

Table of Contents

What is esgf-getCredentials?	2
Getting started	2
Pre-requisites	2
Download	2
Run it	2
Command line UI Guide	2
Graphic UI Guide	3
Setting user	3
Setting output files	4
Retrieve credentials	4
Advanced options	4
Developers Guide	5
Github	5
Architecture	5

What is esgf-getCredentials?

A tool to retrieve user credentials from ESGF. It have one graphic interface and another command line interface.

Getting started

Pre-requisites

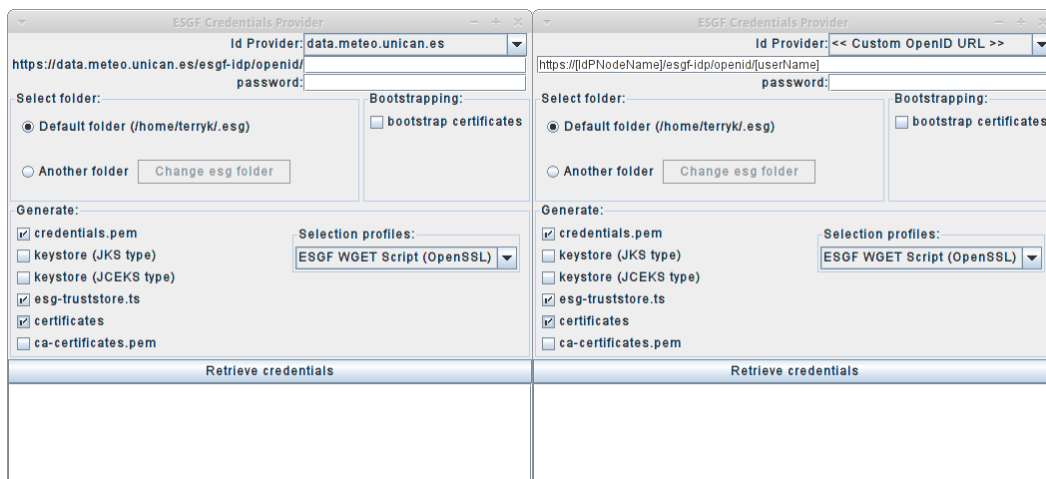
JDK or OpenJDK 6 and upper versions

Download

Download the jar -> [getESGFCredentials-0.1.jar](#) 476.4 KB new

[Other versions..](#)

Run it



Go to download folder:

- In Windows:
 - Open ESGFToolsUI-v0.8.jar
- Command-line interpreter:

```
java -jar ESGFToolsUI-v0.8.jar
```

Command line UI Guide

1. Command line help

```
$ java -jar getESGFCredentials-0.0.2.jar --help
esgf-getcredentials.

Usage:
esgf-getcredentials
esgf-getcredentials (-o <openid> | --openid <openid>) [options]
esgf-getcredentials (-h | --help)
esgf-getcredentials --version
Options:
```

```

-o <openid> --openid <openid> OpenID endpoint from where myproxy information can be gathered.
-p --password <password> OpenID passphrase.
--output <path> Path of folder where the retrieved certificates will be stored[default: /home/terryk/.es
-w --writeall Generate all credentials files. The files generated are the same files generated with op
-b --bootstrap To bootstrapping certificates in myproxy service.
--credentials Write user certificate and private key in pem format.
--cacertspem Write trust CA certificates in pem format.
--cacertsjks Write trust CA certificates in JKS keystore format.
--cacerts Write trust CA certificates in a folder.
--keystorejks Write JKS keystore file. This keystore contains certificate, certificate chain and priva
--keystorejceks Write JCEKS keystore file. This keystore contains certificate, certificate chain and pr
-d --debug Turn debugging info on.
-h --help Show this screen.
--version Show version.

```

1. ESGF WGET Script case

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --credentials --cacerts --cacertsjks
```

1. ESGF WGET Script case (GNU TLS)

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --credentials --cacertspem --cacertsjks
```

1. ToolsUI case

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --keystorejks --cacertsjks
```

1. Aria2 case

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --credentials --cacertspem
```

1. ESGFToolsUI case

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --credentials
```

1. NETCDF DAP case

```
java -jar getESGFCredentials-0.0.2.jar -o <openid> -p <password> --credentials --cacerts --cacertspem --output <path>
```

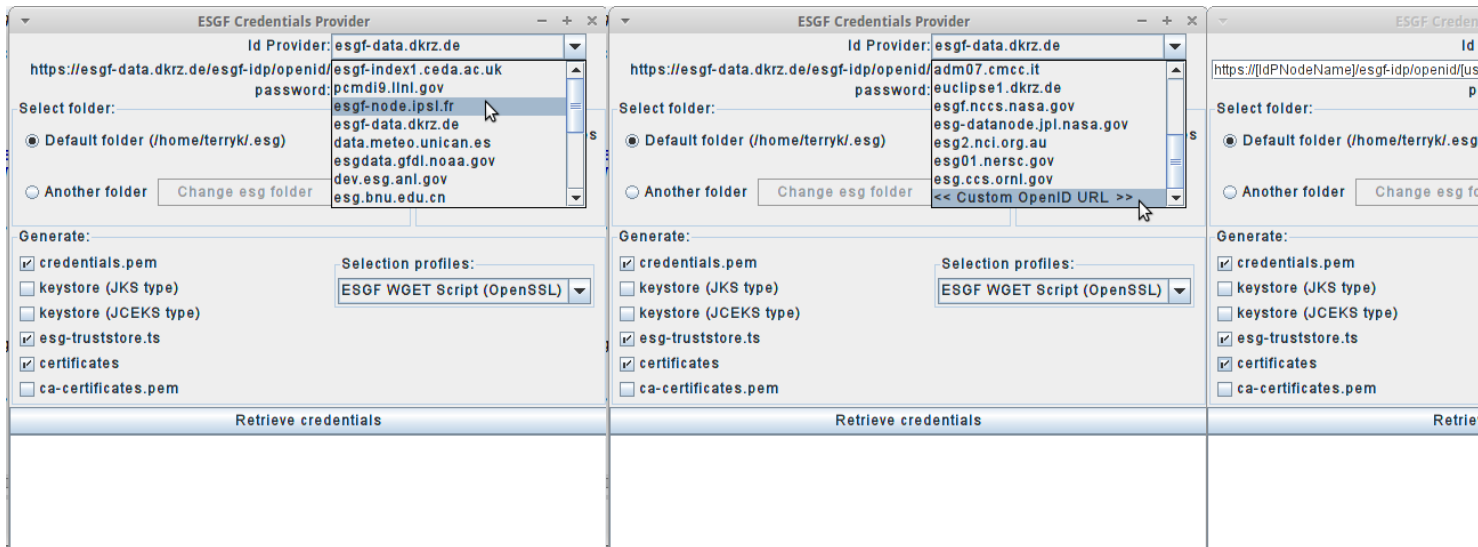
Graphic UI Guide

- In Windows:
 - Open ESGFToolsUI-v0.8.jar
- Command-line interpreter:

```
java -jar ESGFToolsUI-v0.8.jar
```

Setting user

You can select your IdP provider in the top drop-down list. If your IdP provider isn't in the list of providers. Select "Custom OpenID URL", with this option the GUI interface change to be able write OpenID URL's



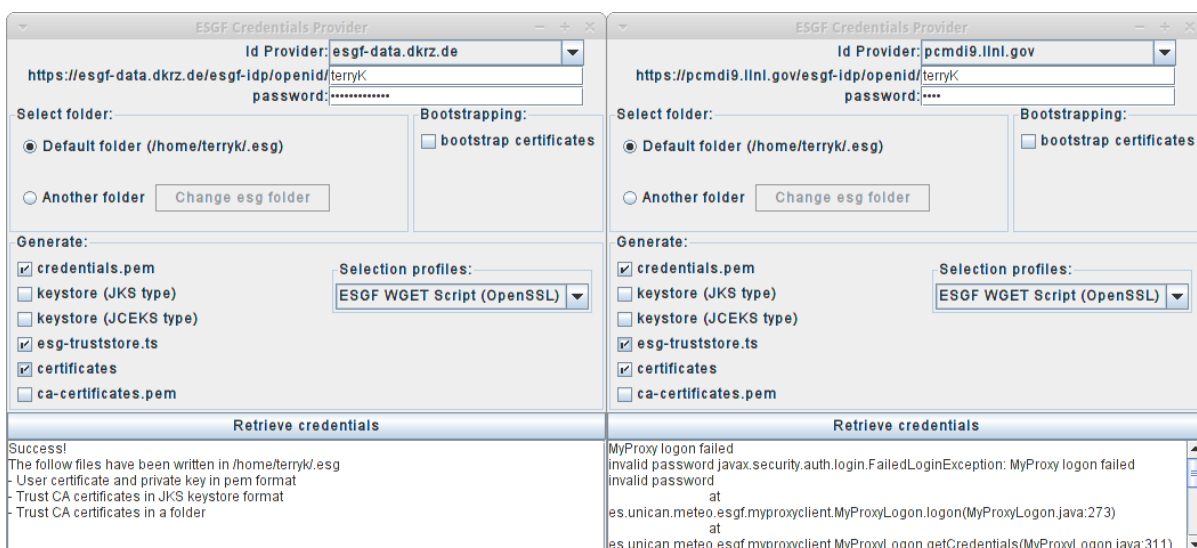
Setting output files

You can select in "Generate" section what output files will be generated in the output folder.

credentials.pem	It's a pem file that contains the x509 user certificate and the RSA private key
keystore (JKS type)	It's a keystore in format JKS which is build with user cert, cert chain and private key
keystore (JCEKS type)	It's a keystore in format JCEKS which is build with user cert, cert chain and private key
esgf-truststore.ts	CA's certificates in keystore in format JKS
certificates	CA's certificate files and policy files in a folder
ca-certificates.pem	CA's certificates in pem format

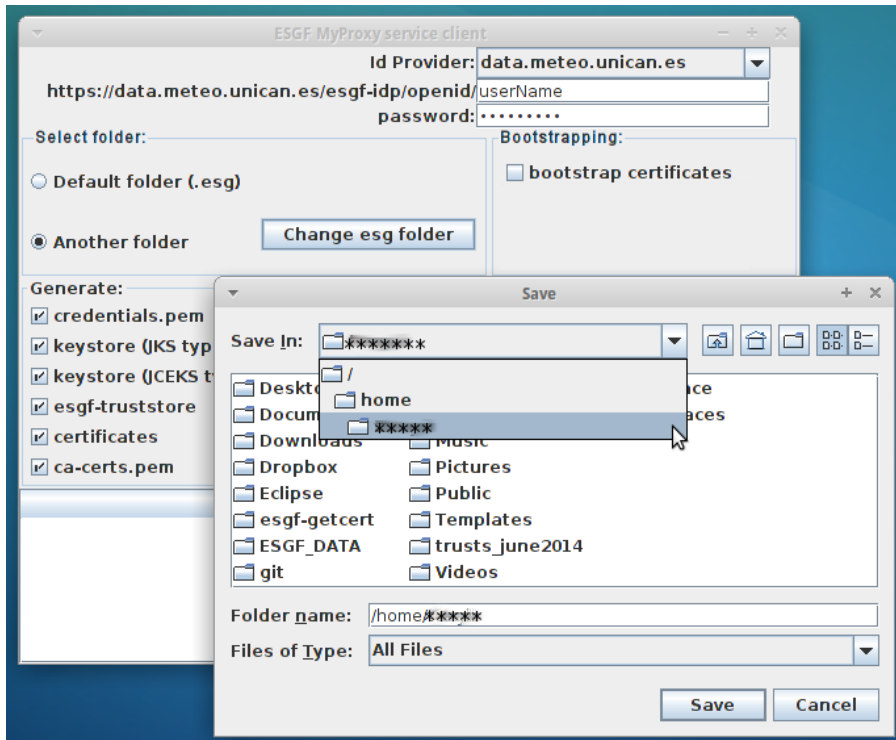
Retrieve credentials

Click on "retrieve credentials" button. If all goes well a success message is shown. However, if some error happens then the Exception is showed



Advanced options

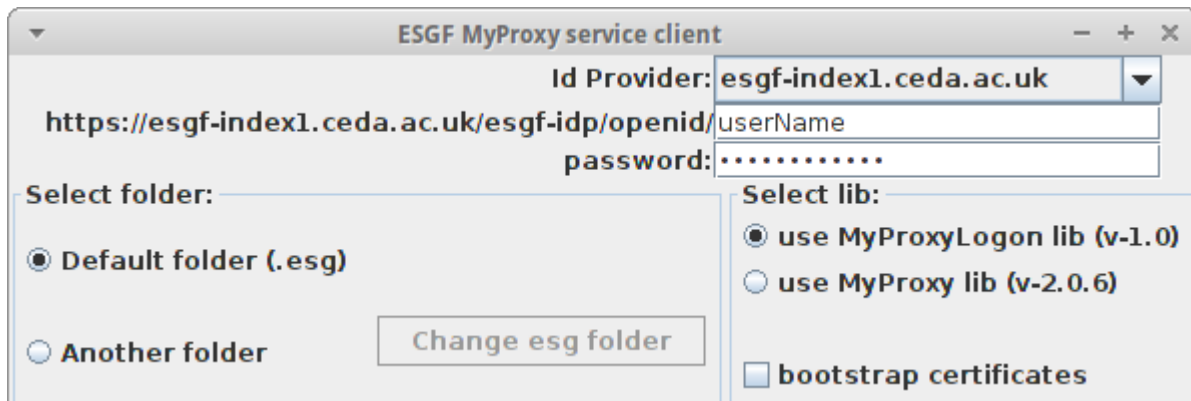
1. You can bootstrap the certificates. For that, select the check box "bootstrap certificates" in "Select Lib" section
1. You can change the output folder. The default is \$USER_HOME/.esg



1. You can download a multilib myproxy version to select it in the "Select Lib" section

- MyProxyLogon lib v1.0
- MyProxy lib v2.0.6

Multilib jar -> [getESGFCredentialsMultLib-0.1.jar](#) 2.3 MB new



Developers Guide

Github

<https://github.com/SantanderMetGroup/esgf-getcredentials>

Architecture