

# Grouper CSV Provisioner

|                           |   |                                |  |   |  |
|---------------------------|---|--------------------------------|--|---|--|
| <a href="#">Wiki Home</a> | <a href="#">Grouper Release Announcements</a> | <a href="#">Grouper Guides</a> | <a href="#">Grouper Deployment Guide</a> | <a href="#">Community Contributions</a> | <a href="#">Internal Developer Resources</a> |
|---------------------------|---|--------------------------------|--|---|--|

Generate CSV file to be SFTP'ed to a server. These are scheduled by quartz cron.

To SFTP setup a site using [this doc](#):

## Configure

grouper-loader.properties

```
#####
## CSV files
## "csvJobId" is the key of the config, change that for your csv report
#####

# set this to enable the instrumentation
# {valueType: "class", mustExtendClass: "edu.internet2.middleware.grouper.app.loader.OtherJobBase"}
otherJob.csvJobId.class = edu.internet2.middleware.grouper.app.reports.GrouperCsvReportJob

# cron string
# {valueType: "string"}
otherJob.csvJobId.quartzCron = 0 21 7 * * ?

# query to run
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.query$"}
otherJob.csvJobId.csvReport.query = select USER_ID, USER_NAME, EMAIL_ADDRESS, AUTH_TYPE, TITLE, DEPARTMENT,
CUSTOM_STRING, DAY_PASS, CUSTOM_STRING2, GROUPS from some_view

# database to hit
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.database$"}
otherJob.csvJobId.csvReport.database = pennCommunity

# remove underscores and capitalize headers, go from USER_NAME to UserName
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.removeUnderscoresAndCapitalizeHeaders$"}
otherJob.csvJobId.csvReport.removeUnderscoresAndCapitalizeHeaders = false

# fileName, e.g. myFile.csv or /opt/whatever/myFile.csv. If blank will create a name
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.database$"}
otherJob.csvJobId.csvReport.fileName = MyFile.csv

# sftp config id (from grouper.properties) if sftp'ing this file somewhere, otherwise blank
# https://spaces.at.internet2.edu/display/Grouper/Grouper+Sftp+files
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.sftp\\.\\.configId$"}
otherJob.csvJobId.csvReport.sftp.configId = someSftpServer

# remote file to sftp to if sftp'ing
# {valueType: "string", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.sftp\\.\\.fileNameRemote$"}
otherJob.csvJobId.csvReport.sftp.fileNameRemote = /data01/whatever/MyFile.csv

# if the file should be deleted from the grouper daemon server after sending it
# {valueType: "boolean", regex: "^otherJob\\.([^.]+)\\.\\.csvReport\\.\\.deleteFile$"}
# otherJob.csvJobId.csvReport.deleteFile = true
```

## Logging

log4j.properties

```
log4j.logger.edu.internet2.middleware.grouper.app.reports.GrouperCsvReportJob = DEBUG
```

#### Sample log message

```
2019-11-23 21:18:20,119: [main] DEBUG GrouperCsvReportJob.run(140) - - job: csv, database: pennCommunity,  
query: select USER_ID, USER_NAME, EMAIL_ADDRESS, AUTH_TYPE, TITLE, DEPARTMENT, CUSTOM_STRING, DAY_PASS,  
CUSTOM_STRING2, GROUPS from some_view, columnsSize: 10, rowsSize: 39193, file: C:  
\Users\mchyzer\AppData\Local\Temp\PennUsers.csv, fileSizeBytes: 3450123, sftpConfigId: depot, fileNameRemote:  
/data01/dir/MyFile.csv, tookMillis: 219178
```

## Using the API

```
OtherJobInput otherJobInput = new OtherJobInput();  
otherJobInput.setJobName("myCsvJob");  
new GrouperCsvReportJob().run(otherJobInput);
```

This is in grouper\_v2\_4\_0\_api\_patch\_81 and newer.