

Instalação do AtoM no CENTOS 7

I. ATUALIZANDO SERVIDOR E INSTALANDO PACOTES NECESSÁRIOS

```
sudo yum update -y && sudo yum upgrade -y  
sudo yum install -y wget nano
```

II. INSTALANDO O NGINX

1º Passo - Repositório EPEL

Para adicionar o repositório EPEL, abra um terminal e use o seguinte comando:

```
sudo yum install epel-release -y
```

2º Passo - Instalar o Nginx

Para instalar o NGINX execute o seguinte comando yum:

```
sudo yum install nginx -y
```

3º Passo - Iniciar o NGINX

Verifique o status do nginx:

```
sudo systemctl status nginx.service
```

Se inactive, execute:

```
sudo systemctl start nginx.service
```

Execute o comando a seguir para habilitar o nginx para inicializar junto com a máquina:

```
sudo systemctl enable nginx.service
```

Execute os seguintes comandos para permitir tráfego HTTP e HTTPS:

```
sudo firewall-cmd --permanent --zone=public --add-service=http  
sudo firewall-cmd --permanent --zone=public --add-service=https  
sudo firewall-cmd --reload
```

4º Passo - Set Up Environment for Server Block Files

1. Criar os diretórios sites-available e sites-enabled:

```
sudo mkdir -p /etc/nginx/sites-available  
sudo mkdir -p /etc/nginx/sites-enabled
```

2. Abrir arquivo de configuração do Nginx:

```
sudo nano /etc/nginx/nginx.conf
```

3. Incluir as duas linhas no arquivo e comentar todo comando do server para adicionar o diretório do sites-enabled:

```
include /etc/nginx/sites-enabled/*;  
server_names_hash_bucket_size 64;
```

Arquivo completo ficaria:

```
user nginx;
worker_processes auto;
error_log /var/log/nginx/error.log;
pid /run/nginx.pid;

include /usr/share/nginx/modules/*.conf;

events {
    worker_connections 1024;
}

http {
    log_format main '$remote_addr - $remote_user [$time_local] "$request" '
        '$status $body_bytes_sent "$http_referer" '
        '"$http_user_agent" "$http_x_forwarded_for"';

    access_log /var/log/nginx/access.log main;
    sendfile on;
    tcp_nopush on;
    tcp_nodelay on;
    keepalive_timeout 65;
    types_hash_max_size 4096;
    include /etc/nginx/mime.types;
    default_type application/octet-stream;
    include /etc/nginx/conf.d/*.conf;
    include /etc/nginx/sites-enabled/*;
    server_names_hash_bucket_size 64;
}
```

4. Verificar se a sintaxe do arquivo está correta:

```
sudo nginx -t
```

Resultado do comando acima:

```
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

III. INSTALANDO O ELASTICSEARCH 5.X

1º Passo - Java 1.8

1. Execute os seguintes comandos:

```
sudo yum install java-1.8.0-openjdk.x86_64 -y
```
2. set \$JAVA_HOME para:

```
export JAVA_HOME=/usr/lib/jvm
```


ou

sudo nano /etc/environment

3. Adicionar a seguinte linha ao final:
export JAVA_HOME=/usr/lib/jvm
4. Salvar o arquivo e sair; depois executar o seguinte comando
source /etc/environment
5. Verificar a versão do java:
java -version

Resposta:

openjdk version "1.8.0_332"
OpenJDK Runtime Environment (build 1.8.0_332-b09)
OpenJDK 64-Bit Server VM (build 25.332-b09, mixed mode)

2º Passo - Elasticsearch

1. Importando as chaves:
sudo rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch
2. Adicionando os repositórios:
sudo nano /etc/yum.repos.d/Elasticsearch.repo
3. Copiar, adicionar o texto a seguir, salvar e sair:
*[elasticsearch-5.x]
name=Elasticsearch repository for 5.x packages
baseurl=https://artifacts.elastic.co/packages/5.x/yum
gpgcheck=1
gpgkey=https://artifacts.elastic.co/GPG-KEY-elasticsearch
enabled=1
autorefresh=1
type=rpm-md*
4. Instalando o Elasticsearch:
*sudo yum install elasticsearch -y
sudo /bin/systemctl start elasticsearch.service*
5. Iniciando o Elasticsearch automaticamente:
*sudo /bin/systemctl daemon-reload
sudo /bin/systemctl enable elasticsearch.service*

IV. INSTALANDO E CONFIGURANDO SERVIDOR ATOM

1º Passo - Criando o arquivo de configuração do Atom

*sudo touch /etc/nginx/sites-available/atom
sudo ln -sf /etc/nginx/sites-available/atom /etc/nginx/sites-enabled/atom
sudo rm /etc/nginx/sites-enabled/default*

2º Passo - Adicionando informações no arquivo de configuração atom

1. Abrir o arquivo:
sudo nano /etc/nginx/sites-available/atom
2. Adicionar o conteúdo a seguir, salvar e sair:

```
upstream atom {  
    server unix:/run/php7.2-fpm.atom.sock;  
}  
server {  
    listen 80;  
    root /usr/share/nginx/atom;  
    server_name _;  
    client_max_body_size 72M;  
  
    location / {  
        try_files $uri /index.php?$args;  
    }  
  
    location ~ /\. {  
        deny all;  
        return 404;  
    }  
  
    location ~* (\.yaml|\.ini|\.tmpl)$ {  
        deny all;  
        return 404;  
    }  
  
    location ~*/(?:(uploads|files)/.*\.php$ {  
        deny all;  
        return 404;  
    }  
  
    location ~* /uploads/r/(.*)/conf/ {  
  
    }  
  
    location ~* ^/uploads/r/(.*)$ {  
        include /etc/nginx/fastcgi_params;  
        set $index /index.php;  
        fastcgi_param SCRIPT_FILENAME $document_root$index;  
        fastcgi_param SCRIPT_NAME $index;  
        fastcgi_pass atom;  
    }  
  
    location ~ ^/private/(.*)$ {  
        internal;  
        alias /usr/share/nginx/atom/$1;  
    }  
}
```

```

location ~ ^/(index|qubit_dev)\.php(/)?$ {
    include /etc/nginx/fastcgi_params;
    fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
    fastcgi_split_path_info ^(.+\.(php|/.*)$);
    fastcgi_pass atom;
}

location ~* \.php$ {
    deny all;
    return 404;
}
}

```

3º Passo - Reload o nginx

```
sudo systemctl reload nginx.service
```

V. INSTALANDO PHP e BIBLIOTECAS

1º Passo: ativando epel

1. Ativar EPEL:

```

sudo yum install -y http://rpms.remirepo.net/enterprise/remi-release-7.rpm
sudo yum install yum-utils -y
sudo yum-config-manager --enable remi-php72

```

2. Instalar pacotes PHP:

```

sudo yum install -y php php-fpm php-mbstring php-gd php-cli php-ldap php-json
php-xml php-xmlrpc php-pecl-apcu php-pecl-apcu-devel
php-pear-MDB2-Driver-mysql php-phpiredis php-mysqlnd php-opcache
php-pecl-zendopcache php-pear-CAS

sudo yum install php-cli php-ldap php-opcache php-zip php-apcu php-memcache
php-gearman -y

```

3. Habilitar o php-fpm para inicializar juntamente com a máquina:

```
systemctl enable php-fpm
```

2º Passo: Configurar PHP-FPM para atom

1. Crie o arquivo:

```
sudo nano /etc/php-fpm.d/atom.conf
```

2. Adicionar o conteúdo a seguir, salvar e sair:

```
[atom]
```

```
; The user running the application
```

```
user = nginx
```

```
group = nginx
```

```
; Use UNIX sockets if Nginx and PHP-FPM are running in the same machinecache
```

```

listen = /run/php7.2-fpm.atom.sock
listen.owner = nginx
listen.group = nginx
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
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
php_admin_value[opcache.enable] = 1
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] = "10.10.10.10,127.0.0.1"
env[ATOM_READ_ONLY] = "off"

```

3. Exclua o arquivo [www.conf](#)
`rm -rf /et/php-fpm.d/www.conf`

VI. INSTALANDO O GEARMAND

1ª Passo: Instalando o Gearmand

1. Executar os comando para instalação
sudo yum -y install https://repo.ius.io/ius-release-el\$(rpm -E '%{rhel}').rpm
sudo yum update -y
2. Instalar os pacotes requeridos
sudo yum -y install gcc-c++ gperf uuid-devel libuuid libuuid-devel uuid boost-devel libevent libevent-devel gcc
3. Instalar o Gearman Lib
sudo yum -y install libgearman-devel

2ª Passo: Atualizando o PHP.INI com as bibliotecas do Gearman

1. Criar o arquivo de configuração do Gearman
sudo touch /etc/php.d/gearman.ini
2. Executar o comando como administrador
sudo echo 'extension=gearman.so' >> /etc/php.d/gearman.ini
3. Verificar se a configuração do PHP refletiu no PHP. Verifique se o “gearman” aparece na lista
php -m
4. Instalar o Servidor Gearman
sudo yum install gearmand -y
5. Habilitando o gearmand para iniciar com os serviços
sudo systemctl enable gearmand
sudo systemctl start gearmand

Referência:

<https://gist.github.com/hemantshekhawat/2607dd5351a81dee8ff4bc521ca4ec40#file-gearmaninstall-centos7-sh>

VII. INSTALANDO O MEMCACHED

sudo yum install -y memcached
sudo yum install -y php-pecl-memcache

VIII. INSTALANDO O ATOM

1. Baixar o pacote do AtoM do repositório

```
wget https://storage.accesstomemory.org/releases/atom-2.6.4.tar.gz
sudo mkdir -p /usr/share/nginx/atom
sudo tar xzf atom-2.6.4.tar.gz -C /usr/share/nginx/atom --strip 1
cd /usr/share/nginx/atom/log/
touch qubit_prod.log
```

2. Configurando as permissões:

```
sudo chown -R nginx:nginx /usr/share/nginx/atom
sudo chmod o= /usr/share/nginx/atom
```

IX. INSTALANDO OUTROS PACOTES

1. Instalando Fop, ImageMagick, Ghostscript e Poppler-utils

```
sudo yum install fop -y
sudo yum install ImageMagick -y
sudo yum install ghostscript -y
sudo yum -y install poppler-utils
```

2. Instalando FFPMEG

```
sudo yum install epel-release

sudo yum -y localinstall --nogpgcheck
https://download1.rpmfusion.org/free/el/rpmfusion-free-release-7.noarch.rpm

sudo yum install ffmpeg ffmpeg-devel -y
ffmpeg -version
```

3. Instalando o Less para compilar o template

```
sudo yum install npm -y
sudo npm install -g less
sudo npm install -g less-plugin-clean-css
sudo ln -s /usr/local/bin/lessc /usr/bin/lessc
```

X. Instalando o MySQL 8.0

1. Instalando o Banco de Dados

```
sudo yum update -y
sudo wget https://dev.mysql.com/get/mysql80-community-release-el7-3.noarch.rpm
sudo rpm --import https://repo.mysql.com/RPM-GPG-KEY-mysql-2022
sudo rpm -Uvh mysql80-community-release-el7-3.noarch.rpm
sudo yum install mysql-server -y
sudo systemctl enable mysqld
sudo systemctl start mysqld
```

2. Recuperar a senha gerada para o banco de dados:
sudo grep 'password' /var/log/mysqld.log
3. Para mudar a senha, deve executar o seguinte comando:
sudo mysql_secure_installation
4. Configurando o Banco de Dados
mysql -h localhost -u root -p -e "CREATE DATABASE atom CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci;"
5. Acessar o banco de dados e executar os seguintes comandos:
sudo mysql -h localhost -u root -p
CREATE USER 'atom'@'%' IDENTIFIED BY '#Atom12345';
GRANT ALL PRIVILEGES ON atom. TO 'atom'@'%';*
6. Se **NÃO** for possível criar o usuário com GRANT anterior, então deve-se utilizar:
CREATE USER 'atom'@'%' IDENTIFIED WITH mysql_native_password BY '#Atom12345';
GRANT ALL PRIVILEGES ON atom. TO 'atom'@'%';*
7. Saia do modo mysql
exit

XI. ACESSAR A APLICAÇÃO E REALIZAR CONFIGURAÇÃO

1. Realizar o restart das aplicações
sudo systemctl restart nginx
sudo systemctl restart php-fpm
sudo systemctl restart elasticsearch
sudo systemctl restart gearmand
sudo systemctl restart memcached
2. Acessar a aplicação e realizar a configuração
http://localhost ou *http:ip-do-servico*
3. Informar os parâmetros configurados
Database name: atom
Database username: atom
Database password: #Atom12345
Database host: localhost
Database port: 3306
Search host: localhost
Search port: 9200
Search index: atom
4. Se aparecer o erro a seguir:

Unable to open PDO connection [wrapped: SQLSTATE[HY000] [2054] The server requested authentication method unknown to the client]

Entrar no modo mysql e rodar:

```
ALTER USER atom@'localhost' IDENTIFIED WITH mysql_native_password
BY '#Atom12345';
GRANT ALL PRIVILEGES ON atom.* TO 'atom'@'%';
Restart
```

5. Usuário e senha padrão da aplicação

Usuário: demo@example.com

Senha: demo

XII. INSTALANDO O ATOM-WORKER

1. Abrir arquivo de configuração do gearmand

```
sudo nano /etc/sysconfig/gearmand
```

2. Colar texto ao final do documento

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

3. Criar arquivo no systemd do Centos 7

```
sudo nano /usr/lib/systemd/system/atom-worker.service
```

4. Colar texto do systemd

```
[Unit]
```

```
Description=AtoM worker
```

```
After=network.target
```

```
# High interval and low restart limit to increase the possibility
```

```
# of hitting the rate limits in long running recurrent jobs.
```

```
StartLimitIntervalSec=24h
```

```
StartLimitBurst=3
```

```
[Install]
```

```
WantedBy=multi-user.target
```

```
[Service]
```

```
Type=simple
```

```
# User=nginx
```

```
# Group=nginx
```

```
WorkingDirectory=/usr/share/nginx/atom
```

```
ExecStart=/usr/bin/php -d memory_limit=-1 -d error_reporting="E_ALL" symfony
```

```
jobs:worker
```

```
KillSignal=SIGTERM
```

```
Restart=on-failure
```

```
RestartSec=30
```

5. Executar os seguintes comandos:

```
sudo systemctl daemon-reload  
sudo systemctl enable atom-worker  
sudo systemctl start atom-worker
```

XIII. ATUALIZAÇÕES/AJUSTES

1º Passo: Configuração do Mysql

1. Criar o arquivo mysqld.cnf, caso não exista:

```
sudo touch /etc/mysql/conf.d/mysqld.cnf
```
2. Acessar o arquivo:

```
sudo nano /etc/mysql/conf.d/mysqld.cnf
```
3. Adicionar o seguinte conteúdo:

```
[mysqld]  
sql_mode=ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION  
optimizer_switch='block_nested_loop=off'
```
4. Alterar o arquivo my.cnf

```
sudo nano /etc/my.cnf
```
5. Adicionar o conteúdo a seguir abaixo da linha [mysqld]

```
!includedir /etc/my.cnf.d  
!includedir /etc/mysql/conf.d
```
6. Reiniciar o serviço mysql

```
systemctl restart mysqld
```
7. Verificar se foi aplicada as alterações

```
select @@global.sql_mode;
```

Referência:

<https://www.accesstomemory.org/pt-br/docs/2.6/admin-manual/maintenance/common-atom-queries/#check-server-sql-modes>

2º Passo: Correção de vulnerabilidade do Log4j

1. Correção de vulnerabilidade do log4j

```
yum install zip -y  
cd /usr/share/elasticsearch/lib/  
zip -q -d log4j-core-*.jar org/apache/logging/log4j/core/lookup/JndiLookup.class  
service elasticsearch restart
```

Referência: <https://wiki.accesstomemory.org/wiki/Development/Security/Log4j-2021-12>

3º Passo: Correção de problemas relacionados ao cache do Atom

1. Criar pasta de cache do atom, caso não exista:
`sudo mkdir -p /usr/share/nginx/atom/cache`
`sudo chown -R nginx:nginx /usr/share/nginx/atom`
2. Concedida permissão 777 para a pasta a seguir:
`sudo chmod -R 777 /usr/share/nginx/atom/cache`
3. Reiniciar serviços
`systemctl restart nginx`
`systemctl restart php-fpm`
`systemctl restart memcached`
`systemctl restart elasticsearch`

4º Passo: Desativando o selinux

4. Desativado o selinux

`sudo nano /etc/sysconfig/selinux`
comentar a linha:
`SELINUX=enforcing`
e escrever abaixo dessa linha:
`SELINUX=disabled`
Salvar o arquivo e sair

Referência: <https://groups.google.com/g/ica-atom-users/c/IGCh7uLZv9E/m/MJxTuYMyAQAj>

5º Passo: Problemas relacionados à senha do banco de dados

1. Liberado acesso geral para o usuário atom:
`GRANT ALL PRIVILEGES ON atom.* TO 'atom'@'%';`
2. Alterando o usuário e adicionando o parâmetro WITH
mysql_native_password:
`ALTER USER 'atom'@%' IDENTIFIED WITH mysql_native_password BY '#Atom12345';`

6º Passo: Não encontrado biblioteca compatível no CentOS 7

1. Não encontrada biblioteca `libsaxon-java` correspondente ao linux

7º Passo: Zeramento da base de dados

1. Caso o banco e usuário não tenham sido criados, então pode-se executar os seguintes comandos:
`CUIDADO!!! Esse comando zera a base de dados`
`cd /usr/share/nginx/atom`
`php -d memory_limit=-1 symfony tools:purge`