

GIS - Support #6990

Create a VM for Notebooks

30/10/2018 12:48 - Philippe May

Status:	Resolved	Start date:	30/10/2018
Priority:	Normal	Due date:	
Assignee:	Philippe May	% Done:	100%
Category:		Estimated time:	0.00 hour
Target version:		Spent time:	0.00 hour
Description			
We have identified that the gis.csr.av domU has issues with the file system. It switches to read only at times (heavy load).			
In order to mitigate the risks, the notebook server should be hosted in its own virtual machine (domU).			
Beside, the plan is to install Gisaf/asyncio on it, in order to get notebooks compliant with the new Gisaf API.			

History

#1 - 30/10/2018 12:53 - Philippe May

Follow the instruction on https://redmine.auroville.org.in/projects/gis/wiki/CSR_server , with:

```
export vm_name=jupyter.csr.av
export vm_ip_addr=172.16.0.8
export vm_password=foobar
xen-create-image --broadcast=172.16.0.255 --dist=stretch --fs=ext4 --gateway=172.16.0.1 --hostname ${vm_name}
--ip=${vm_ip_addr} --lvm=dream.csr --maxmem=512M --memory=256M --mirror=http://ftp.de.debian.org/debian/ --net
mask=255.255.255.0 --password=${vm_password} --size=10G --swap=1G --vcpus=1 --bridge=br0
```

IP address (in /etc/systemd/network/50-wired.network): 192.168.0.19

Next: Ansible.

#2 - 30/10/2018 15:20 - Philippe May

Ansiblizing (roles: avcan, nginx)

Turn it to Buster:

```
apt update
apt upgrade
apt dist-upgrade
# Update /etc/apt/sources.list
apt update
apt upgrade
apt dist-upgrade
```

#3 - 30/10/2018 16:06 - Philippe May

Install Python 3.7:

```
apt install python3.7 python3.7-venv python3-venv
```

Jupyter (try the easy way first: Debian packages, which are up to date in Buster)

```
apt install jupyter jupyter-nbconvert jupyter-notebook
```

```
apt install python3-gdal python3-matplotlib python3-geopandas
```

We'll want to be able to install jupyterlab extensions, which require node+npm:

```
apt install node npm
```

#4 - 30/10/2018 17:10 - Philippe May

Jupyterlab and other python packages from pypi:

```
pip3 install jupyterlab nest_asyncio
```

#5 - 30/10/2018 17:17 - Philippe May

Gisaf

```
cd /usr/local/lib/  
git clone git-av:gisaf gisaf_src  
cd gisaf_src  
# Jupyter notebooks are all set to use the (future) asyncio version, in the graphql branch as of now:  
git checkout graphql  
/usr/local/lib/gisaf_src# pip3 -e .
```

Nginx (/etc/nginx/sites-enabled/jupyter)

```
map $http_upgrade $connection_upgrade {
    default upgrade;
    ''      close;
}

server {
    listen 80;
    #listen 443 ssl;
    #ssl_certificate /etc/nginx/keys/nginx.crt;
    #ssl_certificate_key /etc/nginx/keys/nginx.key;

    server_name jupyter.csr.av;

    gzip on;
    gzip_comp_level 5;
    gzip_types text/plain application/javascript application/x-javascript text/javascript text/xml text/css;

    access_log /var/log/nginx/jupyter.access.log;
    error_log /var/log/nginx/jupyter.error.log;

    root /usr/local/lib/gisaf_src/gisaf-app/dist;

    client_max_body_size 200M;

    location /dashboard-attachment/ {
        alias /var/lib/share/gisaf/dashboard/;
    }

    location /notebooks/ {
        proxy_pass http://127.0.0.1:8889;

        proxy_set_header Host $http_host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-NginX-Proxy true;

        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection "upgrade";
        #proxy_read_timeout 86400;
        proxy_set_header Referer http://localhost;
        proxy_set_header Origin "";
    }
    location / {
        return 301 /notebooks/;
    }
}
```

```
systemctl restart nginx.service
```

Jupyter service (/etc/systemd/system/jupyter-gisaf.service)

```
[Unit]
Description=Jupyter Notebook for Gisaf

[Service]
Type=simple
PIDFile=/run/jupyter-gisaf.pid
ExecStart=/usr/local/bin/jupyter-lab --config=/etc/jupyter/config_gisaf.py
User=www-data
Group=www-data
WorkingDirectory=/usr/local/lib/gisaf_src/notebooks
Restart=always
RestartSec=10
#KillMode=mixed

[Install]
WantedBy=multi-user.target
```

Config (/etc/jupyter/config_gisaf.py)

```
# get the config object
c = get_config()
# in-line figure when using Matplotlib
c.IPKernelApp.pylab = 'inline'
# listen on all interfaces
c.NotebookApp.ip = '*'
# port configuration
c.NotebookApp.port = 8889
# do not open a browser window by default when using notebooks
c.NotebookApp.open_browser = False
# No token. Always use jupyter over ssh tunnel
c.NotebookApp.token = ''
# Place, where we're mounting Docker volumes with user's notebooks
c.NotebookApp.notebook_dir = '/usr/local/lib/gisaf_src/notebooks'
# Allow to run Jupyter from root user inside Docker container
c.NotebookApp.allow_root = True
# Setting up Jupyter base URL
c.NotebookApp.base_url = '/notebooks/'
# Allowing Jupyter iframe embeddings
c.NotebookApp.trust_xheaders = True
c.NotebookApp.tornado_settings = {
    'headers': {
        'Content-Security-Policy': "frame-ancestors 'self' http://* https://*",
    }
}

c.NotebookApp.allow_remote_access = True
c.NotebookApp.trust_xheaders = True

c.NotebookApp.password = 'sha1:37feddeb9486:299e2c162cfdfdb0046f57f7796069701edb9c2a'
c.NotebookApp.allow_password_change = False
```

```
systemctl enable --now jupyter-gisaf.service
```

#8 - 30/10/2018 17:36 - Philippe May

Install custom package: graphene-gino, quick and dirty development on my (Phil) machine.

```
mkdir /usr/local/lib/gisaf_misc
```

Copy from my laptop:

```
rsync -a graphene-gino root@jupyter.csr.av:/usr/local/lib/gisaf_misc
```

Then:

```
cd /usr/local/lib/gisaf_misc/graphene-gino  
pip3 install -e .
```

#9 - 30/10/2018 17:58 - Philippe May

Copy gisaf config in /etc/gisaf/prod.yml.

#10 - 30/10/2018 18:04 - Philippe May

There's a change to be done on **gis2.csr.av** (which should later become Gisaf production server): proxy the attachments of dashboards to jupyter.csr.av.

In /etc/nginx/sites-enabled, add:

```
location /dashboard-attachment {  
    proxy_pass http://jupyter.csr.av/dashboard-attachment/;  
}
```

#11 - 30/10/2018 18:19 - Philippe May

Install contextily (to generate backgrounds on maps):

```
apt install cython3 libgeos-dev libproj-dev  
pip3 install contextily
```

#12 - 30/10/2018 20:34 - Philippe May

Install some Jupyter extensions:

```
jupyter labextension update --all
```

#13 - 30/10/2018 20:34 - Philippe May

Finally, go to <http://jupyter.csr.av> and enjoy :)

#14 - 10/11/2018 15:57 - Philippe May

- *Status changed from New to Resolved*

- *% Done changed from 0 to 100*

Add backup config in the dom0