Installation de jupyterHub

Installation

`useradd -m -d /srv/jupyterhub jupyter`

`chsh jupyter` -> /usr/bin/fish

`sudo -i -u jupyter`

`npm install -u configurable-http-proxy`

`python3 -m pip install --user jupyterhub`

`python3 -m pip install --user oauthenticator`

`python3 -m pip install --user dockerspawner`

`set -U fish_user_paths ~/node_modules/.bin $fish_user_paths`

`set -U fish_user_paths ~/.local/bin/ $fish_user_paths`

`jupyterhub --generate-config` -> a copier en root dans /etc/jupyterhub

Configuration

```
from oauthenticator.generic import GenericOAuthenticator

## Class for authenticating users.

import os

os.environ['OAUTH2_TOKEN_URL'] =
'https://auth.ppsfleet.navy/auth/realms/Ppsfleet/protocol/openid-connect/token'
os.environ['OAUTH2_AUTHORIZE_URL'] =
'https://auth.ppsfleet.navy/auth/realms/Ppsfleet/protocol/openid-connect/auth'
os.environ['OAUTH2_USERDATA_URL'] =
'https://auth.ppsfleet.navy/auth/realms/Ppsfleet/protocol/openid-connect/userinfo'
```

```
c.JupyterHub.authenticator_class = GenericOAuthenticator
c.GenericOAuthenticator.login_service = 'Login with PPSfleet'
c.OAuthenticator.client_id = 'jupyterhub'
c.OAuthenticator.client_secret = 'secret'
c.GenericOAuthenticator.oauth_callback_url =
'https://notebook.ppsfleet.navy/hub/oauth_callback'
c.GenericOAuthenticator.userdata_url =
'https://auth.ppsfleet.navy/auth/realms/Ppsfleet/protocol/openid-connect/userinfo'
c.GenericOAuthenticator.token_url =
'https://auth.ppsfleet.navy/auth/realms/Ppsfleet/protocol/openid-connect/token'
c.GenericOAuthenticator.userdata_method = 'GET'
c.GenericOAuthenticator.userdata_params = {"state": "state"}
c.GenericOAuthenticator.username_key = "preferred_username"

## The public facing URL of the whole JupyterHub application.
#
#  This is the address on which the proxy will bind. Sets protocol, ip, base_url
#  Default: 'http://:8000'
c.JupyterHub.bind_url = 'http://127.0.0.1:8002'

## The internal port for the Hub process.
#
#  This is the internal port of the hub itself. It should never be accessed
#  directly. See JupyterHub.port for the public port to use when accessing
#  jupyterhub. It is rare that this port should be set except in cases of port
#  conflict.
#
#  See also `hub_ip` for the ip and `hub_bind_url` for setting the full bind URL.
#  Default: 8081
c.JupyterHub.hub_port = 8081
c.JupyterHub.hub_ip = '0.0.0.0'

c.JupyterHub.spawner_class = 'dockerspawner.DockerSpawner'

## Set the log level by value or name.
#  Choices: any of [0, 10, 20, 30, 40, 50, 'DEBUG', 'INFO', 'WARN', 'ERROR', 'CRITICAL']
#  Default: 30
c.Application.log_level = 20
```

```
## Set the log level by value or name.
#  See also: Application.log_level
c.JupyterHub.log_level = 'INFO'
```

Service

```
[Unit]
Description=JupyterHub
After=network.target

[Service]
Type=simple
KillMode=mixed
User=jupyter
Environment=PATH=/srv/jupyterhub/.local/bin/:/srv/jupyterhub/node_modules/.bin:/usr/local/sbin
:/usr/local/bin:/usr/bin:/bin:/var/lib/snapd/snap/bin
ExecStart=/srv/jupyterhub/.local/bin/jupyterhub -f /etc/jupyterhub/jupyterhub_config.py
WorkingDirectory=/srv/jupyterhub
Restart=on-failure

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

Révision #6
Créé 2021-02-20 23:44:01 UTC par tjiho
Mis à jour 2025-05-04 17:12:03 UTC par tjiho