SWAMP API: curl examples

Table of Contents:

Short link: https://goo.gl/pMonMY

Setup

```
# beware exposing your password on a shared system
export SWAMPUSER=username
export SWAMPPASS=password
```

```
# create private files
umask 0077
# if using it.cosalab.org
export RWS=swa-rws-it-01.cosalab.org
export CSA=swa-csaweb-it-01.cosalab.org
# if using www.mir-swamp.org
export RWS=swa-rws-pd-01.mir-swamp.org
export CSA=swa-csaweb-pd-01.mir-swamp.org
```

Log in

```
# log in to RWS
curl -f -c rws-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"username\":\"$SWAMPUSER\",\"password\":\"$SWAMPPASS\"}" \
  https://$RWS/login > rws-userinfo.txt
# log in to CSA
curl -f -c csa-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"username\":\"$SWAMPUSER\",\"password\":\"$SWAMPPASS\"}" \
  https://$CSA/login > csa-userinfo.txt
# find my user UUID
perl -n -e 'print $1 if (/\"user_uid\":\"([\w-]+)\"/);' \
  < rws-userinfo.txt > swamp-user-uuid.txt
export SWAMP_USER_UUID=`cat swamp-user-uuid.txt`
# look up additional user info (email address, etc.)
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  https://$RWS/users/$SWAMP_USER_UUID > swamp-user-details.txt
```

List Projects

```
# find "MyProject"
# get my project memberships
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  https://$RWS/users/$SWAMP_USER_UUID/projects/trial \
  > swamp-myproject.txt
# get UUID for "MyProject"
perl -n -e 'print $1 if (/\"project_uid\":\"([\w-]+)\"/);' \
  < swamp-myproject.txt > swamp-project-uuid.txt
export SWAMP_PROJECT_UUID=`cat swamp-project-uuid.txt`
# get my other project memberships (if any)
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  https://$RWS/users/$SWAMP_USER_UUID/projects > swamp-projects.txt
```

List Packages

```
# list public packages (including UUIDs)
curl -f https://$CSA/packages/public > swamp-public-packages.txt
# list packages shared with my project
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/packages/protected/$SWAMP_PROJECT_UUID > \
  swamp-protected-packages.txt
# list package types (C/C++, Java Source Code, etc.)
curl -f https://$CSA/packages/types > swamp-package-types.txt
```

List Tools

```
# list public tools (including UUIDs)
curl -f https://$CSA/tools/public > swamp-public-tools.txt
# list restricted tools (which require accepting a license)
curl -f https://$CSA/tools/restricted > swamp-restricted-tools.txt
```

List Platforms

```
# list public platforms (including UUIDs)
curl -f https://$CSA/platforms/public > swamp-public-platforms.txt
```

List Assessments

```
# list assessments (including UUIDs) per project
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/projects/$SWAMP_PROJECT_UUID/assessment_runs \
  > swamp-assessments.txt
```

List Assessment Results

```
# list assessment results per project
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/projects/$SWAMP_PROJECT_UUID/assessment_results \
  > swamp-assessment-results.txt
perl -n -l -e 'print "$1" while (/\"assessment_result_uuid\":\"([\w-]+)\"/g);' \
  < swamp-assessment-results.txt > swamp-assessment-result-uuids.txt
```

Get SCARF Results

```
# get scarf results for first assessment_result_uuid
export SWAMP_ASSESSMENT_RESULT_UUID=`head -1 swamp-assessment-result-uuids.txt`
curl -f -H "Accept-Encoding: gzip" -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/v1/assessment_results/$SWAMP_ASSESSMENT_RESULT_UUID/scarf \
  > scarf.xml
```

Upload Package

```
# upload my package tarball
```

```
# using hello.tar.gz from https://uofi.box.com/swamp-c-hello
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -X POST \
  -F file=@hello.tar.gz \
  -F user_uid=$SWAMP_USER_UUID \
  https://$CSA/packages/versions/upload > swamp-uploaded-file.txt
# get the destination path UUID
perl -n -e 'print $1 if (/\"destination_path\":\"([\w-]+)\"/);' \
  < swamp-uploaded-file.txt > swamp-dest-path.txt
export SWAMP_DEST_PATH=`cat swamp-dest-path.txt`
# choose my package name
export SWAMP_PACKAGE_NAME=basney-test-23432153
# create the package
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"package_sharing_status\":\"private\",\
      \"name\":\"$SWAMP_PACKAGE_NAME\",\
      \"description\":\"\",\
      \"external_url\":\"\",\
      \"package_type_id\":1}" \
  https://$CSA/packages > swamp-package.txt
# get the package UUID
perl -n -e 'print $1 if (/\"package_uuid\":\"([\w-]+)\"/);' \
  < swamp-package.txt > swamp-package-uuid.txt
export SWAMP_PACKAGE_UUID=`cat swamp-package-uuid.txt`
# create the package version
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"version_string\":\"1.0\", \
      \"version_sharing_status\":\"protected\", \
      \"package_uuid\":\"$SWAMP_PACKAGE_UUID\", \
```

```
        \"notes\":\"\", \
        \"source_path\":\"hello/\", \
        \"config_dir\":\"\", \
        \"config_cmd\":\"\", \
        \"config_opt\":\"\", \
        \"build_file\":\"\", \
        \"build_system\":\"make\", \
        \"build_target\":\"\", \
        \"build_dir\":\"\", \
        \"build_opt\":\"\", \
        \"package_path\":\"$SWAMP_DEST_PATH/hello.tar.gz\"}" \
  https://$CSA/packages/versions/store > swamp-pkgver.txt
# get package version UUID
perl -n -e 'print $1 if (/\"package_version_uuid\":\"([\w-]+)\"/);' \
  < swamp-pkgver.txt > swamp-pkgver-uuid.txt
export SWAMP_PKGVER_UUID=`cat swamp-pkgver-uuid.txt`
# share package version with $SWAMP_PROJECT_UUID
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -X PUT \
  -d "projects[0][project_uid]=$SWAMP_PROJECT_UUID" \
  https://$CSA/packages/versions/$SWAMP_PKGVER_UUID/sharing
```

Download Package

```
# download and untar my package tarball
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/packages/versions/$SWAMP_PKGVER_UUID/download |
  tar xvz
```

Submit Java Assessment Run

```
# choose your package, tool, and project (see above)
```

```
# no need to choose platform for Java...
export SWAMP_PACKAGE_UUID=...
export SWAMP_TOOL_UUID=...
export SWAMP_PROJECT_UUID=...
# for example, to assess Twitter4j using Findbugs
perl -n -e 'print $1 if \
  (/{\"package_uuid\":\"([\w-]+)\",\"name\":\"Twitter4j\",/);' \
  < swamp-public-packages.txt > swamp-twitter4j-uuid.txt
export SWAMP_PACKAGE_UUID=`cat swamp-twitter4j-uuid.txt`
perl -n -e 'print $1 if \
  (/{\"tool_uuid\":\"([\w-]+)\",\"name\":\"Findbugs\",/);' < \
  swamp-public-tools.txt > swamp-findbugs-uuid.txt
export SWAMP_TOOL_UUID=`cat swamp-findbugs-uuid.txt`
# create the A-Run
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"project_uuid\":\"$SWAMP_PROJECT_UUID\",\
      \"package_uuid\":\"$SWAMP_PACKAGE_UUID\",\
      \"tool_uuid\":\"$SWAMP_TOOL_UUID\"}" \
  https://$CSA/assessment_runs > swamp-a-run.txt
# get the A-Run UUID
perl -n -e 'print $1 if (/\"assessment_run_uuid\":\"([\w-]+)\"/);' \
  < swamp-a-run.txt > swamp-a-run-uuid.txt
export SWAMP_ARUN_UUID=`cat swamp-a-run-uuid.txt`
# schedule the A-Run
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -X POST \
  -d "notify-when-complete=true" \
  -d "assessment-run-uuids[]=$SWAMP_ARUN_UUID" \
  https://$CSA/run_requests/one-time > swamp-a-run-request.txt
```

Submit C Assessment Run

```
# choose your package, tool, platform and project (see above)
export SWAMP_PACKAGE_UUID=...
export SWAMP_TOOL_UUID=...
export SWAMP_PLATFORM_UUID=...
export SWAMP_PROJECT_UUID=...
# for example, to assess Nagios using cppcheck on Fedora
perl -n -e 'print $1 if \
   (/{\"package_uuid\":\"([\w-]+)\",\"name\":\"Nagios\",/);' \
  < swamp-public-packages.txt > swamp-nagios-uuid.txt
export SWAMP_PACKAGE_UUID=`cat swamp-nagios-uuid.txt`
perl -n -e 'print $1 if \
   (/{\"tool_uuid\":\"([\w-]+)\",\"name\":\"cppcheck\",/);' < \
  swamp-public-tools.txt > swamp-findbugs-uuid.txt
export SWAMP_TOOL_UUID=`cat swamp-findbugs-uuid.txt`
perl -n -e 'print $1 if \
   (/{\"platform_uuid\":\"([\w-]+)\",\"name\":\"Fedora Linux\",/);' \
  < swamp-public-platforms.txt > swamp-fedora-uuid.txt
export SWAMP_PLATFORM_UUID=`cat swamp-fedora-uuid.txt`
# create the A-Run
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/json; charset=UTF-8" \
  -X POST \
  -d "{\"project_uuid\":\"$SWAMP_PROJECT_UUID\",\
       \"package_uuid\":\"$SWAMP_PACKAGE_UUID\",\
       \"platform_uuid\":\"$SWAMP_PLATFORM_UUID\",\
       \"tool_uuid\":\"$SWAMP_TOOL_UUID\"}" \
  https://$CSA/assessment_runs > swamp-a-run.txt
# get the A-Run UUID
perl -n -e 'print $1 if (/\"assessment_run_uuid\":\"([\w-]+)\"/);' \
  < swamp-a-run.txt > swamp-a-run-uuid.txt
export SWAMP_ARUN_UUID=`cat swamp-a-run-uuid.txt`
```

```
# schedule the A-Run
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -X POST \
  -d "notify-when-complete=true" \
  -d "assessment-run-uuids[]=$SWAMP_ARUN_UUID" \
  https://$CSA/run_requests/one-time > swamp-a-run-request.txt
```

Submit Multiple Runs

```
# schedule the A-Runs, one per line
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -H "Content-Type: application/x-www-form-urlencoded" \
  -X POST \
  -d "notify-when-complete=true" \
  -d "assessment-run-uuids[]=$SWAMP_ARUN_UUID1" \
  -d "assessment-run-uuids[]=$SWAMP_ARUN_UUID2" \
  -d "assessment-run-uuids[]=$SWAMP_ARUN_UUID3" \
  https://$CSA/run_requests/one-time > swamp-a-run-request.txt
```

Get Run Status

```
# view the most recent execution record(s) for my project
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/projects/$SWAMP_PROJECT_UUID/execution_records?limit=1 > swamp-exec-record.txt
# get the execution record UUID
perl -n -e 'print $1 if (/\"execution_record_uuid\":\"([\w-]+)\"/);' \
  < swamp-exec-record.txt > swamp-exec-record-uuid.txt
export SWAMP_EXEC_UUID=`cat swamp-exec-record-uuid.txt`
# get the execution record directly
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/execution_records/$SWAMP_EXEC_UUID > swamp-exec-record.txt
# get the package UUID
```

```
perl -n -e 'print $1 if (/\"package_uuid\":\"([\w-]+)\"/);' \
  < swamp-exec-record.txt > swamp-package-uuid.txt
export SWAMP_PACKAGE_UUID=`cat swamp-package-uuid.txt`
# get package info
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/packages/$SWAMP_PACKAGE_UUID > swamp-package-info.txt
# get the package version
perl -n -e 'print $1 if (/\"package_version_uuid\":\"([\w-]+)\"/);' \
  < swamp-exec-record.txt > swamp-pkgver-uuid.txt
export SWAMP_PKGVER_UUID=`cat swamp-pkgver-uuid.txt`
# download and untar my package tarball
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/packages/versions/$SWAMP_PKGVER_UUID/download |
  tar xvz
```

Get Run Status

```
# view execution records for my project
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/projects/$SWAMP_PROJECT_UUID/execution_records?limit=5
```

Log Out

```
# clear out environment variables
unset SWAMPUSER SWAMPPASS
# log out of RWS
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  -X POST \
  -d "" \
  https://$RWS/logout
# log out of CSA
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -X POST \
```

```
  -d "" \
  https://$CSA/logout
# remove cookie jars
rm -f csa-cookie-jar.txt rws-cookie-jar.txt
```

Error Checking

```
# 'curl -f' sets non-zero exit status ($?) on error
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  https://$RWS/users/nobody
echo $?
```

Delete Package

```
# delete a package
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  -X DELETE \
  https://$CSA/packages/$SWAMP_PACKAGE_UUID
```

Get Current Logged In User

```
# get current logged in user for CSA
curl -f -b csa-cookie-jar.txt -c csa-cookie-jar.txt \
  https://$CSA/users/current
# get current logged in user for RWS
curl -f -b rws-cookie-jar.txt -c rws-cookie-jar.txt \
  https://$RWS/users/current
```