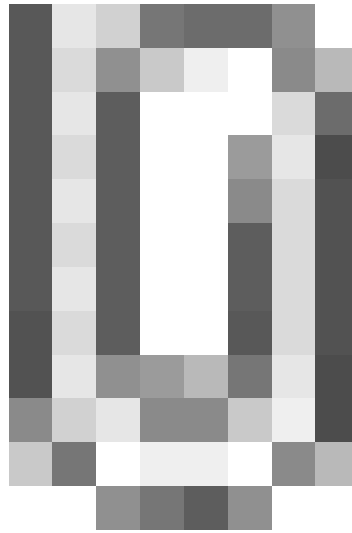
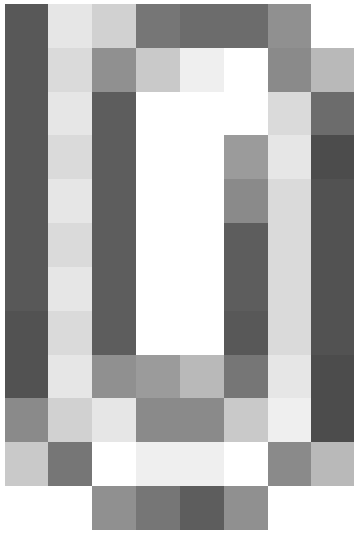


## Table of Contents

<b>What is esgf-getCredentials?</b>	<b>2</b>
<b>Getting started</b>	<b>2</b>
Pre-requisites	2
Download	2
Command line User's Manual	2
Graphic UI User's Manual	3
Setting user	4
Setting output files	5
Advanced options	5
Retrieve credentials	5
Developers Guide	6
Github	6
Architecture	6

## What is esgf-getCredentials?



## Getting started

### Pre-requisites

JDK or OpenJDK 6 and upper versions

### Download

- A) Standard jar -> [getESGFCredentials-0.1.jar](#) 476.4 KB new  
 B) MultiLib jar -> [getESGFCredentialsMultLib-0.1.jar](#) 2.3 MB new  
[Other versions..](#)

## Command line User's Manual

### 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/.esgf]
-w --writeall                  Generate all credentials files. The files generated are the same files generated with option --writeall
-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 private key.
--keystorejceks                Write JCEKS keystore file. This keystore contains certificate, certificate chain and private key.
-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 --cacertsjks
```

1. ESGFToolsUI case

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

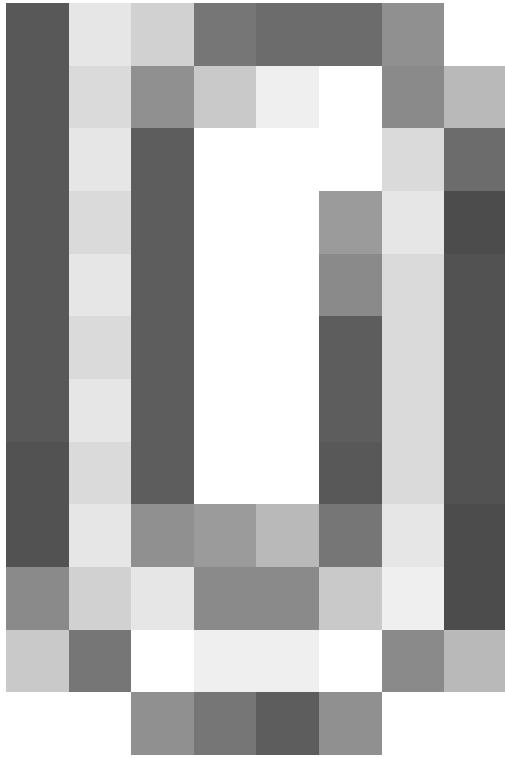
1. NETCDF DAP case

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

## Graphic UI User's Manual

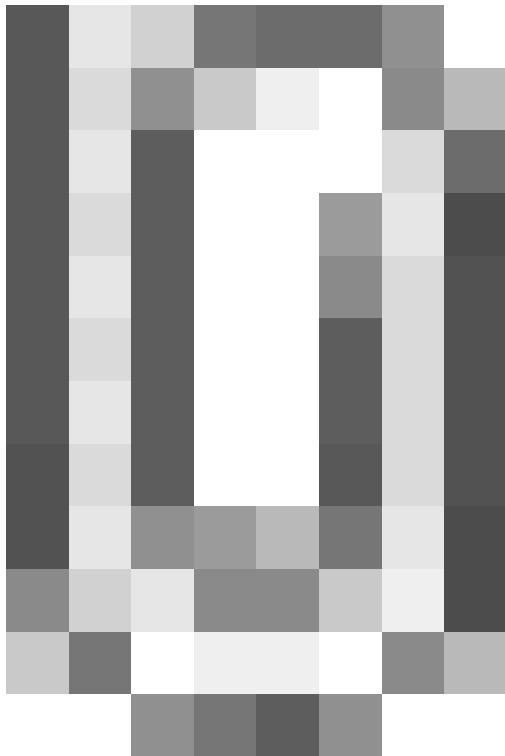
- In Windows:
  - Open getESGFCredentials-0.0.2.jar
- In Linux
  - In shell:

```
java -jar getESGFCredentials-0.0.2.jar
```

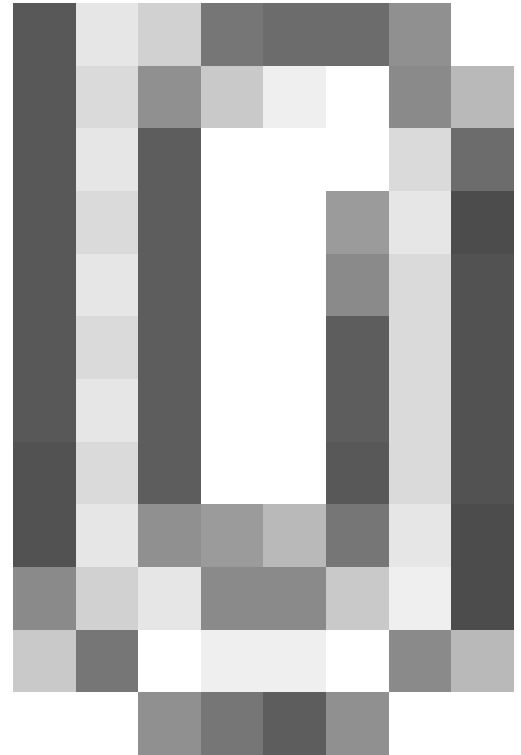
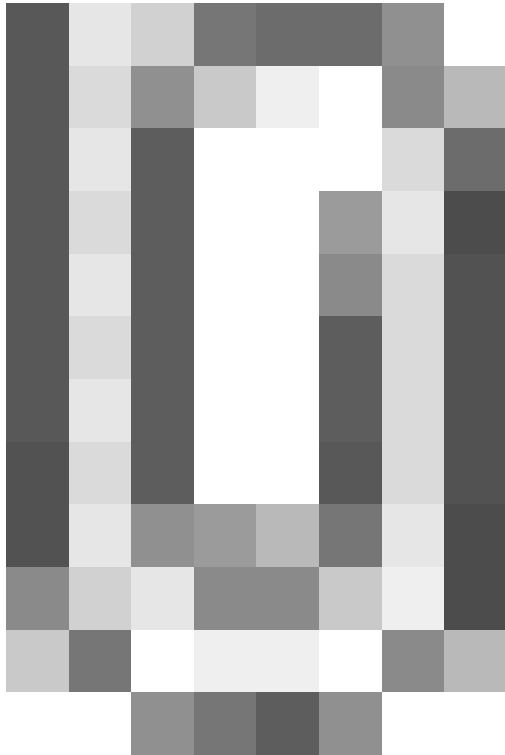


### Setting user

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 [BR?](#)



**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 x509 certificates, RSA keys and CA's certificates
keystore (JCEKS type)	It's a keystore in format JCEKS which is build with x509 certificates, RSA keys and CA's certificates
esgf-truststore	It's a keystore in format JKS which is build with CA's certificates
certificates	It's a directory with CA's certificate files and policy files
ca-certs.pem	It's a pem file with all CA's certificates

**Advanced options**

1. You can select the myproxy client library in the "Select Lib" section

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

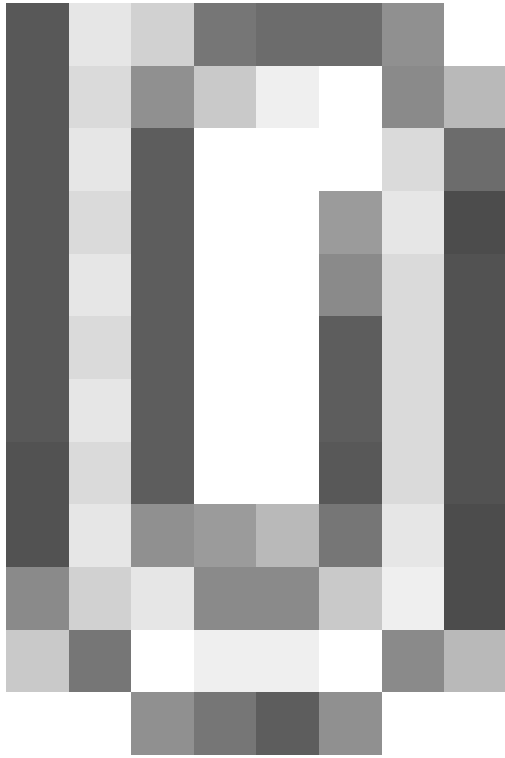
1. You can bootstrap the certificates. For that, select the check box "bootstrap certificates" in "Select Lib" section

1. You can change the output folder. The default is \$USER\_HOME/.esg



**Retrieve credentials**

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



## **Developers Guide**

### **Github**

[?https://github.com/SantanderMetGroup/esgf-getcredentials](https://github.com/SantanderMetGroup/esgf-getcredentials)

### **Architecture**