


```

Using solr_logs_dir=/esg/solr-logs
Using esg_dist_url=http://esg-dn2.nsc.liu.se/esgf/dist
sudo: source: command not found
Sending stop command to Solr running on port 8983 ... waiting 5 seconds to allow Jetty process 16339 to stop gracefully.
Sending stop command to Solr running on port 8984 ... waiting 5 seconds to allow Jetty process 16554 to stop gracefully.
Stopping Globus Services for Data-Node... (GridFTP) stop_globus_services for datanode
globus-gridftp-server: unrecognized service
Stopping Globus Services for Index-Node... (MyProxy server) stop_globus_services for gateway
Stopping myproxy-server: [ OK ]
No MyProxy Process Currently Running...
Tomcat (jsvc) process is running...

stop tomcat: /usr/local/tomcat/bin/jsvc -pidfile /var/run/tomcat-jsvc.pid -stop org.apache.catalina.startup.Bootstrap
(please wait)
postmaster (pid 16024) is running...
Stopping postgresql service: [ OK ]
Stopping httpd: [ OK ]
Running shutdown hooks...

-----
Running Node Services...
node type: [ data index idp compute ] (60)
-----
-----

```

The following command will recursively delete every thredds_data_root defined in esg.ini. umount the data before removing the ESGF node.

Execute `source /usr/local/bin/esg-purge.sh && esg-purge all`

2. Installation from scratch

Change directory to `/usr/local/bin/`

```
[root@spock ~]# cd /usr/local/bin/
```

```

[root@spock bin]# wget -O esg-bootstrap http://distrib-coffee.ipsl.jussieu.fr/pub/esgf/dist/devel/esgf-installer/2.5/esg-b
[root@spock bin]# chmod 555 ./esg-bootstrap
[root@spock bin]# ./esg-bootstrap

```

Your directory should look like this:

```

[root@spock bin]# ls
esg-bootstrap esg-functions esg-init esg-node esg-purge.sh jar_security_scan setup-autoinstall

```

Check your node's version:

```

[root@spock bin]# ./esg-node --version
[VERIFIED]
Version: v2.5.9-master-release
Release: Midgard
Earth Systems Grid Federation (http://esgf.llnl.gov)
ESGF Node Installation Script

```

Set node's type:

```

[root@localhost bin]# ./esg-node --set-type data index
node type set to: [ data index ] (12)

```

Install the node:

```
[root@localhost bin]# ./esg-node --install
Please select the ESGF distribution mirror for this installation (fastest to slowest):
-----
[1] http://dist.ceda.ac.uk/esgf
[2] http://distrib-coffee.ipsl.jussieu.fr/pub/esgf
[3] http://esg-dn2.nsc.liu.se/esgf
[4] http://aims1.llnl.gov/esgf
-----
select [1] >
```

```
What is the fully qualified domain name of this node? [localhost.localdomain]: spock.meteo.unican.es
What is the admin password to use for this installation? (alpha-numeric only) []:
Please re-enter password:
What is the name of your organization? [localhost]: Unican
Please give this node a "short" name: []: Unican
Please give this node a more descriptive "long" name []: Unican
What is the namespace to use for this node? (set to your reverse fqdn - Ex: "gov.llnl") []: es.unican.meteo.spock
What peer group(s) will this node participate in? (esgf-test|esgf-prod) [esgf-test]:
What is the default peer to this node? [spock.meteo.unican.es]: VESGINT-IDX.IPSL.UPMC.FR
What is the hostname of the node do you plan to publish to? [VESGINT-IDX.IPSL.UPMC.FR]:
What email address should notifications be sent as? []: YOUR-EMAIL
Is the database external to this node? [y/N]:
What is the database connection string? [postgresql://dbsuper@localhost:5432/esgct]: postgresql://
What is the (low priv) db account for publisher? [esgct]:
What is the db password for publisher user (esgct)? []:
```

```
Starting Postgress...
Starting postgresql service: [ OK ]
0 S postgres 5614 1 6 80 0 - 53982 poll_s 17:52 ? 00:00:00 /usr/bin/postmaster -p 5432 -D /var/lib/pgsql/d
1 S postgres 5631 5614 0 80 0 - 44735 poll_s 17:52 ? 00:00:00 postgres: logger process
1 S postgres 5633 5614 0 80 0 - 53982 poll_s 17:52 ? 00:00:00 postgres: writer process
1 S postgres 5634 5614 0 80 0 - 53982 poll_s 17:52 ? 00:00:00 postgres: wal writer process
1 S postgres 5635 5614 0 80 0 - 54015 poll_s 17:52 ? 00:00:00 postgres: autovacuum launcher process
1 S postgres 5636 5614 0 80 0 - 44734 poll_s 17:52 ? 00:00:00 postgres: stats collector process
Enter password for postgres user dbsuper:
Re-enter password for postgres user dbsuper:

Please Enter PostgreSQL port number [5432]:>
```

```
Would you like a "system" or "user" publisher configuration:
-----
*[1] : System
[2] : User
-----
[C] : (Custom)
-----
select [1] >

You have selected: 1
Publisher configuration file -> [/esg/config/esgct/esg.ini]

Is this correct? [Y/n]

Your publisher configuration file will be: /esg/config/esgct/esg.ini
What is your organization's id? [Unican]:
```

```
Would you like to configure this node for CMIP6 publishing (additional project dependencies will be installed)? [y/N]
```

```
[VERIFIED]
Looking for keystore [/esg/config/tomcat/keystore-tomcat]... (don't see one)...
Keystore setup:
Launching Java's keytool:
store_password = *****
Would you like to use the DN: (OU=ESGF.ORG, O=ESGF) ? [Y/n]:

Please enter the password for this keystore :
Please re-enter the password for this keystore:
```

```
Enter a single ip address which would be cleared to access admin restricted pages.
You will be prompted if you want to enter more ip-addresses

Do you wish to allow further ips? y/n
n
```

```
[VERIFIED]
Create user credentials
Please enter username for tomcat [dnode_user]:
Please enter password for user, "dnode_user" [*****]: 653e78d101f9105fd65755249edf849411e70657814acf06667d910709f3eaa
Would you like to add another user? [y/N]:
```

```
Please Enter the public (i.e. routable) IP address of this host [10.0.2.15]:> YOUR-PUBLIC-ROUTABLE-IP

Using IP: 10.0.2.15
Do you wish to use an external IDP peer?(N/y):y
Please specify your IDP peer node's FQDN:VESGINT-IDX.IPSL.UPMC.FR\

Server sent 2 certificate(s):

1 Subject CN=vesgint-idx.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  Issuer CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  sha1 8f 1c 62 12 3a 3f 88 be 12 26 c8 f8 f9 3b da 73 a7 f6 f2 04
  md5 1f fb bd cd 25 cd f5 9d 39 42 d4 c3 ef 2d 98 20

2 Subject CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  Issuer CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  sha1 15 1d a1 c3 b9 0d 9a 62 3f 99 24 9e 0d 53 6a 23 3b cd c2 19
  md5 cc 08 18 d6 c6 31 1b 91 f7 51 78 04 a5 18 14 50

Enter certificate to add to trusted keystore or 'q' to quit: [1] >
```

```
Automatic peer with super-node (if you administer one, ensure that it is running) [Y/n]
```

```
Do you want to continue with the Globus installation and setup? [Y/n]
Do you want to register the GridFTP server with Globus? [Y/n]:
Please provide a Globus username []: YOUR-GLOBUS-USER
Globus password []: YOUR-GLOBUS-PASSWORD

Server sent 2 certificate(s):

1 Subject CN=vesgint-idx.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  Issuer CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  sha1 8f 1c 62 12 3a 3f 88 be 12 26 c8 f8 f9 3b da 73 a7 f6 f2 04
  md5 1f fb bd cd 25 cd f5 9d 39 42 d4 c3 ef 2d 98 20

2 Subject CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  Issuer CN=IPSL Simple CA, OU=simpleca.ipsl.upmc.fr, OU=ESGF.ORG, O=ESGF
  sha1 15 1d a1 c3 b9 0d 9a 62 3f 99 24 9e 0d 53 6a 23 3b cd c2 19
```

```
md5      cc 08 18 d6 c6 31 1b 91 f7 51 78 04 a5 18 14 50
```

```
Enter certificate to add to trusted keystore or 'q' to quit: [1] >
```

```
Finished!...
```

In order to see if this node has been installed properly you may direct your browser to:

```
http://spock.meteo.unican.es/thredds
```

```
http://spock.meteo.unican.es/esg-orp
```

```
http://spock.meteo.unican.es/
```

```
Your peer group membership -- : [esgf-test]
```

```
Your specified "default" peer : [VESGINT-IDX.IPSL.UPMC.FR]
```

```
Your specified "index" peer - : [VESGINT-IDX.IPSL.UPMC.FR] (url = http://VESGINT-IDX.IPSL.UPMC.FR/)
```

```
Your specified "idp" peer --- : [VESGINT-IDX.IPSL.UPMC.FR] (name = VESGINT-IDX.IPSL.UPMC.FR)
```

```
Your temporary certificates have been placed in /etc/tempcerts
```

```
You can install them by executing this : esg-node --install-keypair /etc/tempcerts/hostcert.pem /etc/tempcerts/hostkey.pem
```

```
When prompted for the chainfile, specify: /etc/tempcerts/cacert.pem
```

```
[Note: Use UNIX group permissions on /esg/content/thredds/esgcat to enable users to be able to publish thredds catalogs fr
```

```
%> chgrp -R <appropriate unix group for publishing users> /esg/content/thredds
```

```
-----
Administrators of this node should subscribe to the
esgf-node-admins@lists.llnl.gov by sending email to: majordomo@lists.llnl.gov
with the body: subscribe esgf-node-admins
-----
```

```
v2.5.9-master-release
```

Execute the following:

```
[root@spock bin]# ./esg-node --install-keypair /etc/tempcerts/hostcert.pem /etc/tempcerts/hostkey.pem
...
Please set the password for this keystore :
Please re-enter the password for this keystore:
...
certfile> /etc/tempcerts/cacert.pem
certfile>
...
Is the above information correct? [Y/n]
Is the above information correct? [Y/n]
```

Restart the node:

```
[root@spock bin]# ./esg-node restart
```

Check that everything works ([?https://github.com/ESGF/esgf-installer/wiki/ESGF-Post-Installation-Tests](https://github.com/ESGF/esgf-installer/wiki/ESGF-Post-Installation-Tests)).

If the CoG portal does not work follow the instructions on [?https://www.earthsystemcog.org/projects/cog/install_or_upgrade](https://www.earthsystemcog.org/projects/cog/install_or_upgrade).

Now you should be able to log in the CoG portal using the openid "[?https://spock.meteo.unican.es/esgf-idp/openid/rootAdmin](https://spock.meteo.unican.es/esgf-idp/openid/rootAdmin)" and the password that you chose in the installation process. If you have not installed the idp profile, you can administer your CoG portal from "[?https://YOUR-DOMAIN/login2](https://YOUR-DOMAIN/login2)".

Configuration for publishing

The installation process should have created a user in the postgres database, named rootAdmin. You can check it by running `psql -U dbsuper -d esgcat` (to access the postgres cli) and visualizing the table `esgf_security.user`.

```
esgcat=# select * from esgf_security.user;
id | firstname | middlename | lastname | email | username | password
```

```
| dn |
```



```

<attribute
  type="test_group"
  attributeService="https://spock.meteo.unican.es/esgf-idp/saml/soap/secure/attributeService.htm"
  description="Test group for test data"
  registrationService="https://spock.meteo.unican.es/esgf-idp/secure/registrationService.htm"/>

<attribute
  type="cordex_group"
  attributeService="https://spock.meteo.unican.es/esgf-idp/saml/soap/secure/attributeService.htm"
  description="Test group for cordex data"
  registrationService="https://spock.meteo.unican.es/esgf-idp/secure/registrationService.htm"/>

```

Generate your credentials for publication - globus certificate

```
myproxy-logon [ -b ] -s <openid_server> -l <your_esgf_username> -p 7512 -t 72 -o $HOME/.globus/certificate-file
```

The certificate is valid for 72 hours when specified by -t. If you are publishing for the first time, you will need to mkdir \$HOME/.globus and use -b to bootstrap its trustroots with the server. The esgf_username is the simply the username portion of your openid rather than the entire openid string, e.g. sashakames, not [?https://pcmdi.llnl.gov/esgf-idp/openid/sashakames](https://pcmdi.llnl.gov/esgf-idp/openid/sashakames)

Publish the test dataset

For esgprep and esgpublish to be available, execute `source /etc/esg.env`.

```

[root@spock ~]# esgprep mapfile --project test /esg/data/test/
Collecting files      : 1 files
Mapfile(s) generation: 100% |????????????????????????????????????????????????????????????| 1/1 files
Mapfile(s) generated : 1 (see /root/mapfiles)

```

```

[root@spock ~]# esgpublish --service fileservice --map mapfiles/test.test.map --project test --thredds --publish --offline
INFO      2017-06-02 14:59:48,405 Replacing files in dataset: test.test, version 1
INFO      2017-06-02 14:59:48,413 File /esg/data/test/sftlfl.nc exists, skipping
INFO      2017-06-02 14:59:48,416 New dataset version = 2
INFO      2017-06-02 14:59:48,430 Adding file info to database
INFO      2017-06-02 14:59:48,469 Writing THREDDS catalog /esg/content/thredds/esgcat/1/test.test.v2.xml
INFO      2017-06-02 14:59:48,522 Writing THREDDS ESG master catalog /esg/content/thredds/esgcat/catalog.xml
INFO      2017-06-02 14:59:48,533 Reinitializing THREDDS server
INFO      2017-06-02 14:59:48,830 Publishing: test.test
INFO      2017-06-02 14:59:49,871 Result: SUCCESSFUL

```

Notes:

1. --map must point to the file generated by esgprep mapfile
2. --thredds publish data to the data node
3. --publish publish data to the index node
4. --offline is required for publish the test dataset (Why?)
5. This publication works out of the box because esgf installs by default the required /esg/config/esgcat/esg.test.ini file.

Publish CORDEX datasets

See [CORDEXPublication](#)

Known issues during installation

#error "Pycogp requires PostgreSQL client library (libpq) >= 9.1

This error occurs sometimes during installation but removing the node and installing it from scratch seems to solve it...

```

Traceback (most recent call last):
  File "setup.py", line 110, in <module>
    """
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/distutils/core.py", line 111, in setup
    _setup_distribution = dist = klass(attrs)
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 239, in __init__
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 263, in fetch_build_egg
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 568, in resolve
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 806, in best_match
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 818, in obtain_distribution
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 313, in fetch_build_egg
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
  File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 1
distutils.errors.DistutilsError: Setup script exited with error: command 'gcc' failed with exit status 1

Sorry...
This action did not complete successfully
Please re-run this task until successful before continuing further

Also please review the installation FAQ it may assist you
https://github.com/ESGF/esgf.github.io/wiki/ESGFNode%7CFAQ

```

Failed building wheel for Pillow

This error seems unavoidable but it also seems that it doesn't affect the esgf functionality.

Error with hurry (python package)

Installation fails:

```

Installed /usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/pika-0.11.2-py2.7.egg
Searching for hurry
Reading https://pypi.python.org/simple/hurry/
Downloading https://pypi.python.org/packages/22/12/47b718233d5c3be1c0589e65b50bde21ac3408c6d40e5992d517036f9695/hurry-1.0.0-py2.7.egg
Best match: hurry 1.0
Processing hurry-1.0.tar.gz
Writing /tmp/easy_install-dWp151/hurry-1.0/setup.cfg
Running hurry-1.0/setup.py -q bdist_egg --dist-dir /tmp/easy_install-dWp151/hurry-1.0/egg-dist-tmp-BqScK5
Traceback (most recent call last):
  File "setup.py", line 130, in <module>
    """
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/_distutils/core.py", line 129, in setup
    return distutils.core.setup(**attrs)
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/distutils/core.py", line 151, in setup
    dist.run_commands()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/distutils/dist.py", line 953, in run_commands
    self.run_command(cmd)
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/distutils/dist.py", line 972, in run_command
    cmd_obj.run()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/install.py", line 67, in run
    self.do_egg_install()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/install.py", line 117, in do_egg_install
    cmd.run()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 412, in run
    self.easy_install(spec, not self.no_deps)
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 654, in easy_install
    return self.install_item(None, spec, tmpdir, deps, True)
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", line 701, in install

```



```

self.process_distribution(spec, dist, deps)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 746, in process
[requirement], self.local_index, self.easy_install
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/pkg_resources/__init__.py", line 866, in resolve
replace_conflicting=replace_conflicting
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/pkg_resources/__init__.py", line 1146, in best_match
return self.obtain(req, installer)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/pkg_resources/__init__.py", line 1158, in obtain
return installer(requirement)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 673, in easy_i
return self.install_item(spec, dist.location, tmpdir, deps)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 699, in instal
dists = self.install_eggs(spec, download, tmpdir)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 882, in instal
return self.build_and_install(setup_script, setup_base)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 1150, in build
self.run_setup(setup_script, setup_base, args)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 1136, in run_s
run_setup(setup_script, args)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 253, in run_setup
raise
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/contextlib.py", line 35, in __exit__
self.gen.throw(type, value, traceback)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 195, in setup_context
yield
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/contextlib.py", line 35, in __exit__
self.gen.throw(type, value, traceback)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 166, in save_modules
saved_exc.resume()
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 141, in resume
six.reraise(type, exc, self._tb)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 154, in save_modules
yield saved
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 195, in setup_context
yield
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 250, in run_setup
_execfile(setup_script, ns)
File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/sandbox.py", line 45, in _execfile
exec(code, globals, locals)
File "/tmp/easy_install-dWp151/hurry-1.0/setup.py", line 6, in <module>
from ez_setup import use_setuptools
ImportError: cannot import name __version__

Sorry...
This action did not complete successfully
Please re-run this task until successful before continuing further

Also please review the installation FAQ it may assist you
https://github.com/ESGF/esgf.github.io/wiki/ESGFNode%7CFAQ

```

Solution:

```

source /usr/local/conda/bin/activate esgf-pub
pip install hurry==1.0

```

And rerun the installation (esg-autoinstall)

Installing a custom certificate in the ESGF Node

You should own your certificate file (hostcert.crt) and your private key (hostkey.key). Your /etc/httpd/conf/esgf-httpd.conf must reference your certificate and key:

```
228     SSLVerifyClient optional
229     SSLVerifyDepth 10
230     SSLCertificateFile /etc/certs/hostcert.crt
231     #SSLCACertificateFile /etc/certs/esgf-ca-bundle.crt
232     SSLCertificateKeyFile /etc/certs/hostkey.key
233     #SSLCertificateChainFile /etc/certs/cachain.pem
234     SSLOptions +StdEnvVars +ExportCertData
```

Then you have to import your certificate and your key into your tomcat keystore (located in /esg/config/tomcat/ and named esg-truststore.ts and keystore-tomcat). They are configurated in /usr/local/tomcat/conf/server.xml.

1. If the self-signed certificate is installed in keystore-tomcat, remove it with `keytool -delete -alias ALIAS -keystore keystore-tomcat`, where alias can be obtained with `keytool -v -list -keystore keystore-tomcat`.
1. Execute `# openssl pkcs12 -export -in /etc/certs/hostcert.crt -inkey /etc/certs/hostkey.key -out server.p12 -name my-esgf-node -CAfile /etc/certs/hostcert.crt -caname root` and `keytool -importkeystore -deststorepass PASSWORD -destkeypass PASSWORD -destkeystore keystore-tomcat -srckeystore server.p12 -srcstoretype PKCS12 -srcstorepass PASSWORD -alias my-esgf-node`
1. Ensure it has been correctly installed with `keytool -v -list -keystore keystore-tomcat`.
1. Restart the node: `esg-node restart`
1. More info in [?Stackoverflow](#)

Notes on x509 certificates and ESGF

[?Certificate SAN](#)

[?Certificate import private key into keystore](#)

ssl.conf

```
[ca]
default_ca = CA_default

[CA_default]
dir = ./ca
database = $dir/index.txt
new_certs_dir = $dir/newcerts
serial = $dir/serial
private_key = ./cakey.pem
certificate = ./cacert.pem
default_days = 3650
default_md = sha256
policy = policy_anything
copy_extensions = copyall

[policy_anything]
countryName = optional
stateOrProvinceName = optional
localityName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional

[req]
prompt = no
distinguished_name = req_distinguished_name
req_extensions = v3_ca
```

```
[req_distinguished_name]
CN = esgf-f2f-test

[v3_ca]
subjectAltName = @alt_names

[alt_names]
IP.1 = 127.0.0.1
IP.2 = 172.28.128.3
IP.3 = 172.28.128.4
IP.4 = 172.28.128.5
DNS.1 = localhost
```

Add certificate to truststore `keytool -import -trustcacerts -alias casmg -keystore esg-truststore.ts -file /etc/certs/f2f/ssl.crt`

Redirect index to IPSL

```
Listen 443
LoadModule ssl_module modules/mod_ssl.so
NameVirtualHost *:443
<VirtualHost *:443>
    Redirect permanent / https://esgf-node.ipsl.upmc.fr/
```

Local publication in a Data Index IdP ESGF node

1. IMPORTANT: `setenforce 0`
2. Edit `/usr/local/cog/cog_install/settings.py` to ALLOW access from everywhere (`['* ']`)
3. Add your node domain to `esgf_idp_static.xml` and `esgf_known_providers.xml`
4. `cp /etc/certs/hostcert.pem /etc/grid-security/certificates/$(openssl x509 -hash -noout < /etc/certs/hostcert.pem).0`
5. Create user permissions in postgres database as explained in <https://acme-climate.atlassian.net/wiki/spaces/ESGF/pages/15073316/Guide+to+ESGF+Publishing+and+Best+Practices>
6. You should be able to log into cog
7. Modify `esgf_ats_static.xml`, eg:

```
<attribute
    type="test_group"
    attributeService="https://esgf.test.es/esgf-idp/saml/soap/secure/attributeService.htm"
    description="Test group for test data"
    registrationService="https://esgf.test.es/esgf-idp/secure/registrationService.htm"/>
```

8. Modify `esgf_policies_local.xml`, eg:

```
<!-- The following statements makes resources that contain '*.test.*' freely available for download -->
<policy resource="*.test.*" attribute_type="ANY" attribute_value="" action="Read"/>
<policy resource="*.test.*" attribute_type="test_group" attribute_value="publisher" action="Write"/>

<!-- The following statements makes all resources freely available for download -->
<policy resource="*" attribute_type="ANY" attribute_value="" action="Read"/>
```

You may need to change the file

`/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/esgset-3.5.4-py2.7.egg/esgset/publish/utility.py`

```
27 UPDATE_TIMESTAMP = "/tmp/publisher-last-check"
28 DEFAULT_CERTS_LOCATION_SUFFIX = "/etc/grid-security/certificates"
29
```

```
...
1241     if home is not None:
1242         service_certs_location = DEFAULT_CERTS_LOCATION_SUFFIX
1243
```

Modify /esg/config/esgcert/esg.ini

```
hessian_service_certfile = %(home)s/.globus/certificate-file
#hessian_service_certs_location = %(home)s/.globus/certificates
hessian_service_certs_location = /etc/grid-security/certificates
hessian_service_debug = false
```

11. Add host and CA certificates to /esg/config/tomcat/esg-truststore.ts
12. Add host certificate to /esg/config/tomcat/keystore-tomcat
13. You should be able to publish

References

- [?ESGF Installation From Scratch](#)