Sci-GaIA OAR Documentation

Release latest

March 30, 2016

Contents

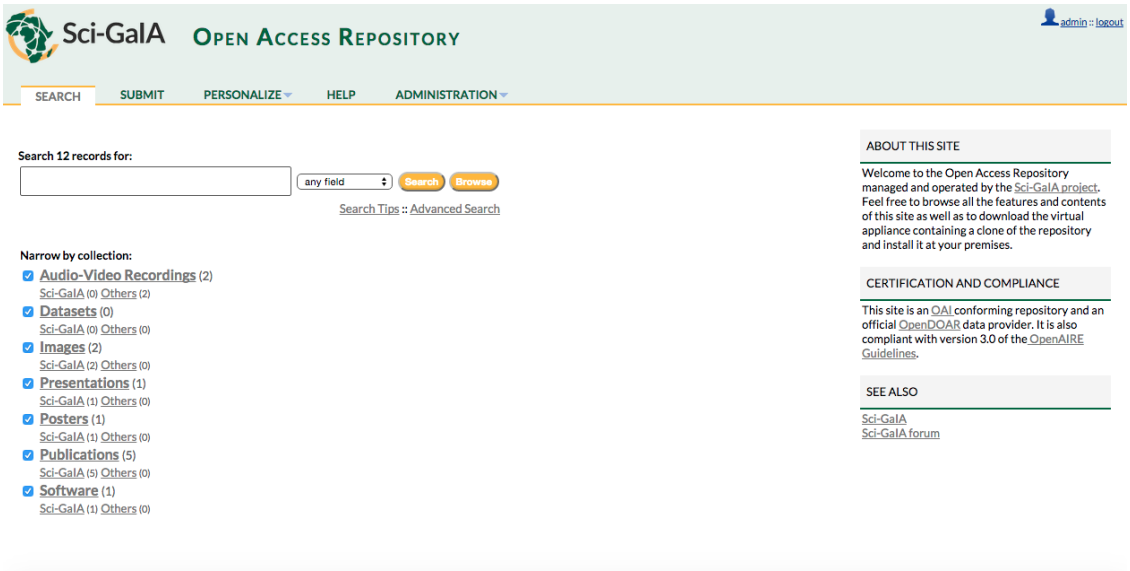
Welcome to OAR's documentation. In this document, we will cover the basic steps for installation, customisation and configuration of the virtual appliance providing the Invenio-based open-access repository at your site.

Version Available 1.0

OAR runs on

- Operating System : Ubuntu 14.04.3 LTS
- Python : 2.7.6
- MySQL Version 14.14 Distribution 5.5.44
- Invenio 1.2.1

The virtual appliance contains a clone of Sci-GaIA Open Access Repositories Sci-GaIA OAR. If you'd like to install your own open access repository, fully standards and metadata compliant, you can simply download this appliance and deploy it on your virtualization environment or private cloud.



CHAPTER 1

Virtual Machine

1.1 About

Version Available 1.2

OAR runs on

- SO Ubuntu 14.04.3 LTS
- Python 2.7.6
- Mysql Ver 14.14 Distrib 5.5.46
- Invenio 1.2.1

The virtual appliance contains a clone of Sci-GaIA Open Access Repositories Sci-GaIA OAR, if you'd like to install your own open access repository based on standard technologies, you can simply download this clone and deploy it on your virtualization environment.

1.2 Deploying OAR

To deploy your own open access repository, you can download the image from here, the file size is about 10GB. In this way you download the Sci-GaIA Open Access Repository template that can be deployed on your virtualization environment. The image is in QCOW format, but can be easily converted in other format as you need, using **qemu** utils.

This guide shows you two examples of how to use virtual appliance template in a Openstack based cloud infrastructure and in a local Virtualbox environment.

1.2.1 First Access

Before you can do the first access to your newly OAR installation, please contact us to get the default OAR template credentials. This template allows login only with keys and don't permit SSH root login, for security reasons. Once you get default credentials, login into the OAR installation from the virtualization environment console and perform the the following steps.

Warning: If you don't do this you will get hacked.

1. Add your ssh public keys to the invenio user

Note: You can use your preferred way to do this stuff. For example, if you maintain your public keys with the github service you can do the following:

- wget https://github.com/<github_username>.keys
- mv <github_username>.keys .ssh/authorized_keys

2. Test remote login:

```
ssh invenio@<oar_ip_address>
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

 System information disabled due to load higher than 1.0

 Get cloud support with Ubuntu Advantage Cloud Guest:
    http://www.ubuntu.com/business/services/cloud
```

3. Setup firewall according your security requirements, the default rules applied to the the template are the following:

```
sudo iptables -L -n
Chain INPUT (policy DROP)
target     prot opt source               destination
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0            state RELATED,ESTABLISHED
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:0x3F/0x00
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:!0x17/0x02 s...
DROP       tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:0x3F/0x3F
ACCEPT     all  --  0.0.0.0/0            0.0.0.0/0
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:22
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:80
ACCEPT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp dpt:443
REJECT     tcp  --  0.0.0.0/0            0.0.0.0/0            tcp flags:0x16/0x02 re...
REJECT     all  --  0.0.0.0/0            0.0.0.0/0            reject-with icmp-host-...

Chain FORWARD (policy ACCEPT)
target     prot opt source               destination

Chain OUTPUT (policy ACCEPT)
target     prot opt source               destination
```

1.3 Deployment Examples

1.3.1 Openstack deployment

This section shows how to the deploy the OAR image template on an Openstack cloud based infrastructure.

Note: The steps below describe the process using the **Openstack Dashboard**, if you cannot access Openstack Dashboard, you can issue the equivalent Command Line Interface commands.

1. Create a new image in the image service, clicking the *Images* link in the left side menu and then click *Create Image* button

2. Fill all fields with your desidered values (see Figure *1* as example) and then click *Save* button.

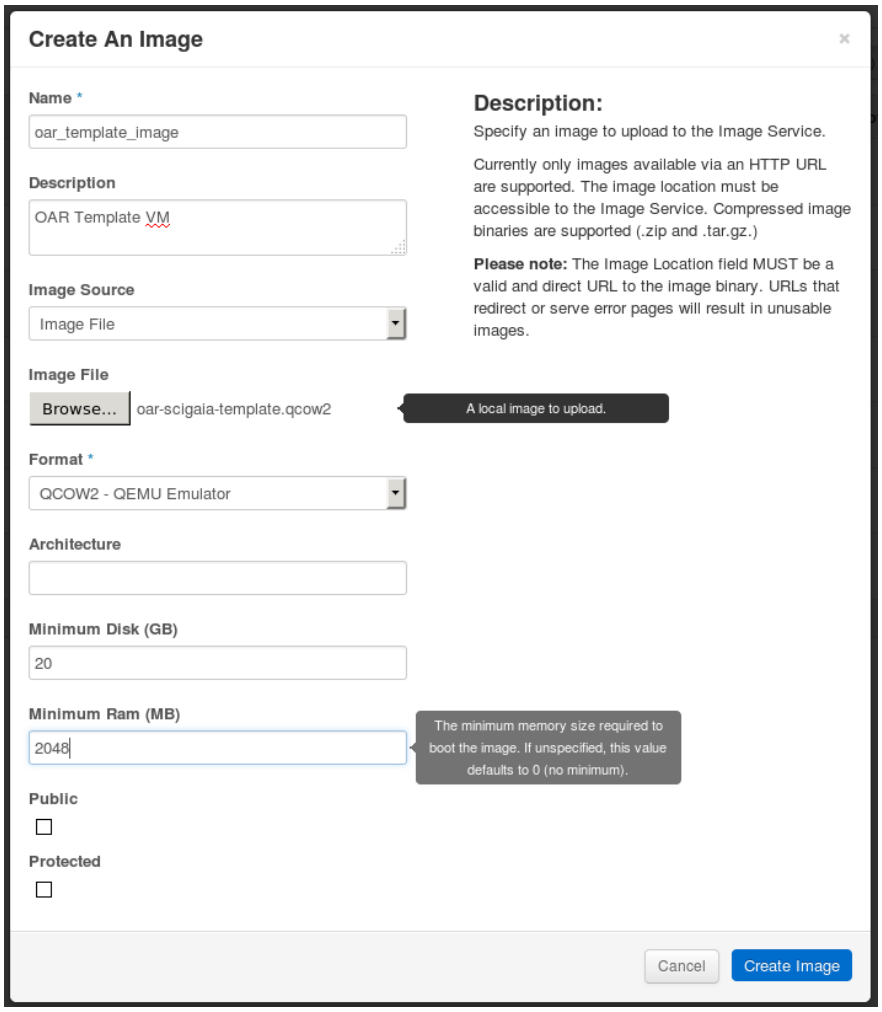


Fig. 1.1: Create new image.

Note: Pay attention to *Minimun disk* value: the OAR template require at least 20GB.

3. **Once the image becomes ready, create a new instance:**
   (a) Click *Instances* link in the left side menu.
   (b) Click *Launch Instance* button.
4. Fill all fields with your desidered values for all tabs (see Figure *2* as example) and then click *Save* button.
5. Wait until the new instaces *Power State* becomes *Running*.
6. Open the instance console, and follow the *First Access* steps.

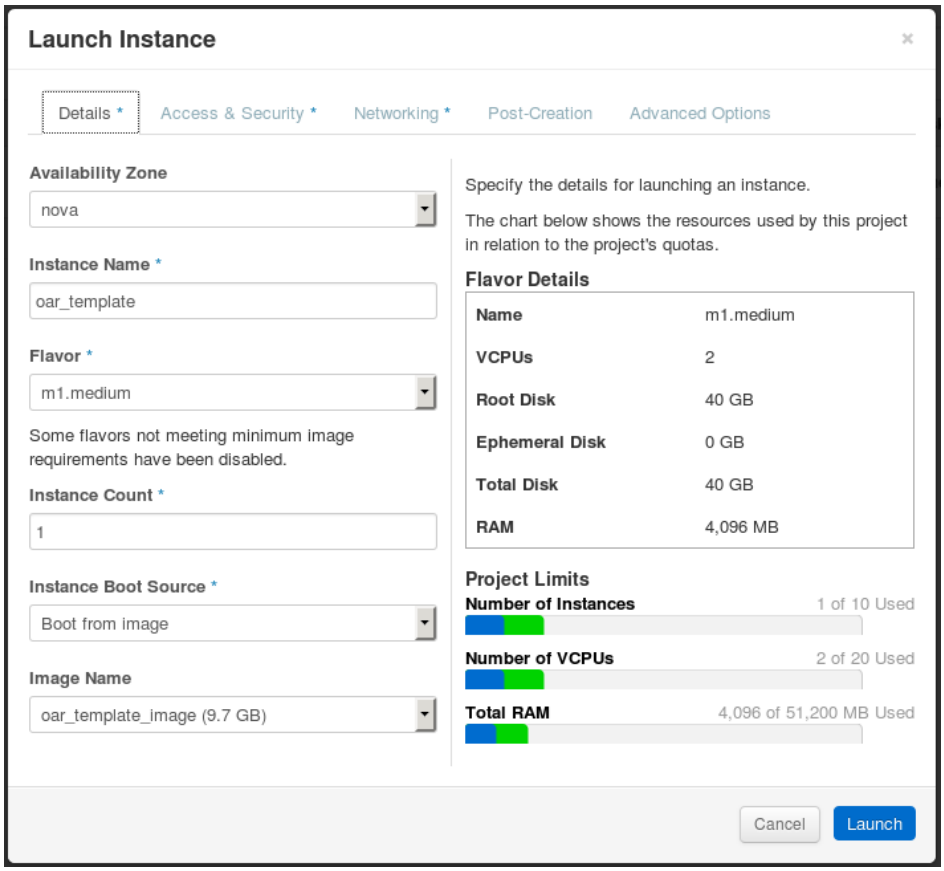


Fig. 1.2: Create new instance.

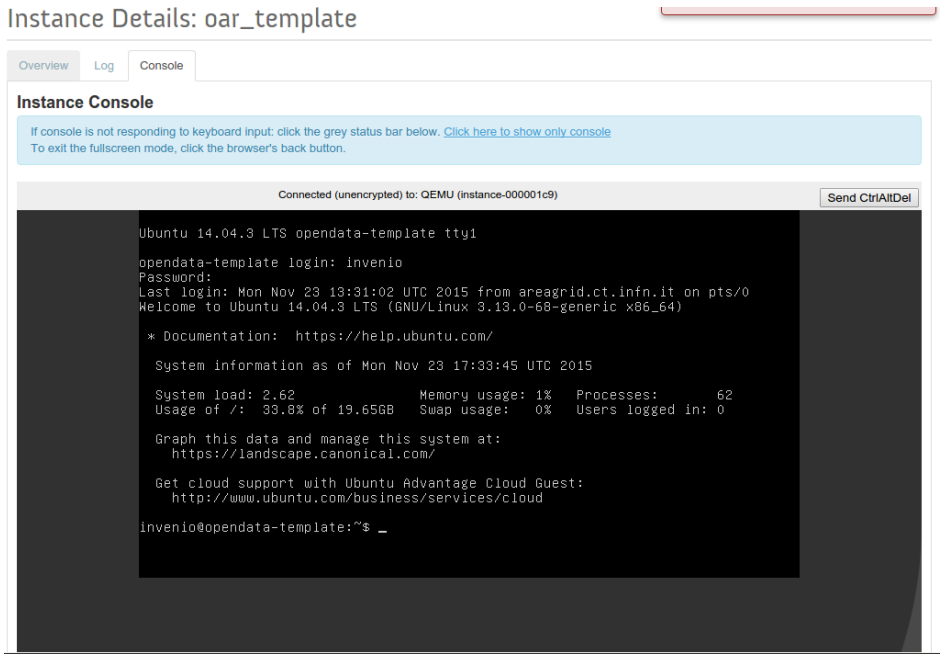


Fig. 1.3: OAR instance console.

1.3.2 VirtualBox deployment

> **Warning:** This deployment example is provided just for test or demostrative purposes, don't use for production environment.

Note: Sometimes you could experiment problems deploying OAR on Virtualbox using the provided QCOW image. In this case you can convert the disk format from qcow2 to vdi using qemu utils, as described in the *Troubleshooting* section.

In order to deploy the image on Virtualbox you should:

1. create a new vitual machine (see Figure *3*) specifing your machine name, OS type anchitecture, then click *Next* button;
2. specify the machine RAM size, use at least 2GB of RAM (see Figure *4*), click *Next* button;
3. attach the downloaded image as disk (see Figure *5*);
4. finally start the virtual machine. It may take some time before start, depends on your hardware.

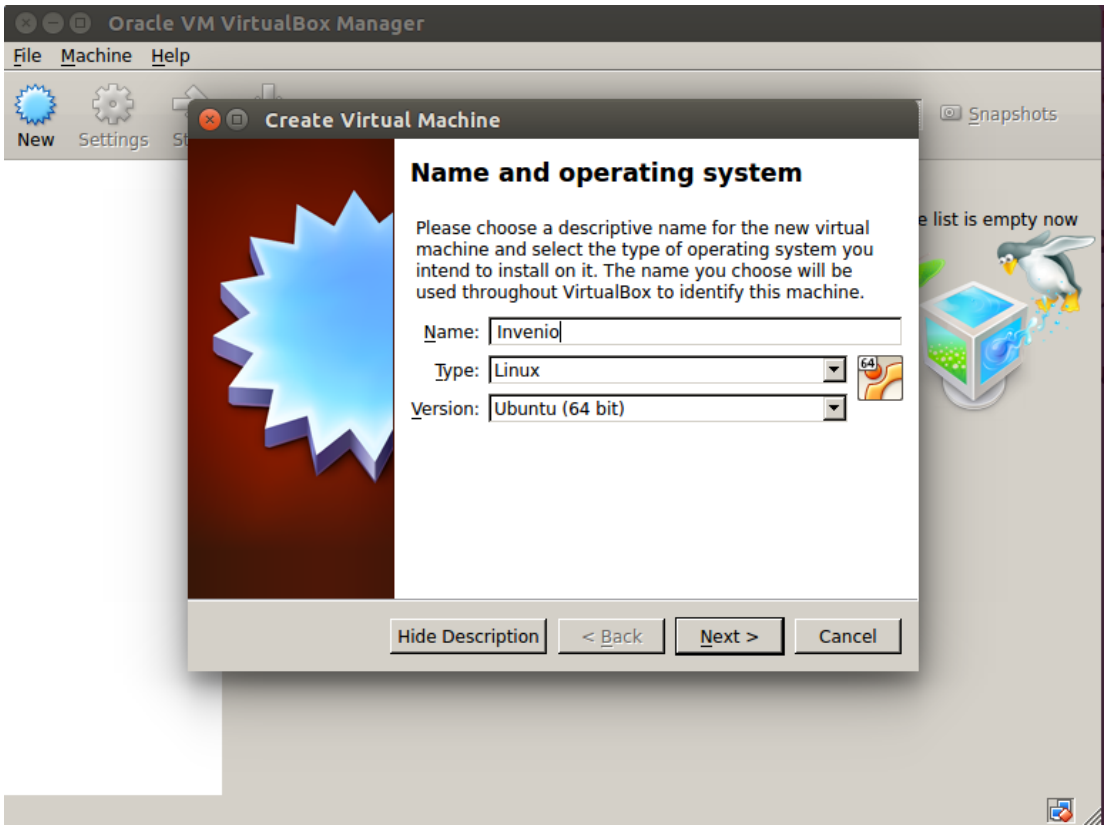


Fig. 1.4: Create new Virtual Machine.

Once the virtual machine is up and running provide the default credentials to login into (see Figure *6*).

The image is equiped with 20GB dinamically allocated disk, if you need more disk space you can perform the following commmands:

1. shtdown the Virtual machine;

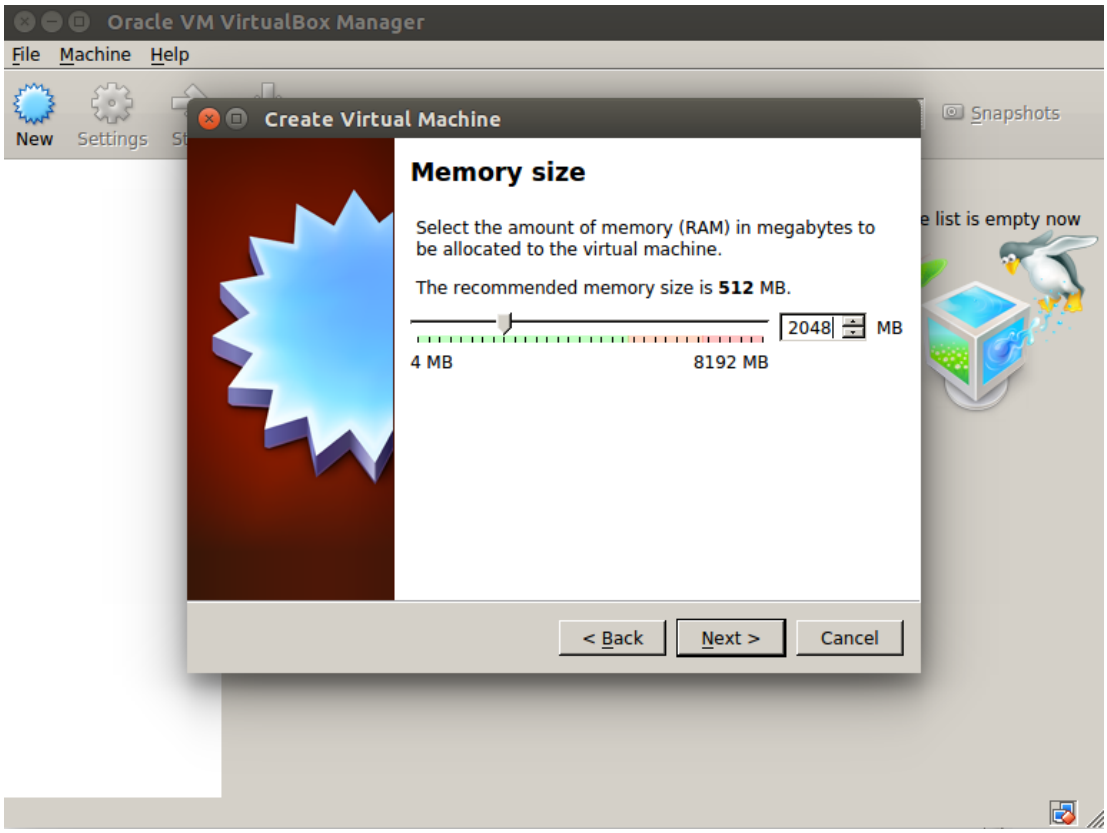


Fig. 1.5: Specifiy the RAM size.

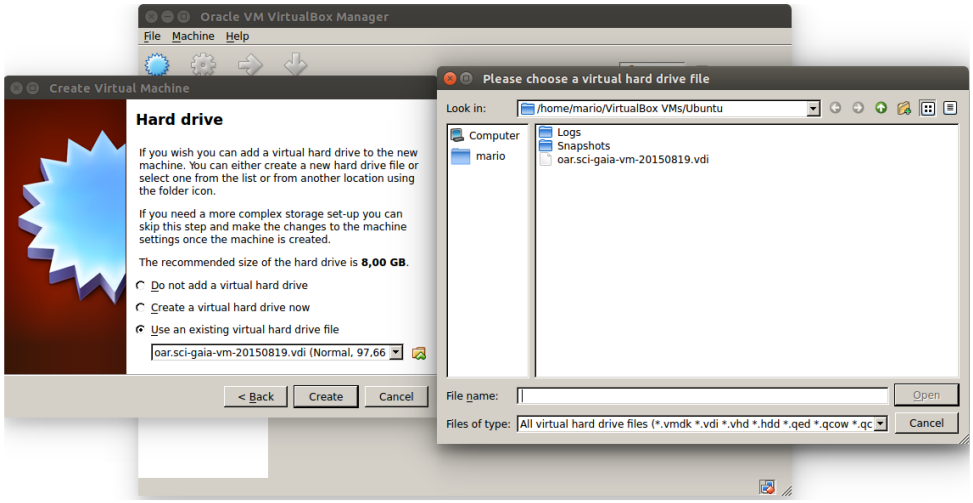


Fig. 1.6: Attach oar image.

Fig. 1.7: OAR template console.

2. from your guest system perform the **VBoxManage modifyhd** specifying the new Hard disk size in MB:

```
VBoxManage modifyhd /path/to/the/oar.sci-gaia-vm-20150819.vdi --resize <new_size(MB)>
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
```

3. restart the Virtual Machine, login into and check the disk size using:

```
invenio@opendata-template:~$ df -Th
```

1.4 Troubleshooting

In this section there are some possible solutions to the problems you could face during the OAR template deployment.

1.4.1 Cannot access Virtual Machine

Problem

Although you provide the right credentials you cannot access the Virtual Machine from console, see Figure 7

Solution

This problem is often related to the keyboard layout loaded, please check the special character typing them temporarly on the username to be sure that you are typing the right password.

1.4.2 Disk extension

Problem

If you successfully excuted a disk extension, but when you check the size you still see the default size.

Fig. 1.8: Error Accessing the Virtual Machine.

```
root@opendata-template:~# df -Th
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/sda1      ext4       20G  7.3G   12G  39% /
none           tmpfs     4.0K     0  4.0K   0% /sys/fs/cgroup
udev           devtmpfs  997M   12K  997M   1% /dev
tmpfs          tmpfs     201M  376K  200M   1% /run
none           tmpfs     5.0M     0  5.0M   0% /run/lock
none           tmpfs    1001M     0 1001M   0% /run/shm
none           tmpfs     100M     0  100M   0% /run/user

root@opendata-template:~# fdisk -l

Disk /dev/sda: 104.9 GB, 104857600000 bytes
4 heads, 32 sectors/track, 1600000 cylinders, total 204800000 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00045d27

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048   204799999   102398976   83  Linux
```

Solution

Problably you need to perform the **resize2fs** to enlarge the file system, as shown below that expands the disk size from 20GB to 100GB:

```
root@opendata-template:~# resize2fs /dev/sda1
resize2fs 1.42.9 (4-Feb-2014)
Filesystem at /dev/sda1 is mounted on /; on-line resizing required
old_desc_blocks = 2, new_desc_blocks = 7
The filesystem on /dev/sda1 is now 25599744 blocks long.

root@opendata-template:~# df -Th
```

```
Filesystem     Type      Size  Used Avail Use% Mounted on
/dev/sda1      ext4       97G  7.3G   85G   8% /
none           tmpfs     4.0K     0  4.0K   0% /sys/fs/cgroup
udev           devtmpfs  997M   12K  997M   1% /dev
tmpfs          tmpfs     201M  376K  200M   1% /run
none           tmpfs     5.0M     0  5.0M   0% /run/lock
none           tmpfs    1001M     0 1001M   0% /run/shm
none           tmpfs     100M     0  100M   0% /run/user
```

1.4.3 Virtualbox instance doen't start

Problem

As pointed in the *VirtualBox deployment* section you couldn't be able to start the Virtual Machine due to Hard Disk related problems.

Solution

In this case you tray to convert the downloaded image format from QCOW2 to VDI. Following the steps to convert image format.

1. Install **qemu-utils**

```
apt-get install qemu-utils
```

2. Convert the image format:

```
qemu-img convert -f qcow2 <qcow2_VM_filename> -O vdi <VDI_file_VM_filename>
```

3. Use the just created vdi image to start the Virtual Machine.

CHAPTER 2

OAR Configuration

See also:

- About Invenio
- Invenio Documentation
- Admin HOWTO guides

Getting Started

1. Edit your **invenio-local.conf**

```
$ sudo -u www-data vim /opt/invenio/etc/invenio-local.conf # edit as follows
```

and put wanted values there:

Site URL

```
CFG_SITE_URL = http://yoursite.org
CFG_SITE_SECURE_URL = https://yoursite.org
```

Site Name

```
## CFG_SITE_NAME -- the visible name of your Invenio installation.
CFG_SITE_NAME = Institute

## CFG_SITE_NAME_INTL -- the international versions of CFG_SITE_NAME
## in various languages.  (See also CFG_SITE_LANGS below.)
CFG_SITE_NAME_INTL_en =  Institute
CFG_SITE_NAME_INTL_fr = Institut
```

SuperUser and Email Address

```
# CFG_SITE_SUPPORT_EMAIL -- the email address of the support team for
# this installation:

CFG_SITE_SUPPORT_EMAIL = admin@sci-gaia.eu


# CFG_SITE_ADMIN_EMAIL -- the email address of the 'superuser' for
# this installation.  Enter your email address below and login with
# this address when using Invenio inistration modules.  You
# will then be automatically recognized as superuser of the system.

CFG_SITE_ADMIN_EMAIL =  admin@sci-gaia.eu
```

Mail Server

```
# CFG_MISCUTIL_SMTP_HOST -- which server to use as outgoing mail server to
# send outgoing emails generated by the system, for example concerning
# submissions or email notification alerts.

CFG_MISCUTIL_SMTP_HOST = yourserver
```

2. Propagate these changes to all installed files:

```
$ sudo -u www-data /opt/invenio/bin/inveniocfg --update-all
```

3. Update Apache configuration file, either by running:

```
$ sudo -u www-data /opt/invenio/bin/inveniocfg --create-apache-conf
```

or by manually editing virtual host configuration files

```
sudo -u www-data vim /opt/invenio/etc/apache/invenio-apache-vhost*.conf.
```

4. You can restart your Apache server now:

```
$ sudo /etc/init.d/apache2 restart
```

5. Remove help pages (user|admin|hacking) cache (please first ensure that you have not mistakenly edited these files to add custom information, instead of editing the source of the help pages):

```
$ sudo -u www-data rm -r /opt/invenio/var/cache/webdoc/
```

(Cache will be automatically recreated based on the source file when one accesses a page. You can force the creation of these pages by accessing the table of content for each section: http://yoursite.eu/help/contents, http://yoursiste.eu/help/admin/contents and http://yoursite.eu/help/hacking/contents)

6. In order to customize categories, you must run

```
cd /opt/invenio/bin
sudo -u www-data ./bibindex
sudo -u www-data ./webcoll
sudo -u www-data ./bibsched
```

and run (r) all processes in the bibsched window

7. Put your bibsched queue back to automatic mode, and you are done. (See more: Howto Run Invenio installation)

```
cd /opt/invenio/bin/
sudo -u www-data ./bibsched
```

CHAPTER 3

OAR - DOI/PID

If you would like to change the DOI/PID Prefix

```
cd /opt/invenio/var/www/form
sudo vim request_doi.py
```

```
#!/usr/bin/env python

import json,cgi,time
import httplib2, sys, base64, codecs

res=[]
retCode=0
errCode=''
doi='11623'
res =  "%s/sci-gaia:%s" % (doi,time.time())
print "Content-type: application/json\n\n"
print json.dumps(res)
```

Change the prefix line “res” from %s/sci-gaia:%s to %s/<repo-name>:%s where <repo-name> is the name you want to give to your repository.

For each new record, send the following email:

```
*Send to*: <handles@sci-gaia.eu>

*Subject*: OAR <repo-name> - new PID


Dear Handle Server Administrators,

Could you please register the PID of the following resource?

CREATE 11623/<repo-name>:<unique-id>
100 HS_ADMIN 86400 1110 ADMIN 300:111111111111:0.NA/11623
2 URL 86400 1110 UTF8 https://<repo-name>/record/<id>
3 DESC 86400 1110 UTF8 <Title of the record>

Best regards,

The Librarian of the <repo-name> Open Access Repository
```

CHAPTER 4

External Authentication: Shibboleth

Version Available 1.0

External Authentication: Shibboleth

- Shibboleth 2.5.2
- Apache 2.4.7
- Invenio 1.2.1

If your institution has setup Single Sign-On solution based on SAML, here are the steps to follow in order to integrate Shibboleth with Invenio 1.2.1 as a Service Provider.

Installing necessary OS packages

```
# apt-get install libapache2-mod-shib2
```

Configuring Shibboleth

Modify the file `/etc/shibboleth/shibboleth2.xml` as follows:

```
# diff /etc/shibboleth/shibboleth2.xml
23c23,24c24,
<                           entityID="https://oar.sci-gaia.eu/shibboleth"  attributePrefix="ADFS_"
<                           REMOTE_USER="mail eppn persistent-id targeted-id" signing="true">
---
>                           entityID="https://example.com/shibboleth"
>                           REMOTE_USER="eppn persistent-id targeted-id">
36c36
<                    checkAddress="false" handlerSSL="true" cookieProps="http">
---
>                    checkAddress="false" handlerSSL="false" cookieProps="http">
44,45c44,45
<              <SSO
<                   discoveryProtocol="SAMLDS" discoveryURL="https://gridp.garr.it/ds/WAYF">
---
>              <SSO entityID="https://idp.example.org/idp/shibboleth"
>                   discoveryProtocol="SAMLDS" discoveryURL="https://ds.example.org/DS/WAYF">
69c69
<          <Errors supportContact="admin@sci-gaia.eu"
---
>          <Errors supportContact="root@localhost"
81,83d80
<          <MetadataProvider type="XML" uri="https://gridp.garr.it/metadata/gridp-test.xml"
```

```
<                backingFilePath="gridp-test.xml" reloadInterval="7200">
<          </MetadataProvider>
```

Modify the file `/etc/shibboleth/attribute-map.xml` uncommenting LDAP-based attributes

Copy your certificate and key into `/etc/shibboleth` with name `sp-cert.pem` and `sp-key.pem` respectively and restart the service.

```
# service shibd restart
```

Plugging SSO into Invenio

In order to activate the particular Shibboleth SSO authentication support you should do:

1. customizing the `external_authentication_sso.py` file in order to support your particular system
2. properly setting up `access_control_config.py` file
3. properly configuring your Apache module and update your Apache configuration

For the Sci-GaIA Project the previous steps have been implemented as follows:

1. Download the file `external_authentication_sso_scigaia.py` in `/opt/invenio/lib/python/invenio`

`external_authentication_sso_scigaia.py`.

2. Modify the file `access_control_config.py`

```
#sudo vim /opt/invenio/lib/python/invenio/access_control_config.py


> else:
                CFG_EXTERNAL_AUTH_DEFAULT = 'Local'
                CFG_EXTERNAL_AUTH_USING_SSO = False
                CFG_EXTERNAL_AUTH_LOGOUT_SSO = None
                CFG_EXTERNAL_AUTHENTICATION = {
                "Local": None,
                "Robot": ExternalAuthRobot(enforce_external_nicknames=True, use_zlib=False),
                "ZRobot": ExternalAuthRobot(enforce_external_nicknames=True, use_zlib=True)
        }

---

< else:
        import external_authentication_sso_scigaia as ea_sso
        CFG_EXTERNAL_AUTH_USING_SSO = "SCI-GAIA"
        CFG_EXTERNAL_AUTH_DEFAULT = CFG_EXTERNAL_AUTH_USING_SSO
        CFG_EXTERNAL_AUTH_LOGOUT_SSO = 'https://oar.sci-gaia.eu/Shibboleth.sso/Logout'
        CFG_EXTERNAL_AUTHENTICATION = {
        CFG_EXTERNAL_AUTH_USING_SSO : ea_sso.ExternalAuthSSOSCIGAIA(True),
                "Local": None
        #     "Robot": ExternalAuthRobot(enforce_external_nicknames=True, use_zlib=False),
        #     "ZRobot": ExternalAuthRobot(enforce_external_nicknames=True, use_zlib=True)
        }
```

Add a new method into `/opt/invenio/lib/python/invenio/webuser.py`

```
def get_mail_from_mail_group(mailgroup):
"""Return the first registered mail from colon or semicolon
   group of email. Return the mailgroup when the email does not exists."""
try:
```

```
        for mail in re.split(";|,",mailgroup):
                res = run_sql("SELECT email FROM user WHERE email LIKE %s", ("%"+mail+"%",))
                if res:
                        return res[0][0]
except OperationalError:
        register_exception()

return mailgroup
```

```
# service apache2 restart
```

3. Apache configuration

```
# a2enmod ssl
```

Edit the file `/opt/invenio/etc/apache/invenio-apache-vhost-ssl.conf`.

Set the variables

`SSLCertificateFile` and `SSLCertificateKeyFile` to your certificate and key and comment/uncomment depending on your apache version. Finally append the following to your virtual host:

```
    <Location "/Shibboleth.sso/">
    #   SSLRequireSSL   # The modules only work using HTTPS
    #   AuthType shibboleth
    #   ShibRequireSession On
    #   ShibRequireAll On
    #   ShibExportAssertion Off
    #   require valid-user
    #   Allow from all
       SetHandler shib
    </Location>
    <Location ~ "/youraccount/login|Shibboleth.sso/">
       SSLRequireSSL
       AuthType shibboleth
       ShibRequestSetting requireSession 1
       require valid-user
    </Location>
    Alias "/shibboleth" "/var/www/shibboleth"
    <Directory "/var/www/shibboleth">
       Options MultiViews
       AllowOverride None
       Order allow,deny
       Allow from all
    </Directory>
```

Enable the site:

```
# a2ensite invenio-ssl
# service apache2 restart
```

Publish the metadata of your SP in a Federation.

For GrIDP contacts are avaible in this page

CHAPTER 5

Post-configuration

This chapter will walk you through a few basic functional checks of your newly configured repository. Be sure to follow this documentation only *after* finishing the full customisation section.

Your installation contains its own copy of the Invenio documentation, which is kept under .. Refer to this documentation during the course of this chapter.

5.1 Submission of a new document or object.

The first task is to see whether the submission of a sample document is working. In order to check this, do the following :

5.2 Dealing with submissions

Once a user submits a new object, the site librarian has to process it in a specific workflow

CHAPTER 6

Support

6.1 Questions and comments

If there are questions or comments regarding this documentation or the service itself, please open a topic at the African e-Infrastructures Forum under the "Open Access" category.

6.2 Issues or errors

If you find issues or errors in the this documentation, please open an issue. For direct help or support, as a last resort, you can contact :

- Roberto BARBERA
- Rita RICCERI
- Mario TORRISI