Jagger Federation management tool: Installation guide

Following commands are copied form presentation „**Jagger Federation management tool : Installation guide**”

 # update your system and install base tools

```
yum -y update
yum -y install  git wget unzip mc telnet
```

 # install update firewall package

```
yum -y install firewalld
systemctl enable firewalld.service
systemctl start firewalld.service
firewall-cmd --permanent --add-port=22/tcp
firewall-cmd –reload
```

 # install and update Apache

```
yum -y install httpd mod_ssl mod_rewrite
systemctl enable httpd.service
systemctl start httpd.service
firewall-cmd --list-ports
firewall-cmd --permanent –add-port=80/tcp
firewall-cmd --permanent --add-port=443/tcp
firewall-cmd –reload
```

 # install and setup MariaDB

```
yum -y install mariadb-server
systemctl enable mariadb.service
systemctl start mariadb.service
mysql_secure_installation
```

This set of commands will add new package repo source, install, enable autostart and launch MariaDB. Last command is used to secure current installation by setting root account for MariaDB.

 # configure HTTPS

Update /etc/httpd/conf.d/ssl.conf file
Set proper path HTTPS cert: SSLCertificateFile
Set proper path for HTTPS key: SSLCertificateKeyFile

```
systemctl restart httpd.service
```

 # install environment tools

```
yum -y install epel-release yum-utils
yum -y install http://rpms.remirepo.net/enterprise/remi-release-7.rpm
yum-config-manager --enable remi-php73
yum -y update
yum -y install php php-common php-opcache php-mcrypt php-gd php-curl php-mysqlnd php-intl php-xml php-mbstring php-xmlrpc php-soap php-bcmath php-cli php-zip php-gearman python-pip
systemctl restart httpd.service
```

install Gearman and composer tools

```
yum -y install memcached gearmand java openssl java-11-openjdk-devel
systemctl enable gearmand.service
systemctl start gearmand
yum -y install  --disablerepo=ius composer
```

install Jagger Resource Registry

```
git clone https://github.com/Edugate/Jagger /opt/rr3
```

edit mcedit /opt/rr3/application/composer.json :

add line in “require” section "symfony/console": "*", edit line "doctrine/orm": "*",

```
cd /opt
wget -O 3.1.11.zip https://codeload.github.com/bcit-ci/CodeIgniter/zip/3.1.11
unzip 3.1.11.zip && rm 3.1.11.zip
mv CodeIgniter-3.1.11 codeigniter
cp /opt/codeigniter/index.php /opt/rr3/
```

edit mcedit /opt/rr3/index.php - update $system_path = '/opt/codeigniter/system';

update apache configuration with Jagger

add conf file /etc/httpd/conf.d/z-01-jagger.conf

```
Alias /rr3 /opt/rr3
  <Directory /opt/rr3>
       #  you may need to uncomment next line
       Require all granted

        RewriteEngine On

         RewriteCond %{HTTPS} !=on
         RewriteRule ^/?(.*) https://%{SERVER_NAME}/rr3/$1 [R,L]

        RewriteBase /rr3
        RewriteCond $1 !^(Shibboleth\.sso|index\.php|logos|signedmetadata|flags|images|app|schemas|fonts|
styles|images|js|robots\.txt|pub|includes)
        RewriteRule  ^(.*)$ /rr3/index.php?/$1 [L]
  </Directory>
  <Directory /opt/rr3/application>
        Order allow,deny
        Deny from all
  </Directory>

systemctl restart httpd.service
```

update Jagger configuration

```
cd /opt/rr3/
./install.sh
cd /opt/rr3/application/; composer update; composer install
cd /opt/rr3/application/config
cp config-default.php config.php
cp config_rr-default.php config_rr.php
cp database-default.php database.php
cp email-default.php  email.php
cp memcached-default.php memcached.php
```

initialize Jagger database

```
mysql -u root -p
```

In mysql database call following queries:
create database : create database rr3 CHARACTER SET utf8 COLLATE utf8_general_ci;
create user: grant all on rr3.* to rr3user@'localhost' identified by 'rr3pass';
apply changes : flush privileges;

edit file /opt/rr3/application/config/database.php :
$db['default']['dbdriver'] = 'mysqli';
$db['default']['username'] → set db username (eg. rr3user)
$db['default']['password'] → set db password (eg. rr3pass)
$db['default']['database'] → ser db name (eg. rr3)
$db['default']['dsn'] → update/change 'dbname=CHANGEME' (eg. dbname=rr3)

If you have enabled SELinux then call following set of commands

```
setsebool httpd_can_network_connect_db 1
chcon -t httpd_sys_rw_content_t /opt/rr3/ -R;
chcon -t httpd_sys_rw_content_t /opt/rr3/application/models/Proxies -R
or simply disable it using: setenforce 0
```

Change owner of directories

```
chown apache:apache -R /opt/rr3/ /opt/codeigniter
chown apache:apache -R /opt/rr3/application/models/Proxies
```

populate database with required tables.

```
cd /opt/rr3/application
./doctrine orm:schema-tool:create
./doctrine orm:generate-proxies
```

Update configuration /opt/rr3/application/config/config_rr.php :
$config['rr_setup_allowed'] = TRUE;

update /opt/rr3/application/config/config.php:
setup
$config['base_url'] = 'https://yourhost.example.com/rr3';
If your connection is not secured, then edit config.php : $config['cookie_secure'] = FALSE;

open: https://yourhost.example.com/rr3/setup

fill data, thus create administrative account

after creating admin jagger account
update config-rr.php : $config['rr_setup_allowed'] = FALSE;
open: https://yourhost.example.com/rr3

Jagger Resource Registry signing tool

update configuration file $config['featenable']['tasks'] = TRUE;

Used option can be configured in config_rr.php by setting $config['mq'] equal to gearman accordingly and enable selected module within same file. Set following options:
$config['mq'] = 'gearman';

$config['gearman'] = TRUE;

SAML flow require metadata to be signed. Certificate used to sign metadata should be generated. It can be done by using openssl. Self-signed certificate will be generated into xmlsectool folder.

 # Install xmlsectool

```
cd /opt/xmlsectool
openssl req -x509 -newkey rsa:4096 -keyout key.key -out cert.crt -days 3650 -subj
"/C=MY/L=City/O=NREN/OU=Federation/CN=www.federation.my"
provide password of 4 -1024 symbols length
```

Setup metadata signing tool and its environment. gearman extension will be registered in python and signed metadata output folder will be created.

```
pip install --upgrade "pip < 21.0"; pip install gearman
mkdir /opt/rr3/signedmetadata
chown -R apache:apache /opt/rr3/signedmetadata
```

!!! create and insert data to /opt/gearman-worker-metasigner.py, using provided with presentation template

Update /opt/gearman-worker-metasigner.py and update ‘cerpass’ (with passwor which was used to creation key/cert command).

Prepare gearman worker service

mcedit /etc/systemd/system/gearman-workers.service

```
[Unit]
Description=gearman-workers service
After=gearmand.service
[Service]
Type=simple
Restart=always
RestartSec=1
User=apache
ExecStart=/usr/bin/env python /opt/gearman-worker-metasigner.py

[Install]
WantedBy=multi-user.target
```

And setup

```
systemctl enable gearman-workers
systemctl start gearman-workers
systemctl status gearman-workers
```

error may occurs – user/group for apache have to be updated in /opt/gearman-worker-metasigner.py file

Metadata can be signed periodically by using internal Jagger cron tool. In order to run added in Jagger Task Scheduler jobs, jcron monitor script should be started. Before that email.php file have to be configured.

mcedit /etc/systemd/system/rr3gworker-jcronmonitor.service

```
[Unit]
Description=RR3 gworkers jcronmonitor service
After=network.target
[Service]
```

```
Type=simple
Restart=always
RestartSec=1
User=apache
ExecStart=/usr/bin/env php /opt/rr3/index.php gworkers jcronmonitor

[Install]
WantedBy=multi-user.target

systemctl enable rr3gworker-jcronmonitor.service
systemctl start rr3gworker-jcronmonitor.service
```

Now metadata can be signed manually by pressing dedicated button.

mail configuration

Mail are send periodically by using internal Jagger cron tool. In order to run added in Jagger Task Scheduler jobs, jcron mailscript should be started.

mcedit /etc/systemd/system/rr3gworker-mailqueue.service

```
[Unit]
Description=RR3 gworkers mailqueuesender service
After=network.target

[Service]
Type=simple
Restart=always
RestartSec=1
User=apache
ExecStart=/usr/bin/env php /opt/rr3/index.php gworkers mailqueuesender

[Install]
WantedBy=multi-user.target

systemctl enable rr3gworker-mailqueue.service
systemctl start rr3gworker-mailqueue.service
```