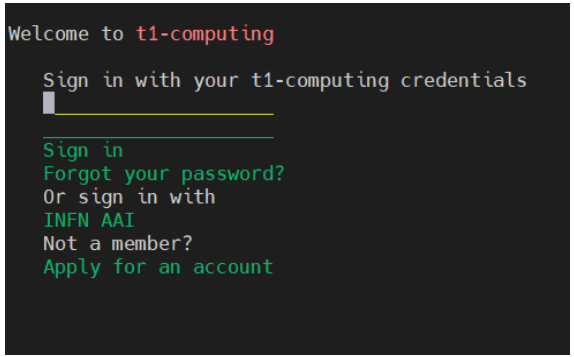
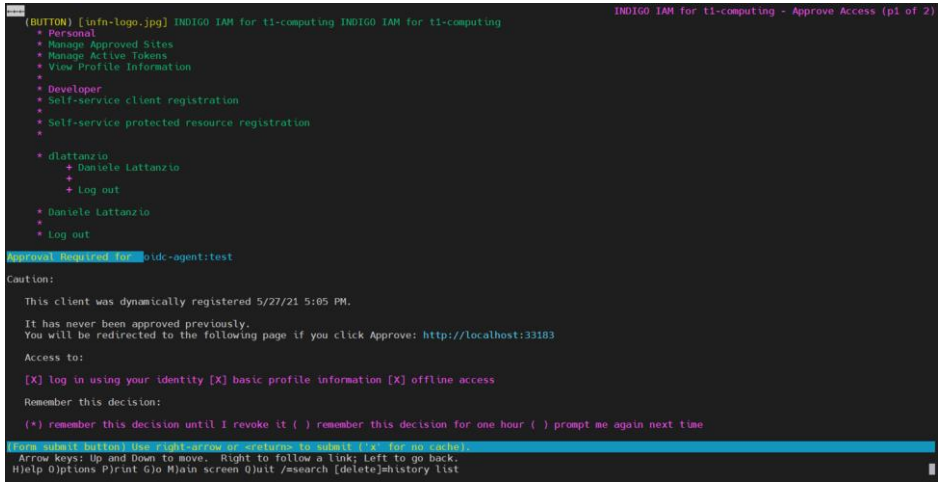
Vault & htgettoken

User Support group

Background

- Soluzione in ottica passaggio da certificati X509 a JWT tokens
- Situazione attuale gestione token: IAM + OIDC agent.
- Fornire maggiore sicurezza per storage token
- Immediatezza: ogni utente deve registrare un client da command line
  - procedura tramite browser da terminale non sempre intuitiva





Vault & htgettoken

- https://github.com/fermitools/htvault-config
- https://github.com/fermitools/htgettoken
- https://github.com/fermitools/htgettoken/files/6063416/CHEP21_Paper_Htgettoken.pdf

- Pacchetto sviluppato al FNAL da Dave Dykstra
- Hashicorp Vault: software gestione segreti - utile per storage dei refresh token
- Htgettoken: command line tool per richiedere token interagendo con un server Vault.

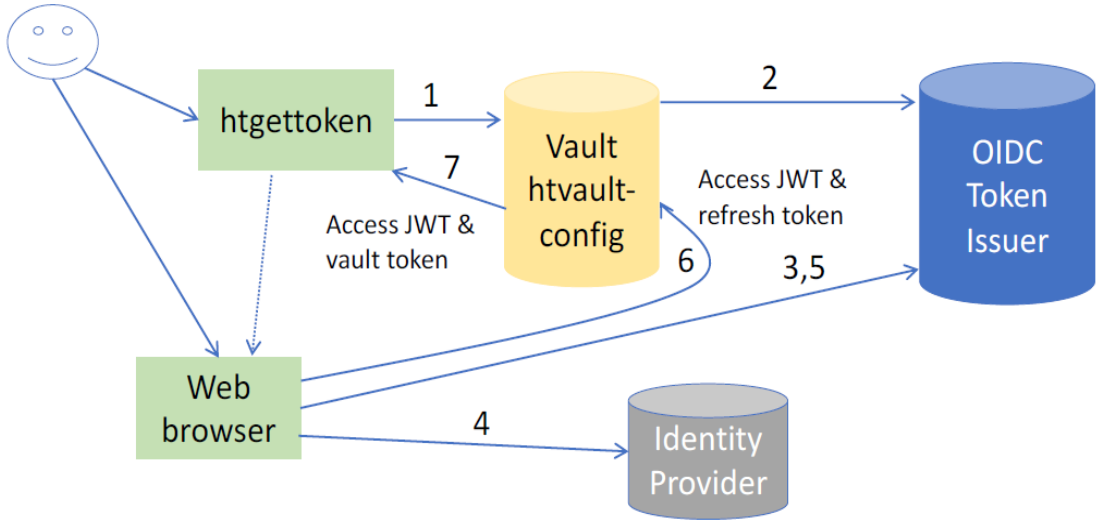
File di configurazione:

- ➢ Un client per ciascun IAM
- ➢ No registrazione client da parte dell'utente
- ➢ utente effettua iscrizione a IAM

```
MYFQDN="`uname -n`"
ISSUERS="default xdc indigo"
default_OIDC_CLIENT_ID="***"
default_OIDC_CLIENT_SECRET="***"
default_OIDC_SERVER_URL="https://iam-t1-computing.cloud.cnaf.infn.it/"
default_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
default_OIDC_CALLBACKMODE="device"
default_OIDC_USERCLAIM="preferred_username"

xdc_OIDC_CLIENT_ID="***"
xdc_OIDC_CLIENT_SECRET="***"
xdc_OIDC_SERVER_URL="https://iam.extreme-datacloud.eu/"
xdc_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
xdc_OIDC_CALLBACKMODE="device"
xdc_OIDC_USERCLAIM="preferred_username"

indigo_OIDC_CLIENT_ID="***"
indigo_OIDC_CLIENT_SECRET="***"
indigo_OIDC_SERVER_URL="https://ds-518.cr.cnaf.infn.it/"
indigo_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
indigo_OIDC_CALLBACKMODE="device"
indigo_OIDC_USERCLAIM="preferred_username"
```

Passi principali

1. OIDC flow tramite web browser:
   - Reindirizzamento a IAM → autenticazione via IDP → autorizzazione del client



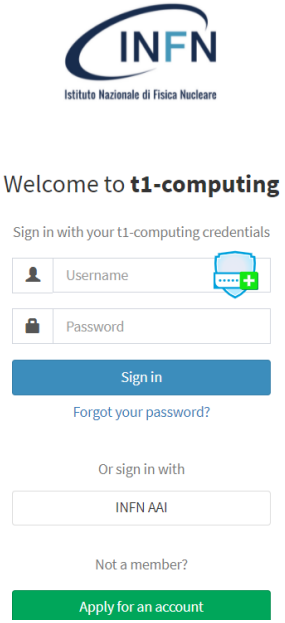
Passi principali

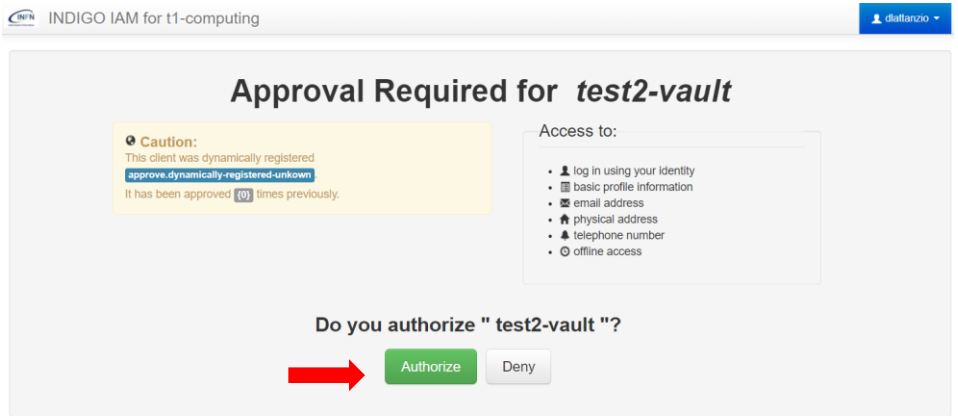
1. OIDC flow tramite web browser:
   - Reindirizzamento a IAM → autenticazione via IDP → autorizzazione del client

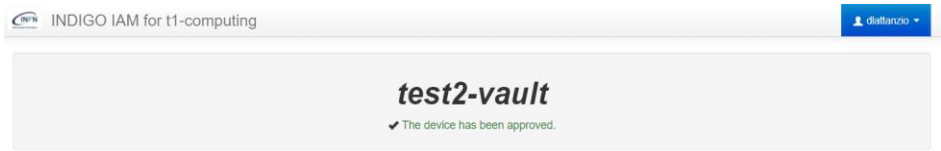
```
[dlattanzio@us-ops ~]$
[dlattanzio@us-ops ~]$ htgettoken -v -a vm-131-154-161-16.cr.cnaf.infn.it -i default
Attempting OIDC authentication with https://vm-131-154-161-16.cr.cnaf.infn.it:8200

Complete the authentication via web browser at:
    https://iam-t1-computing.cloud.cnaf.infn.it/device?user_code=1YIOP5

Couldn't open browser with 'xdg-open', please open URL manually
```

Passi principali

1. OIDC flow tramite web browser:
   - Reindirizzamento a IAM → autenticazione via IDP → autorizzazione del client







Passi principali

1. OIDC flow tramite web browser:
   - Reindirizzamento a IAM → autenticazione via IDP → autorizzazione del client

```
[dlattanzio@us-ops ~]$ htgettoken -v -a vm-131-154-161-16.cr.cnaf.infn.it -i default
Attempting OIDC authentication with https://vm-131-154-161-16.cr.cnaf.infn.it:8200

Complete the authentication via web browser at:
    https://iam-t1-computing.cloud.cnaf.infn.it/device?user_code=1YIOP5

Couldn't open browser with 'xdg-open', please open URL manually

Storing vault token in /tmp/vt_u10164
Saving credkey to /home/USER-SUPPORT/dlattanzio/.config/htgettoken/credkey-default-default: dlattanzio
Saving refresh token to https://vm-131-154-161-16.cr.cnaf.infn.it:8200
  at path secret/oauth-default/creds/dlattanzio:default
Getting bearer token from https://vm-131-154-161-16.cr.cnaf.infn.it:8200
  at path secret/oauth-default/creds/dlattanzio:default
Storing bearer token in /run/user/10164/bt_u10164
[dlattanzio@us-ops ~]$
```

Passi principali

2. Rinnovo bearer token tramite vault token:

```
[dlattanzio@us-ops ~]$ htgettoken -v -a vm-131-154-161-16.cr.cnaf.infn.it -i default
Credkey from /home/USER-SUPPORT/dlattanzio/.config/htgettoken/credkey-default-default: dlattanzio
Attempting to get bearer token from https://vm-131-154-161-16.cr.cnaf.infn.it:8200
  using vault token from /tmp/vt_u10164
  at path secret/oauth-default/creds/dlattanzio:default
Storing bearer token in /run/user/10164/bt_u10164
[dlattanzio@us-ops ~]$
```

Passi principali

3. Richiesta nuovo vault token tramite kerberos:

   - L'utente ottiene ticket kerberos tramite comando kinit

```
[dlattanzio@us-ops ~]$ kinit -V dlattanzio
Using existing cache: persistent:10164:krb_ccache_uKYF473
Using principal: dlattanzio@CR.CNAF.INFN.IT
Password for dlattanzio@CR.CNAF.INFN.IT:
Authenticated to Kerberos v5
[dlattanzio@us-ops ~]$
```

   - Autenticazione su vault tramite kerberos e ottenimento dei nuovi token

```
[dlattanzio@us-ops ~]$ htgettoken -v -a vm-131-154-161-16.cr.cnaf.infn.it -i default
Credkey from /home/USER-SUPPORT/dlattanzio/.config/htgettoken/credkey-default-default: dlattanzio
Initializing kerberos client for host@vm-131-154-161-16.cr.cnaf.infn.it
Negotiating kerberos with https://vm-131-154-161-16.cr.cnaf.infn.it:8200
Storing vault token in /tmp/vt_u10164
Attempting to get bearer token from https://vm-131-154-161-16.cr.cnaf.infn.it:8200
  at path secret/oauth-default/creds/dlattanzio:default
Storing bearer token in /run/user/10164/bt_u10164
[dlattanzio@us-ops ~]$
```

Configurazione kerberos

```
MYFQDN="`uname -n`"
ISSUERS="default xdc indigo"
default_OIDC_CLIENT_ID="***"
default_OIDC_CLIENT_SECRET="***"
default_OIDC_SERVER_URL="https://iam-t1-computing.cloud.cnaf.infn.it/"
default_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
default_OIDC_CALLBACKMODE="device"
default_OIDC_USERCLAIM="preferred_username"

xdc_OIDC_CLIENT_ID="***"
xdc_OIDC_CLIENT_SECRET="***"
xdc_OIDC_SERVER_URL="https://iam.extreme-datacloud.eu/"
xdc_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
xdc_OIDC_CALLBACKMODE="device"
xdc_OIDC_USERCLAIM="preferred_username"

indigo_OIDC_CLIENT_ID="***"
indigo_OIDC_CLIENT_SECRET="***"
indigo_OIDC_SERVER_URL="https://ds-518.cr.cnaf.infn.it/"
indigo_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
indigo_OIDC_CALLBACKMODE="device"
indigo_OIDC_USERCLAIM="preferred_username"

LDAPURL="ldap://ldap-m.cr.cnaf.infn.it"
LDAPDN="ou=people,ou=cnaf,o=infn,c=it"
LDAPATTR="uid"
KERBEROSPOLICYDOMAIN="@cr.cnaf.infn.it"
```

```
[root@vm-131-154-161-16 ~]# cat /etc/krb5.conf
#
# /etc/krb5.conf configuration file managed by puppet.
# Please do not edit - all manual changes will be overwritten!
#

includedir /etc/krb5.conf.d/

includedir /var/lib/sss/pubconf/krb5.include.d/
[logging]
 default = FILE:/var/log/krb5libs.log
 kdc = FILE:/var/log/krb5kdc.log
 admin_server = FILE:/var/log/kadmind.log

[libdefaults]
 default_realm = CR.CNAF.INFN.IT
 dns_lookup_realm = false
 ticket_lifetime = 24h
 renew_lifetime = 7d
 forwardable = true
 pkinit_anchors = /etc/pki/tls/certs/ca-bundle.crt

 # default_realm = EXAMPLE.COM
 default_ccache_name = KEYRING:persistent:%{uid}
 dns_lookup_kdc = false

[realms]
 CR.CNAF.INFN.IT = {
  kdc = kdc-m.cr.cnaf.infn.it
  kdc = kdc-s.cr.cnaf.infn.it
  admin_server = kdc-m.cr.cnaf.infn.it
 }


[domain_realm]
 cr.cnaf.infn.it = CR.CNAF.INFN.IT
 .cr.cnaf.infn.it = CR.CNAF.INFN.IT

[root@vm-131-154-161-16 ~]#
```

Nota su kerberos:

Test di kerberos + vault + htgettoken in job HTCondor (si ringrazia Federico Fornari):

1. Richiesta bearer token a iam-t1-computing. Si autorizza il client e si ottengono vault token, bearer token e credkey file che vengono messi in gpfs per poter essere utilizzati dal job.

```
[fornariborexino@ui-tier1 htgettoken]$ cat create-htgettoken-token-files.sh
#!/bin/bash
export PREFIX=/storage/gpfs_data/borexino/users/fornariborexino
$PREFIX/htgettoken/htgettoken \
 --cafile=$PREFIX/geant-ov-rsa-ca.pem \
 --configdir=$PREFIX/htgettoken \
 --vaulttokenfile=$PREFIX/htgettoken/vt \
 -a vm-131-154-161-16.cr.cnaf.infn.it -i "$1"
[fornariborexino@ui-tier1 htgettoken]$ ./create-htgettoken-token-files.sh default
Attempting OIDC authentication with https://vm-131-154-161-16.cr.cnaf.infn.it:8200

Complete the authentication via web browser at:
   https://iam-t1-computing.cloud.cnaf.infn.it/device?user_code=F0PABN

Couldn't open browser with 'xdg-open', please open URL manually

Storing vault token in /storage/gpfs_data/borexino/users/fornariborexino/htgettoken/vt
Saving credkey to /storage/gpfs_data/borexino/users/fornariborexino/htgettoken/credkey-default-default
Saving refresh token ... done
Attempting to get token from https://vm-131-154-161-16.cr.cnaf.infn.it:8200 ... succeeded
Storing bearer token in /run/user/33615/bt_u33615
[fornariborexino@ui-tier1 htgettoken]$
```

Nota su kerberos:

2. Sottomissione di un job di test va a buon fine

3. Alla scadenza del vault token, htgettoken tenta di ottenerne uno nuovo via kerberos sfruttando il credkey file. L'operazione fallisce poiché non gli è stato fornito il ticket di kerberos

4. Risottomettiamo modificando l'eseguibile, facendo sì che il job si prenda il ticket di kerberos. Otteniamo ancora errore: «permission denied». Il nome utente dell'account da cui è stato sottomesso il job è diverso da quello contenuto nel credkey (cioè lo username di iam-t1-computing). Lo username kerberos è fornariborexino, mentre quello IAM è ffornari. Dovrebbero coincidere?

5. Risposta: sì. Per verificarlo è stato instanziato uno iam di test amministrabile, in cui è stato registrato un account con username uguale a quello dell'account kerberos (fornariborexino).

Nota su kerberos:

6. Infine è stato configurato vault per interfacciarsi con questo IAM:

```
MYFQDN="`uname -n`"
ISSUERS="default xdc indigo"
default_OIDC_CLIENT_ID="***"
default_OIDC_CLIENT_SECRET="***"
default_OIDC_SERVER_URL="https://iam-t1-computing.cloud.cnaf.infn.it/"
default_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
default_OIDC_CALLBACKMODE="device"
default_OIDC_USERCLAIM="preferred_username"

xdc_OIDC_CLIENT_ID="***"
xdc_OIDC_CLIENT_SECRET="***"
xdc_OIDC_SERVER_URL="https://iam.extreme-datacloud.eu/"
xdc_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
xdc_OIDC_CALLBACKMODE="device"
xdc_OIDC_USERCLAIM="preferred_username"

indigo_OIDC_CLIENT_ID="***"
indigo_OIDC_CLIENT_SECRET="***"
indigo_OIDC_SERVER_URL="https://ds-518.cr.cnaf.infn.it/"
indigo_OIDC_SCOPES="openid,profile,email,address,phone,offline_access"
indigo_OIDC_CALLBACKMODE="device"
indigo_OIDC_USERCLAIM="preferred_username"

LDAPURL="ldap://ldap-m.cr.cnaf.infn.it"
LDAPDN="ou=people,ou=cnaf,o=infn,c=it"
LDAPATTR="uid"
KERBEROSPOLICYDOMAIN="@cr.cnaf.infn.it"
```

Nota su kerberos:

Risultato: Questa volta il credkey file contiene “fornariborexino”, e, con il ticket kerberos a disposizione, l’autenticazione via kerberos va a buon fine. Il vault token è ricreato per richiedere con successo il bearer token.

Conclusione

Essenzialmente ci sono 2 complicazioni:

1) Occorre passare al job anche il ticket kerberos.

2) esiste un aspetto negativo legato al fatto che lo username dell’account kerberos e lo username dell’account iam debbano coincidere. In generale un utente che ha un singolo account su IAM potrebbe appartenere a più esperimenti ed avere più account kerberos.

Poiché quindi la validità del vault token è 1 settimana (di default), questo sembra un tempo ragionevole per completare le operazioni di un job. Pertanto il rinnovo tramite kerberos potrebbe diventare non indispensabile