FederationStats

Federation usage statistics

> **A szócikk vagy fejezet még megírásra vár**

Federation visualization project - discountinued.

- source (ruby on rails) https://repo.niif.hu/gitweb/gitweb.cgi?p=federation-stats.git;a=summary
- live demo

Running the sample project

- Install Ruby
- Install Rails (`gem install rails`)
- Setup a `development.sqlite3` database with the `rake db:setup` command
- Fire up `script/server`, it will run the project on localhost:3000

Statistic types

Currently we have the following types of statistics:

- Unique users per day (`USER_COUNT`)
- AuthnResponse per day (`AUTH`)
- AuthnResponse per service per day (`SSO_TO_SERVICE`)

Log statistics format

The following simple format is used to convey statistics from IdPs to the central module - the white spaces (new lines) are important:

```
ENTITYID #ENTITYID#
APIKEY #API_KEY#
DATE yyyy-mm-dd

STAT #STAT_ID#
xxxx

STAT #STAT_ID#
yyyy

STAT #STAT_ID#
ww | #PEER_ENTITY_1#
zz | #PEER_ENTITY_2#
```

The following sample might help understanding the format:

```
ENTITYID https://idp.niif.hu/idp/shibboleth
APIKEY 0123.......
DATE 2009-03-18

STAT AUTH
68 logins

STAT USER_COUNT
16 unique userids

STAT SSO_TO_SERVICE
1        | urn:geant:niif.hu:niifi:sp:register.ca.niif.hu
12       | https://repo.niif.hu/shibboleth
1        | https://sandbox.aai.niif.hu/shibboleth
5        | https://sysmonitor.hbone.hu/shibboleth
10       | https://www.ki.iif.hu/shibboleth
1        | https://noc6.vh.hbone.hu/shibboleth
21       | https://webadmin.iif.hu/shibboleth
3        | https://rrd-ma.perfsonar.vh.hbone.hu/shibboleth
7        | https://ugyeletes.vh.hbone.hu/shibboleth
2        | https://noc.grid.niif.hu/shibboleth
1        | https://wiki.voip.niif.hu/shibboleth
2        | https://netmonitor.hbone.hu/shibboleth
2        | https://idp.sch.bme.hu:443/opensso/sp/test
```

Running the log statistics collector

This following script can be used the collect statistics from the idp audit logs of Shibboleth 2 IdP generated on the day before running. It is based on Peter Schober's audit_r7.py, and good for run from daily cronjob:

```
#!/bin/bash

#Config section
PARSER_COMMAND="/opt/shibboleth-idp/bin/audit_r7.py"
SOURCEDIR="/opt/shibboleth-idp/logs"
TARGETDIR="/tmp"

ENTITYID="idp-entity-id"
APIKEY="aaa..."
LOCATION2PUT="https://fedstats.example.org/import_stats"

DATE=`date -d "yesterday" +"%Y-%m-%d"`
SOURCEFILE="$SOURCEDIR/idp-audit-$DATE.log"

#Should not edit below this

if [-f $SOURCEFILE ]()
then
    LOGINS=`$PARSER_COMMAND -l $SOURCEFILE`
    UNIQUE_LOGINS=`$PARSER_COMMAND -u $SOURCEFILE`
    SERVICES=`$PARSER_COMMAND -p $SOURCEFILE | sed '/^[0-9]/p' -n`

    TARGETFILE="stat-$DATE.log"

echo "ENTITYID $ENTITYID
APIKEY $APIKEY
DATE $DATE

STAT AUTH
$LOGINS

STAT USER_COUNT
$UNIQUE_LOGINS
```

```
STAT SSO_TO_SERVICE
$SERVICES
" > $TARGETDIR/$TARGETFILE

    wget -q --no-check-certificate --post-file=$TARGETDIR/$TARGETFILE $LOCATION2PUT -O
/dev/null
    rm $TARGETDIR/$TARGETFILE
fi
```

The script below can be used the collect statistics from all the idp audit logs placed in a folder.

```
#!/bin/bash

#Config section
PARSER_COMMAND="/opt/shibboleth-idp/bin/audit_r7.py"
SOURCEDIR="/opt/shibboleth-idp/logs"
TARGETDIR="/tmp"

ENTITYID="idp-entity-id"
APIKEY="aaa..."
LOCATION2PUT="https://fedstats.example.org/import_stats"

FILES="idp-audit-*.log"

#Should not edit below this
cd $SOURCEDIR
for f in $FILES
do
  if [-f $f ]()
  then
    echo "Processing $f file..."
    DATE=${f:10:10}
    LOGINS=`$PARSER_COMMAND -l $f`
    UNIQUE_LOGINS=`$PARSER_COMMAND -u $f`
    SERVICES=`$PARSER_COMMAND -p $f | sed '/^[0-9]/p' -n`

    TARGETFILE="stat-$DATE.log"
```

```
    echo "ENTITYID $ENTITYID
APIKEY $APIKEY
DATE $DATE

STAT AUTH
$LOGINS

STAT USER_COUNT
$UNIQUE_LOGINS

STAT SSO_TO_SERVICE
$SERVICES
" > $TARGETDIR/$TARGETFILE

    wget -q --no-check-certificate --post-file=$TARGETDIR/$TARGETFILE $LOCATION2PUT -O
/dev/null
    rm $TARGETDIR/$TARGETFILE
  fi
done
```

Feeding the database with the statistics

The federation statistics rails project contains the `script/stat_parser/file.rb` command which can process the statistics format and load the data to the database. Note that this script currently contains an absolute path for the `script/runner` script, so you must fix this before use.

Using HTTP-Post to feed the database

When deployed, the rails project provides a `/import_stats` URL to which one could POST the generated statistics file.

Creating IdPs

Use the rails console to create new idps:

```
$ RAILS_ENV=production script/console

>> Entity.create :name => 'foo', :entity_type => 'idp'

=> #<Entity id: 1, name: "foo", entity_type: "idp", created_at: "2010-11-29 14:55:40",
updated_at: "2010-11-29 14:55:40", api_key: "da9l233a45698fa5c4a252e301e3da2sf5ece24e">
```

Változat #2
cziernorbert hozta létre 2026-04-14 13:22:11 CEST
cziernorbert frissítette 2026-04-17 13:00:58 CEST