admin_value[opcache.memory_consumption] = 192
```

```
php_admin_value[opcache.interned_strings_buffer] = 16
```

```
php_admin_value[opcache.max_accelerated_files] = 4000
```

```
php_admin_value[opcache.validate_timestamps] = 0
```

```
php_admin_value[opcache.fast_shutdown] = 1
```

```
# This is a good place to define some environment variables, e.g. use
```

```
# ATOM_DEBUG_IP to define a list of IP addresses with full access to the
```

```
# debug frontend or ATOM_READ_ONLY if you want AtoM to prevent
```

```
# authenticated users
```

```
env[ATOM_DEBUG_IP] = "192.168.1.10,127.0.0.1"
```

```
env[ATOM_READ_ONLY] = "off"
```

```
-
```

```
enable and start php-fpm
```

```
# systemctl enable php-fpm
```

```
# systemctl start php-fpm
```

-

```
install Java
```

```
# yum install java-1.7.0-openjdk.x86_64
```

-

```
install ImageMagick
```

```
# yum install ImageMagick
```

-

```
install Ghostscript
```

```
# yum install ghostscript
```

-

```
install FFmpeg
```

-

```
install ATrpms repo
```

```
# yum install
```

```
http://ftp-stud.fht-esslingen.de/Mirrors/atrpms/dl.atrpms.net/el7-x86_64/atrpms/stable/atrpms-repo-7-7.el7.x86_64.rpm
```

```
-
```

```
edit /etc/yum.repos.d/atrpms.repo
```

```
# vim /etc/yum.repos.d/atrpms.repo
```

```
-
```

```
and fix following line
```

```
...
```

```
#baseurl=http://dl.atrpms.net/el$releasever-$basearch/atrpms/stable
```

```
baseurl=http://ftp-stud.fht-esslingen.de/Mirrors/atrpms/dl.atrpms.net/el$releasever-$basearch/atrpms/stable/
```



enabled=1

...

-

install ffmpeg packages

# yum install ffmpeg ffmpeg-devel

-

install pdftotext (part of poppler-utils)

# yum install poppler-utils

-

install Apache FOP

**NB:** RedHatCentOS have the 1.1 FOP version in the default repo, but the 1.1 FOP don't work whit AtoM 2.2

-

make a temporary directory

```
# mkdir ~/Tmp
```

```
# cd ~/Tmp
```

-

download package from Apache

```
# wget https://archive.apache.org/dist/xmlgraphics/fop/binaries/fop-1.0-bin.tar.gz
```

-

extract files

```
# tar -zxvf fop-1.0-bin.tar.gz
```

```
# rm fop-1.0-bin.tar.gz
```

-

move the files into right directory

```
# mv fop-1.0 /usr/share
```

-

create a link and the environment file

```
# ln -s /usr/share/fop-1.0/fop /usr/bin/fop
```

```
# echo 'FOP_HOME="/usr/share/fop-1.0"' >> /etc/environment
```

-

install German job server and make a worker

-

install Gearman, enable and run

```
# yum install gearmand php-pecl-gearman
```

```
# systemctl enable gearmand
```

```
# systemctl start gearmand
```

-

edit the option file

```
# vim /etc/sysconfig/gearmand
```

-

and paste

```
## Settings for gearmand
```

```
# OPTIONS=""
```

```
OPTIONS="--listen=* --port=4730"
```

-

create the option file for the worker

```
# vim /etc/sysconfig/atom-worker
```

```
-
```

and paste (please: customize properly)

```
# Variables setting for atom-worker script
```

```
# location of webroot for AtoM
```

```
# default: /var/www/html/atom
```

```
LOCATION=/var/www/html/atom
```

```
# Log file
```

```
# BE CAREFUL: not implemented now
```

#LOGFILE=/var/log/gearmand/atom-worker.log

# Pid file

# default: /var/run/atom-worker.pid

PIDFILE=/var/run/atom-worker.pid

# User with run atom-worker

# default: apache

USER=apache

# Group with run atom-worker

# default: apache

GROUP=apache

# Path/file php program

# Note: \$(which php) find automatly the full path. Alternatly you can write by hand.

# Ex.: /usr/local/bin/php

# default: \$(which php)

PROGPHP=\$(which php)

-

create the script to control the worker

# vim /usr/local/bin/atom-worker

-

and paste

```
#!/bin/bash
```

```
#
```

```
# AtoM worker (germand) SystemD service.
```

```
# Source function library.
```

```
./etc/init.d/functions
```

```
if [ -f /etc/sysconfig/atom-worker ]; then
```

```
./etc/sysconfig/atom-worker
```

```
fi
```



```
[[ -z "$LOCATION" ]] && LOCATION=/var/www/html/atom
```

```
#[[ -z "$LOGFILE" ]] && LOGFILE=/var/log/gearmand/atom-worker.log
```

```
[[ -z "$PIDFILE" ]] && PIDFILE=/var/run/atom-worker.pid
```

```
[[ -z "$USER" ]] && USER=apache
```

```
[[ -z "$GROUP" ]] && GROUP=apache
```

```
[[ -z "$PROGPHP" ]] && PROGPHP=$(which php)
```

```
case "$1" in
```

```
start)
```

```
rc=0
```

```
/usr/bin/su -g $GROUP -s /bin/bash $USER -c '/usr/bin/php -d memory_limit=-1 -d error_reporting="E_ALL" /var/www/html/atom/symfony jobs:worker >> /var/log/gearmand/atom-worker.log 2>&1 &'
```

```
rc=$?
```

```
echo "`pgrep -fl "/usr/bin/php -d memory_limit=-1 -d error_reporting=E_ALL /var/www/html/atom/symfony" | awk '{print ""$1""}'`" > ${PIDFILE}
```

```
::
```

```
stop)
```

```
[ "$EUID" != "0" ] && exit 4
```

```
rc=0
```

```
/usr/bin/kill -9 `cat ${PIDFILE}`
```

```
rc=$?
```

```
echo "`date +"%Y-%m-%d %H:%M:%S"` > stop atom-worker" >> /var/log/gearmand/atom-worker.log
```

```
/bin/rm ${PIDFILE}
```

```
;;
```

```
status)
```

```
echo $"Configured devices:"
```

```
;;
```

```
restart|reload|force-reload)
```

```
rc=0
```

```
$0 stop
```

```
$0 start
```

```
rc=$?
```

```
;;
```

```
*)
```

```
echo $"Usage: $0 {start|stop|status|restart|reload|force-reload}"
```

```
exit 2
```

```
esac
```

```
exit $rc
```

```
-
```

```
set file permissions
```

```
# chmod +x /usr/local/bin/atom-worker
```

```
# mkdir /var/log/gearmand
```

```
# chown apache.apache /var/log/gearmand
```

```
-
```

```
create the systemd script
```

```
# vim /usr/lib/systemd/system/atom-worker.service
```

```
-
```

and paste

```
[Unit]
```

```
Description=Gearman worker for AtoM
```

```
After=network.target
```

```
[Service]
```

```
Type=simple
```

```
RemainAfterExit=yes
```

```
ExecStart=/usr/local/bin/atom-worker start
```

ExecReload=/usr/local/bin/atom-worker restart

ExecStop=/usr/local/bin/atom-worker stop

[Install]

WantedBy=multi-user.target

-

enable and start the worker

# systemctl daemon-reload

# systemctl enable atom-worker

# systemctl start atom-worker

# systemctl status atom-worker

-

set logrotate.

-

Create gearman log rotate

```
# vim /etc/logrotate.d/gearman
```

-

and paste

```
/var/log/gearmand.log
```

```
/var/log/gearmand/*.log
```

```
{
```

```
daily
```

missingok

notifempty

delaycompress

rotate 10

compress

size 1M

}

-

install Elasticsearch

-

create /etc/yum.repos.d/elasticsearch.repo

# vim /etc/yum.repos.d/elasticsearch.repo



-

and paste

```
[elasticsearch-1.5]
```

```
name=Elasticsearch repository for 1.5.x packages
```

```
baseurl=http://packages.elastic.co/elasticsearch/1.5/centos
```

```
gpgcheck=1
```

```
gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch
```

```
enabled=0
```

```
[elasticsearch-1.7]
```

```
name=Elasticsearch repository for 1.7.x packages
```

baseurl=http://packages.elastic.co/elasticsearch/1.7/centos

gpgcheck=1

gpgkey=http://packages.elastic.co/GPG-KEY-elasticsearch

enabled=1

-

update yum database

# yum update

-

install

# yum -y install elasticsearch

-

enable and activate

```
# systemctl enable elasticsearch
```

```
# systemctl start elasticsearch
```

-

if you like to test, tray

```
# curl -X GET http://localhost:9200/
```

-

Prepare a database for AtoM

-

create a database

```
# mysql -h localhost -u root -p -e "CREATE DATABASE atom CHARACTER SET  
utf8 COLLATE utf8_unicode_ci;"
```

-

create a user for "atom" DB

```
# mysql -h localhost -u root -p -e "GRANT INDEX, CREATE, SELECT, INSERT,
UPDATE, DELETE, ALTER, LOCK TABLES ON atom.* TO 'atom'@'localhost'
IDENTIFIED BY 'Admin!Pass2';"
```

## 2. Install AtoM

-

Go into temporary directory

```
# cd ~/Tmp
```

-

download package

```
# wget https://storage.accesstomemory.org/releases/atom-2.2.0.tar.gz
```

-

extract files

```
# tar xzf atom-2.2.0.tar.gz
```

-

copy the extract into webroot

```
# cp -R ./atom-2.2.0 /var/www/html/atom
```

-

set correct permission

```
# chown -R apache.apache /var/www/html/atom
```

-

rename the configuration file

```
# mv /var/www/html/atom/config/config.php  
/var/www/html/atom/config/config.php.distro
```

-

start Apache httpd and Php-pfm

```
# systemctl enable httpd
```

```
# systemctl start httpd
```

```
# systemctl enable php-fpm
```

```
# systemctl start php-fpm
```

-

use a web browser and point to <http://atom.example.com/atom> (or <http://192.168.1.10/atom> )

-

fit the fields with the dates:

Database name: atom

Database username: atom

Database password: atom1Pass

Database host: localhost

Database port: 3306

Search host: localhost

Search port: 9200

Search index: atom

-

the rest of the fields can be filled as you need:

Site title

Site description

Site base URL

Username

E-mail address

Password

Now that AtoM is installed and running!

### **3. Bibliography**

<https://www.accesstomemory.org/>

<https://www.accesstomemory.org/it/docs/2.2/admin-manual/installation/linux/#installation-linux>

<https://groups.google.com/forum/#!forum/ica-atom-users>

<https://www.ica-atom.org/>

<https://wiki.ica-atom.org/>

<https://www.elastic.co/downloads/>

<http://gearman.org/>

<http://www.if-not-true-then-false.com/2010/install-memcached-on-centos-fedora-red-hat>

<http://www.itzgeek.com/how-tos/linux/centos-how-tos/install-memcached-on-centos-7.htm>

<https://www.accesstomemory.org/en/docs/2.2/admin-manual/installation/linux/>

<http://stackoverflow.com/questions/14982921/how-to-install-gearman-with-php-extension>

<http://www.itzgeek.com/how-tos/linux/ubuntu-how-tos/install-elasticsearch-on-centos-7-ubuntu-14-10-linux-mint-17-1.html>