Client LDAP CentoOS/RHEL 8

Da tempo orami CentOS, ovvero “**C**ommunity **ENT**erprise **O**perating **S**ystem”, è alla versione 8 in upstream mode, dunque anche RHEL “**R**ed **H**at **E**ntrprise **L**inux” è alla medesima versione.

Vediamo nell’articolo come integrare un server CentOS 8 o RHEL 8 con OpenLDAP per abilitarlo ad accedervi con autenticazione centralizzata. Tra le prime cose che notiamo “sparisce” nscd in favore di SSSD.

SSSDè l’acronimo di **System Security Services Daemon** (link: https://sssd.io/), ovvero il *demone dei servizi di sicurezza del sistema,* un software originariamente sviluppato per il sistema operativo Linux che fornisce un set di demoni per gestire l’accesso ai servizi di directory services e ai meccanismi di autenticazione. Gli inizi di SSSD risiedono nel progetto software open-source FreeIPA (Identity, Policy and Audit).

Aggiornare il sistema tramite dnf o yum

Per prima cosa bisogna aggiornare il sistema operativo, per farlo, usare i seguenti comandi:

```
dnf update -y
oppure
yum update -y
```

Installazione SSSD su CentOS/RHEL 8

Dopo che l’aggiornamento è andato a buon fine installare SSSD e gli SSSD tools.

```
dnf install sssd sssd-tools -y
oppure
yum install sssd sssd-tools -y
```

Configurazione SSSD per autenticazione OpenLDAP su CentOS/RHEL 8

Prossimo passo è configurare SSSD per abilitare l'autenticazione OpenLDAP sul sistema locale.

SSSD non usa una sua configurazione di default, il file quindi non esisterà, crearne uno manualmente:

vim /etc/sssd/sssd.conf

Copiare il contenuto seguente nel file **sssd.conf**. Assicuriamoci di aver sostituito i parametri con i valori corretti del nostro ambiente.

```
[sssd]
services = nss, pam, sudo
config_file_version = 2
domains = default

[sudo]

[nss]

[pam]
offline_credentials_expiration = 60

[domain/default]
ldap_id_use_start_tls = True
cache_credentials = True
ldap_search_base = dc=ldap,dc=acme,dc=com
id_provider = ldap
auth_provider = ldap
chpass_provider = ldap
access_provider = ldap
sudo_provider = ldap
ldap_uri = ldap://ad.acme.com
ldap_default_bind_dn                                    =
cn=readonly,ou=system,dc=ldap,dc=acme,dc=com
ldap_default_authtok = hjsd4Rtsd
ldap_tls_reqcert = demand
```

```
ldap_tls_cacert = /etc/openldap/cacerts/ca.cert.pem
ldap_tls_cacertdir = /etc/openldap/cacerts/
ldap_search_timeout = 50
ldap_network_timeout = 60
ldap_sudo_search_base = ou=SUDOers,dc=ldap,dc=acme,dc=com
ldap_access_order = filter
ldap_access_filter = (objectClass=posixAccount)
```

Salvare ed uscire dal file. Nota che abbiamo anche configurato il nostro server OpenLDAP per fornire i diritti sudo come mostrato dalle configurazioni;

```
services = nss, pam, sudo
...

[sudo]
...

ldap_sudo_search_base = ou=SUDOers,dc=acme,dc=com
```

Se non usi OpenLDAP per le regole di sudo, puoi rimuovere queste configurazioni.

ldap_tls_cacert nel file sssd.conf sopra.

Ora scarica il certificato CA del server OpenLDAP e salvalo nel file specificato dalla direttiva ldap_tls_cacert del file sssd.conf.

```
openssl s_client -connect ldap.acme.com:636 -showcerts < /dev/null | openssl x509 -text
```

copia il certificato ed incollalo nel file **/etc/openldap/cacerts/ca.cert.pem**.

```
vim /etc/openldap/cacerts/ca.cert.pem

-----BEGIN CERTIFICATE-----
MIIFxzCCA6+gAwIBAgIUV+l4aOvMCLlNQRKOpt9YfxcxA8MwDQYJKoZIhvcNAQEL
BQAwczELMAkGA1UEBhMCS0UxEDAOBgNVBAgMB05haXJvYmkxDDAKBgNVBAcMA05h
```

```
...
...
5deiMlJkrYv7wZ0prq0QO5lduGBuD9UJvRa8LBV0GEAiHZL5PJOnREHObbAH90
7E
eixIJpkcC4wguMaXDNqIv6WGdQtRUyIP8tdByXYJGrbRW0K/K9qEaIZhJiAES1
Qy
8U96RdYBpLvDctRch1kIfvnAVffTxmObAGI9n64O89p48kocJwNI/XQNRg==
-----END CERTIFICATE-----
```

Ora apriamo il file /etc/openldap/ldap.conf e configuriamo le linee seguenti:

vim /etc/openldap/ldap.conf

Fondamentalmente, devi definire la posizione del certificato CA, la base di ricerca OpenLDAP, l'URI e se stai fornendo le direttive SUDO tramite OpenLDAP anche la base SUDOers.

```
BASE    dc=ldap,dc=acme,dc=com
URI     ldaps://ldap.acme.com:636
SUDOERS_BASE    ou=SUDOers,dc=ldap,dc=acme,dc=com
...
...
TLS_CACERT      /etc/openldap/cacerts/ca.cert.pem
...
```

Salva ed esci dal file di configurazione.

Configurare il Name Service Switch e PAM su CentOS/RHEL 8

Successivamente, è necessario aggiornare NSS e PAM per utilizzare SSSD per gestire le risorse di autenticazione

Nelle precedenti versioni di CentOS potevamo usare il tool **authconfig** ma è stato sostituito da **authselect**.

Authselect è un'utilità che semplifica la configurazione dell'autenticazione dell'utente, soprattutto durante l'utilizzo dell'SSSD.

Configurare il profilo SSSD.

Il comando Authselect quando viene utilizzato per creare un profilo SSSD modificherà i seguenti file:

- /etc/pam.d/system-auth
- /etc/pam.d/password-auth
- /etc/pam.d/fingerprint-auth
- /etc/pam.d/smartcard-auth
- /etc/pam.d/postlogin
- /etc/nsswitch.conf

Pertanto, esegui un backup di questi file nel caso in cui le cose non funzionino. Dopo aver salvato il backup di questi file rimuovili in modo che il comando successivo li possa ricostruire.

Crea un profilo SSSD. Questo comando avrà esito positivo solo se hai rimosso i file sopra.

```
authselect select sssd
```

In caso contrario, puoi sovrascrivere i file aggiungendo l'opzione --force.

```
authselect select sssd --force
```

```
Backup stored at /var/lib/authselect/backups/2022-03-23-11-52-12.yMO4TA
Profile "sssd" was selected.
The following nsswitch maps are overwritten by the profile:
- passwd
- group
- netgroup
- automount
- services

Make sure that SSSD service is configured and enabled. See SSSD documentation for more information.
```

Quindi, affinché il sistema possa recuperare i diritti sudo da SSSD/OpenLDAP modifica il file /etc/nsswitch.conf inserendo la

riga seguente:

```
echo "sudoers:    files sss" >> /etc/nsswitch.conf
```

Configurae la creazione automatica della Home Directory

Per abilitare la creazione automatica della home directory al primo accesso dobbiamo installare **oddjob-mkhomedir** il quale fornisce il modulo **pam_oddjob_mkhomedir**.

```
dnf install oddjob-mkhomedir -y
o
yum install oddjob-mkhomedir -y
```

Avviare oddjobd ed inserirlo per essere avviato allo start del sistema

```
systemctl enable --now oddjobd
```

Carica il modulo **pam_oddjob_mkhomedir** nel file di autenticazione PAM /etc/pam.d/system-auth per abilitare la creazione automatica della home directory.

```
echo "session optional pam_oddjob_mkhomedir.so skel=/etc/skel/ umask=0022" >> /etc/pam.d/system-auth
```

Restart oddjobd.

```
systemctl restart oddjobd
```

Running SSSD

Before you can start SSSD, you need to check configuration for any typos or permissions;

Prima di avviare SSSD verifica che la confiugrazione non abbia errori nei file o permessi sbagliati:

```
sssctl config-check
```

File ownership and permissions check failed. Expected root:root and 0600.

Come suggerisce l'output diamo l'accesso in read/write (600) al file /etc/sssd/ al solo proprietario del file (root).

```
chown -R root: /etc/sssd
```

```
chmod 600 -R /etc/sssd
```

La configurazione è ora corretta, possiamo impostare SSSD per essere avviato al boot.

```
systemctl enable --now sssd
```

Controliamo lo stato

```
systemctl status sssd
```

```
● sssd.service - System Security Services Daemon
     Loaded: loaded (/usr/lib/systemd/system/sssd.service; enabled; vendor preset: enabled)
   Active: active (running) since Sun 2019-12-08 16:57:07 EAT; 42min ago
 Main PID: 779 (sssd)
    Tasks: 3 (limit: 5073)
   Memory: 60.6M
   CGroup: /system.slice/sssd.service
           ├─779 /usr/sbin/sssd -i --logger=files
             ├─800 /usr/libexec/sssd/sssd_be --domain implicit_files --uid 0 --gid 0 --logger=files
           └─801 /usr/libexec/sssd/sssd_nss --uid 0 --gid 0 --logger=files
```

Test autenticazione OpenLDAP via SSSD

Supponendo che tu abbia già creato i tuoi utenti e gruppi OpenLDAP, verifichiamo di poter accedere, prima di tutto, vediamo se il tuo nome utente LDAP sul sistema venga riconosciuto con il comando id.

```
id domenico.tricarico
```

Dovresti ottenere una voce simile a;

```
uid=1002(domenico.tricarico)   gid=1002(domenico.tricarico)
groups=1002(sysadmin)
```

Se non riesci a trovare l’utente assicurati di controllare i log syslog e i log sssd. Altrimenti, puoi riavviare sssd

```
systemctl restart sssd
```

Controlla di nuovo l’utente con il comando id.

Se tutto va bene, esegui un accesso ssh locale per testare l’autenticazione LDAP

```
ssh -l domenico.tricarico 127.0.0.1
oppure
ssh domenico.tricarico@127.0.0.1
```

```
The authenticity of host '127.0.0.1(::1)' can't be
established.
ECDSA key fingerprint is
SHA256:iMRNJQa8gU0t6fHx6nzmAU+ZygA/3J2BC6zzwzqfY4o.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '127.0.0.1' (ECDSA) to the list of
known hosts.
domenico.tricarico@localhost's password:
[domenico.tricarico@ldapserver ~]$ pwd
/home/domenico.tricarico
```

Verifica diritti sudo.

Innanzitutto, se hai assegnato all’utente i diritti sudo, puoi verificare eseguendo il comando seguente sul tuo server OpenLDAP:

```
export SUDOERS_BASE=ou=SUDOers,dc=ldap,dc=acme,dc=com
```

```
ldapsearch -b "$SUDOERS_BASE" -D
cn=admin,dc=ldap,dc=acme,dc=com -W -x
```

```
...
# sudo, SUDOers, ldap.acme.com
```

```
dn: cn=sudo,ou=SUDOers,dc=ldap,dc=acme,dc=com
objectClass: top
objectClass: sudoRole
cn: sudo
sudoUser: domenico.tricarico
sudoHost: ALL
sudoRunAsUser: ALL
sudoCommand: ALL
...
```

Ora proviamo il sudo:

```
[domenico.tricarico@ldapserver ~]$ sudo su -

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

    #1) Respect the privacy of others.
    #2) Think before you type.
    #3) With great power comes great responsibility.

[sudo] password for domenico.tricarico: Last login: Wed Mar 23 13:49:47 EAT 2022 from 172.16.0.15 on pts/0 [root@ldapserver ~]#
```