Manually Connecting an SSSD Client to an Active Directory Domain

Updated May 9 2017 at 2:20 PM - English ▾ ()

The recommended way to configure a System Security Services Daemon (SSSD) client to an Active Directory (AD) domain is using the **realmd** suite. See the Windows Integration Guide (https://access.redhat.com/documentation/en-US/Red_Hat_Enterprise_Linux/7/html-single/Windows_Integration_Guide/index.html#sssd-ad-proc).

If you do not want to use **realmd**, this procedure describes how to configure the system manually.

1. Set up the Linux system as an AD client and enroll it within the AD domain. This is done by configuring the Kerberos and Samba services on the Linux system.
    i. Install the following packages:

```
# yum install krb5-workstation samba-common-tools sssd-ad
```

    ii. Set up Kerberos to use the AD Kerberos realm.
        i. Open the Kerberos client configuration file.

```
# vim /etc/krb5.conf
```

        ii. Configure the `[logging]` and `[libdefaults]` sections so that they connect to the AD realm.

```
[logging] default = FILE:/var/log/krb5libs.log [libdefaults] default_realm =
EXAMPLE.COM dns_lookup_realm = true dns_lookup_kdc = true ticket_lifetime = 24h
renew_lifetime = 7d rdns = false forwardable = yes
```

        If auto-discovery is not used with SSSD, then also configure the `[realms]` and `[domain_realm]` sections to explicitly define the AD server.

    iii. Configure the Samba server to connect to the AD server.
        i. Open the Samba configuration file.

```
# vim /etc/samba/smb.conf
```

ii. Set the AD domain information in the `[global]` section.


```
[global] workgroup = EXAMPLE client signing = yes client use spnego = yes
kerberos method = secrets and keytab log file = /var/log/samba/%m.log password
server = AD.EXAMPLE.COM realm = EXAMPLE.COM security = ads
```

iv. Add the Linux machine to the AD domain.

i. Obtain Kerberos credentials for a Windows administrative user.

```
# kinit Administrator
```

ii. Add the machine to the domain using the `net` command.

```
# net ads join -k Joined 'server' to dns domain 'example.com'
```

This creates a new keytab file, `/etc/krb5.keytab` .

List the keys for the system and check that the host principal is there.

```
# klist -k
```

2. If necessary, install the `oddjob-mkhomedir` package to allow SSSD to create home directories for AD users.

```
# yum install oddjob-mkhomedir
```

3. Use `authconfig` to enable SSSD for system authentication. Use the `--enablemkhomedir` to enable SSSD to create home directories.

```
# authconfig --update --enablesssd --enablesssdauth --enablemkhomedir
```

4. Open the SSSD configuration file.

```
# vim /etc/sssd/sssd.conf
```

5. Configure the AD domain.

i. In the `[sssd]` section, add the AD domain to the list of active domains. This is the name of the domain entry that is set in *[domain/NAME]* in the SSSD configuration file.

Also, add `pac` to the list of services; this enables SSSD to set and use MS-PAC information on tickets used to communicate with the AD domain.

```
[sssd] config_file_version = 2 domains = ad.example.com services = nss, pam, pac
```

ii. Create a new domain section at the bottom of the file for the AD domain. This section has the format *domain/NAME*, such as `domain/EXAMPLE` . For each provider, set the value to `ad` , and give the connection information for the specific AD instance to connect to.

```
[domain/AD.EXAMPLE] id_provider = ad auth_provider = ad chpass_provider = ad
access_provider = ad
```

iii. Enable credentials caching; this allows users to log into the local system using cached information, even if the AD domain is unavailable.

```
cache_credentials = true
```

Tags **active_directory (/tags/active_directory)** **ipa (/tags/ipa)** **sssd (/taxonomy/tags/sssd)**

Article Type **General (/article-type/general)**
