

ESGF Local Node Deployment Tutorial

This page shows how to deploy an ESGF Node that provides data and index services and belongs to the esgf-test federation. The purpose of this node is to test the process of publication in the ESGF before publishing into production.

This page assumes that command are executed by the root user (or sudo -s).

Index

1. Prerequisites
2. Previous installation clean up
3. Installation from scratch
4. Configuration for publication
5. Publish the test dataset
6. Publish CORDEX datasets
7. Known issues
8. References

0. Prerequisites

1. You have to create a globus account - [?https://www.globusid.org/create](https://www.globusid.org/create)
2. Open ports: [Required open ports](#)

1. Previous installation clean up

Execute `/usr/local/bin/esgf-node stop` in order to stop the current ESGF services (in case they are running).

```
[root@spock ~]# /usr/local/bin/esgf-node stop

EEEEEEEEEEEEEEEEEEEE     SSSSSSSSSSSSSSS          GGGGGGGGGGGGGGFFFFFFFFFFFFFFFFFFFFFFFF
E:::::E SS:::::S          GGG:::::GF:::::F
E:::::E ES:::::SSSSSS     GG:::::GF:::::F
EE:::::EEEEEEEE:::::E SSSSSSS G:::::GGGGGGG:::GFF:::::FFFFFFF:::F
 E:::::E     EEEEEES:::::S           G:::::G           GGGGGG  F:::::F           FFFFFF
 E:::::E           S:::::S           G:::::G           F:::::F
 E:::::EEEEEEEE     S:::::SSSS     G:::::G           F:::::FFFFFFF
 E:::::E           SS:::::SSSS     G:::::G           GGGGGGGGGG  F:::::F
 E:::::E           SSS:::::SS     G:::::G           G:::::G  F:::::F
 E:::::EEEEEEEE           SSSSSS:::::S G:::::G           GGGGG:::G  F:::::FFFFFFF
 E:::::E           S:::::SG:::::G           G:::::G  F:::::F
 E:::::E           EEEEE           S:::::S G:::::G           G:::::G  F:::::F
EE:::::EEEEEEEE:::::ESSSSSSS  S:::::S  G:::::GGGGGGG:::GFF:::::FF
E:::::E ES:::::SSSSSS     GG:::::GF:::::FF
E:::::E ES:::::SSSSSS     GGG:::::GGG:::GF:::::FF
EEEEEEEEEEEEEEEEEEEE     SSSSSSSSSSSSS          GGGGGG  GGGGFFFFFFFFFFFF.llnl.gov

Checking that you have root privs on spock.meteo.unican.es... [OK]
Checking requisites...

Using IP: 193.144.184.40
Stopping search services...
Using solr_workdir=/usr/local/src/esgf/workbench/esg/solr-5.5.3
Using solr_install_dir=/usr/local/solr-home/slave-8983
Using solr_data_dir=/esg/solr-index/slave-8983
Using solr_server_dir=/usr/local/solr
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)
-----
-----

```

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/esgctet]: postgresql://
What is the (low priv) db account for publisher? [esgctet]:
What is the db password for publisher user (esgctet)? []:
```

```
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/esgctet/esg.ini]
```

```
Is this correct? [Y/n]
```

```
Your publisher configuration file will be: /esg/config/esgctet/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/esgset 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.earthsystemco2.org/projects/cog/install_or_upgrade](https://www.earthsystemco2.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 esgset` (to access the postgres cli) and visualizing the table `esgf_security.user`.

```

esgset=# select * from esgf_security.user;
id | firstname | middlename | lastname | email | username | password | dn |
ion_token | notification_code
-----+-----+-----+-----+-----+-----+-----+-----+-----
1 | Admin | | User | emailOfTheAdmin | rootAdmin | hashOfThePassword | | https://domain/esgf-idp

```

```
-b50e-d43692adc5e5 |
```

In order to test the publication, create a new user using the CoG web interface ([?https://index_node_fgdn](https://index_node_fgdn)). You should click on 'Create Account' and fill the form. Once the user is created using the CoG interface, it should be visible in the `esgf_security.user` table of the postgres database.

```
esgct=# select * from esgf_security.user;
id | firstname | middlename | lastname | email | username | password | dn |
ion_token | notification_code
-----+-----+-----+-----+-----+-----+-----+-----+-----+
1 | Admin | | User | emailOfTheAdmin | rootAdmin | hashOfThePassword | |
-b50e-d43692adc5e5 | | | 0
2 | zequi | | cimadevilla | emailOfZequi | zequi | hashOfThePassword | |
```

Once the user is created, create permissions and roles as follows:

(reference documentation - [?https://acme-climate.atlassian.net/wiki/display/ESGF/Guide+to+ESGF+Publishing+and+Best+Practices](https://acme-climate.atlassian.net/wiki/display/ESGF/Guide+to+ESGF+Publishing+and+Best+Practices))

```
esgct=# select * from esgf_security.role;
id | name | description
---+-----+-----
1 | super | Super User
2 | none | None
3 | default | Standard
4 | publisher | Data Publisher
5 | admin | Group Administrator
6 | user | user role
(6 rows)

esgct=# select * from esgf_security.group;
id | name | description | visible | automatic_approval
---+-----+-----+-----+-----
1 | wheel | Administrator Group | t | t
2 | test_group | test group | t | t
3 | cordex_group | cordex group | t | t
(3 rows)

esgct=# select * from esgf_security.permission;
user_id | group_id | role_id | approved
-----+-----+-----+-----
2 | 2 | 4 | t
2 | 2 | 6 | t
2 | 3 | 6 | t
2 | 3 | 4 | t
(4 rows)
```

Add the following elements to `/esg/config/esgf_policies_local.xml`

```
<policy resource=".*test.*" attribute_type="test_group" attribute_value="user" action="Read"/>
<policy resource=".*test.*" attribute_type="test_group" attribute_value="publisher" action="Write"/>
<policy resource=".*cordex.*" attribute_type="cordex_group" attribute_value="user" action="Read"/>
<policy resource=".*cordex.*" attribute_type="cordex_group" attribute_value="publisher" action="Write"/>
```

Add the following elements to `/esg/config/esgf_ats_static.xml`

```
<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"/>
```



```

File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 263, in fet
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/pkg_resources.py", line 568, in resolv
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_m
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
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/dist.py", line 313, in fet
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
File "/usr/local/uvcdat/2.2.0/lib/python2.7/site-packages/setuptools-1.4-py2.7.egg/setuptools/command/easy_install.py", l
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/22/12/47b718233d5c3belc0589e65b50bde21ac3408c6d40e5992d517036f9695/hurry-1.0.
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/__init__.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/command/install.py", line 67, in run
    self.do_egg_install()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/install.py", line 117, in do_egg_inst
    cmd.run()
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/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/command/easy_install.py", line 654, in easy_i
    return self.install_item(None, spec, tmpdir, deps, True)
  File "/usr/local/conda/envs/esgf-pub/lib/python2.7/site-packages/setuptools/command/easy_install.py", line 701, in instal
    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 proces
    [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_install
    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 install_item
    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 install_eggs
    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_and_install
    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_setup
    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 (esgf-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 /esgf/config/tomcat/ and named esgf-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: `esgf-node restart`
1. More info in [?Stackoverflow](#)

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/

```

References

- [?ESGF Installation From Scratch](#)