IISAS FedCloud client

Content

This is a merged presentation of several abstracts:

- FedCloud client
- Dynamic DNS
- Secrets Store

For each service:

- Brief overview of existing features
- Ongoing developments and expected outcomes

FedCloud client

Viet Tran viet.tran@savba.sk
Institute of Informatics, Slovak Academy of Sciences
Slovakia

Existing features

- Simple installation and usages
  - Single command for installation, intuitive usages
  - Extensive documentation, quickstart, cheat sheet, and inline help
- Secure token management via oidc-agent or mytoken
  - Automatic retrieval/renew of access tokens
  - Tokens not exposed in plaintext
- VO and site abstractions
  - Using memorable VO and site names instead of project IDs and site URLs
- Scripting and programming
  - JSON outputs for machine processing
  - Python library

Setting up FedCloud client

- Installing FedCloud client

```
$ pip install fedcloudclient
```

- Setting OIDC access token (or mytoken, oidc-agent)

```
$ export OIDC_ACCESS_TOKEN=xxxxxxxxxxxxxxx
(or
$ export FEDCLOUD_MYTOKEN=xxxxxxxxxxxxxxxx)
```

- That is all

Listing sites in EGI Federated Cloud

```
$ fedcloud site list
100IT
BIFI
CESGA-CLOUD
CESGA
…
```

Listing VO memberships

```
$ fedcloud token list-vos
biomed
demo.fedcloud.egi.eu
fedcloud.egi.eu
mswss.ui.savba.sk
vo.access.egi.eu
vo.ai4eosc.eu
```

Executing Openstack commands

```
$ fedcloud openstack image list --site IFCA-LCG2 --vo vo.ai4eosc.eu
Site: IFCA-LCG2, VO: vo.ai4eosc.eu, command: image list
+--------------------------------------+------------------------------+--------+
| ID                                   | Name                         | Status |
+--------------------------------------+------------------------------+--------+
| 9082297e-d1cf-46a2-baaa-c09ea040ab75 | AlmaLinux-9.1                | active |
| 3da4053f-34cb-4266-bb02-5f30987d9a91 | ...
```

Execute a command on all sites

```
$ fedcloud openstack server list -i -c ID -c Name --site ALL_SITES --vo vo.ai4eosc.eu

Site: IISAS-FedCloud, VO: vo.ai4eosc.eu, command: server list -c ID -c Name
+--------------------------+-------------------------------------------------------------+
| ID                       | Name                                                        |
+--------------------------+-------------------------------------------------------------+
| 44a93c0c-7e0b-401e...    | nomad-ai4eosc-wn-a00f2202-4d85-11ee-b620-9aee49d2abae       |
| 007fa7f6-39d4-4096 ...   | nomad-ai4eosc-front-98b8c04e-4d85-11ee-b620-9aee49d2abae    |

Site: IFCA-LCG2, VO: vo.ai4eosc.eu, command: server list -c ID -c Name
+--------------------------+-------------------------------------------------------------+
| ID                       | Name                                                        |
+--------------------------+-------------------------------------------------------------+
| 36d89087-6b0e-429a-...   | host-ifca-dc1-server2                                       |
| 472f4bb3-ca66-4227-...   | host-ifca-dc1-cli2                                          |
```

Scripting and programming

FedCloud client is designed for automation:

- Native JSON outputs
- Support for different shells (Windows, Linux)
- Customizable configuration files
- Directly usable as a Python library

Setting up working environment for cloud-native clients

- Setting a working environment for external tools is easy

```
$ fedcloud site env --site IFCA-LCG2 --vo vo.ai4eosc.eu
export OS_AUTH_URL="https://api.cloud.ifca.es:5000/v3/";
export OS_AUTH_TYPE="v3oidcaccesstoken";
export OS_IDENTITY_PROVIDER="egi.eu";
export OS_PROTOCOL="openid";
export OS_PROJECT_ID="f44e296a9ea441548456d25fb5b467c9";
export OS_ACCESS_TOKEN="..."
```

Ongoing developments

Expected features for version 2.0 (in alpha status)

- Full customization for config files
  - Ability to define customized config files for national/regional federation
  - Multiple config files may exist, switching via envvar FEDCLOUD_CONFIG_FILE or option `-c`
- Logging facilities
  - Customizable logs for better debugging
- And many more small improvements

More information

https://fedcloudclient.fedcloud.eu/usage.html

https://fedcloudclient.fedcloud.eu/cheat.html

Dynamic DNS service

Dynamic DNS: basic features

- Dynamic DNS service enables
  - Registering preferred hostnames
  - Associating the hostnames with servers
  - Updating IP address without using personal credential
- With Dynamic DNS
  - Convenient access to services via hostnames instead of IPs
  - Secure access with SSL certificates
- Many projects/services are already using Dynamic DNS :-)
  - Any services with domains fedcloud.eu,
  - AI4EOSC: dev.ai4eosc.eu, cloud.ai4eosc.eu
  - ...

Demo: Dynamic DNS service

Service migration and HA via Dynamic DNS

- Simple migration of service endpoint from a failed server to the backup server in one minute
  - Implementation in bash or Python script for full automation
  - No user credentials required for running the script

- Example: Secrets Store service
  - https://vault.docs.fedcloud.eu/design.html

Extra supports on requests

- Supporting wildcards in hostnames
  - Commonly used in Kubernetes clusters or gateways
- Adding new domains (for federations, projects, ...)
  - AI4EOSC: dev.ai4eosc.eu, cloud.ai4eosc.eu
- SSL certificates for registered hostnames
  - (according to time availabilities)

Ongoing developments

- API for registering hostnames (currently only via GUI)
  - Enabling full automation
  - Requested by IM developers and others
- VO-based authorization
  - Hostnames in some domains restricted to VO members
  - Requested by many
- Issuing SSL certificates for registered hostnames in full automation
  - Avoiding quota of free SSL certificates from LetEncrypt
  - Closing the gap and enabling full automation of service deployment

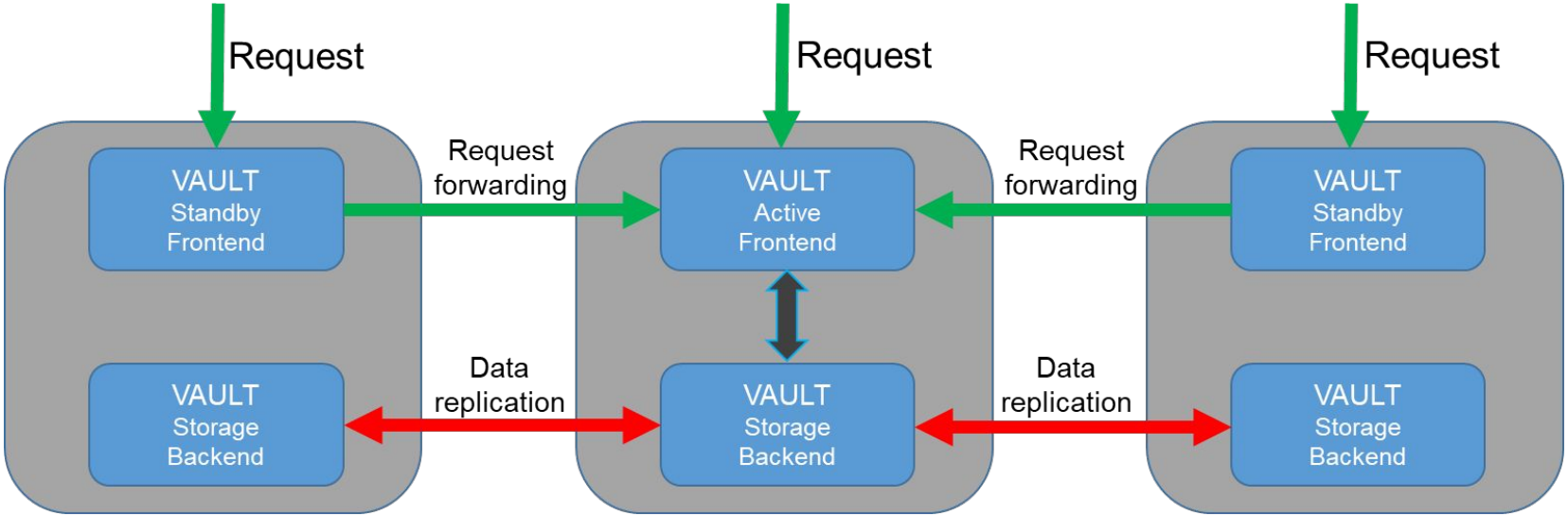
Secrets Store service

Existing features

- High availability setup with three nodes located at IISAS, INFN and IFCA
  - Geographically distributed for minimizing downtimes due disaster events
- Dedicated client as a module for FedCloud client
  - Utilizing full ecosystem, e.g. token management, VO abstractions
- Additional advanced features for enhancing securities
  - Client-side encryption
  - Lockers for secret delivery without access tokens

High-availability setup

Three nodes, geographically distributed at IISAS (Slovakia), INFN (Italy) and IFCA (Spain)



Universal endpoint via Dynamic DNS

- Three endpoints, each can serve user requests:
  - https://vault-iisas.services.fedcloud.eu:8200 (IISAS)
  - https://vault-infn.services.fedcloud.eu:8200 (INFN)
  - https://vault-ifca.services.fedcloud.eu:8200 (IFCA)

- Main, universal endpoint https://vault.services.fedcloud.eu:8200 is assigned to IFCA or INFN endpoint via Dynamic DNS

- NEW: new universal endpoint https://secrets.egi.eu/

Easy-to-use client

- Authentication via access tokens (integrated with oidc-agent and mytoken)
- Working out of the box, no setup
- Simple, easy-to-use commands

```
$ fedcloud secret put my_app_secrets mysql_password=123456 admin_password=abcdef

$ fedcloud secret list
my_app_secrets

$ fedcloud secret get my_app_secrets
key             value
--------------  -------
admin_password  abcdef
mysql_password  123456
```

Client-side encryption

- Users may need to store very sensitive secrets that absolutely nobody else can read them, by any means
  - Access tokens may be compromised
  - 2FA authentications are not suitable for automation
  - Solution: users encrypt the secrets before uploading
  - Very easy to use, fully automatic and transparent

```
$ fedcloud secret put certificate cert=@hostcert.pem key=@hostkey.pem --encrypt-key
my-secret-passphrase

$ fedcloud secret get certificate cert --decrypt-key my-secret-passphrase
```

- Security tips: use different passphrases for different secrets

Summary

Summary

- The tool and services are working in full production in EGI Federated Cloud
- A lot of efforts were put in making them easy to use
- Ongoing developments according to user requests

Additional info

- Homepage: https://www.fedcloud.eu/
- FedCloud client: https://fedcloudclient.fedcloud.eu/
- Dynamic DNS: https://docs.egi.eu/users/compute/cloud-compute/dynamic-dns/
- Secrets Store: https://vault.docs.fedcloud.eu/