SSSD

* Identity resolution - NNS module
* Authentication – PAM module
* Caching for offline access and reduced database processing
* Multiple sources in single configuration
  + Common sources: LDAP. AD, KRB

It is divided into several components

* NSS Client (For user look up details on Unix)
  + When clients of NSS like “getent passwd user” makes a request, the request is passed to “/lib64/libnss\_sss.so”
    - Configuration file for NSS
      * /etc/nsswitch.conf
* PAM enabled Client (For Authentication)
  + When clients of PAM like “ssh user@host” makes a request, the request is passed to “/lib64/security/pam\_sss.so”
    - Configuration file for PAM
      * /etc/pam.d
* Plugins supported by SSSD
  + eDirectory LDAP
  + OpenLDAP (Krb5)
  + Active Directory
  + IPA

**The information SSSD collects is store in a cached file like “/var/lib/sss.db”.**

Steps to configure SLES12 to resolve and authenticate users in Active Directory using the AD back-end plugin

## Join SLES 12 server to Active Directory Domain

To join SLES 12 server to the AD Domain we will use Samba and Kerberos. To complete the configuration, information from the windows server needs to be gathered. After this the below configuration files must be modified

* Kerberos
  + /etc/krb5.conf
* Samba
  + /etc/samba/smb.conf
* Hosts
  + /etc/hosts

The hosts file will be updated with windows server information since we won’t be using DNS.

* Under **Tools > Active Directory > Users & Computers** you will find the domain name
* On windows server **CMD** type “ipconfig /all”. It will give you the IP address of the windows server

We will now have the Windows AD information that we need for our configuration.

**Windows AD Domain Information**

* **Domain =** AD.DOMAIN.COM
* **Windows Server Name =** WIN2012SRV
* **Windows Server IPADDRESS =** 192.168.157.131
* **AD Administrator =** cn=Administrator.users.ad.domain.com

Later, we will create our test user = Jane Doe / jdoe

### STEPS

One method of connecting the SLES server to the Windows domain is to use the samba client. The Kerberos client and the Samba client will be needed for this method.

1. Refresh the repository
   1. zypper ref
2. Install krb5-client
   1. zypper in krb5-client
3. Install samba-client
   1. zypper in samba-client
   2. zipper in samba-winbind
4. Configure Kerberos
   1. Take a backup of the krb5.conf file before making the edits
   2. Configure /etc/krb5.conf with the domain information we gathered in previous steps
      1. When the DNS resolution isn’t configured remember to update the setting as below to prevent errors resulting from reversed DNS resolution
         1. rdns = false in the [libdefaults] section
   3. Update the necessary properties as shown in the document
5. Configure Samba
   1. Take a backup of the smb.conf file before making the edits
   2. Configure /etc/samba/smb.conf

* 1. 

1. Configure hosts file
   1. 192.168.157.131 win2012srv win2012srv.ad.domain.com ad ad.domain.com
2. Join the SLES 12 Server to the AD domain
   1. Authenticate to the AD domain by typing the below command
      1. kinit Administrator
   2. Create SLES server as a computer in the AD domain by typing the below command
      1. net ads join -k **(Ignore the errors regarding the DNS. Key point is that we should see that the domain is joined)**
3. Verify that the SLES server is a part of the domain byb going in to users and computers > computers and you should see the SLES12SP-DEMO computer what we have joined.
4. Now create a test user named “Jane Doe” and give it a uid as “jdoe”
   1. Click on password never expires
5. Test GSSAPI connectivity with ldapsearch
   * 1. /usr/bin/ldapsearch -H ldap://win2012srv.ad.domain.com/ -Y GSSAPI -N -b "dc=ad,dc=domain,dc=com" "(&(objectClass=user)(sAMAccountName=testuser))"

## Configure SSSD

1. Refresh the repository
   1. zypper ref
2. Install sssd package
   1. zypper in sssd
3. Install sssd-ad package
   1. zypper in sssd-ad
4. Modify SSSD configuration file
   1. Take a backup of the sssd.conf file before making the edits
   2. Configure file at /etc/sssd/sssd.conf

* 1. 

## Configure NSS

1. Modify NSS configuration file
   1. Take a backup of the nsswitch.conf file before making the edits
   2. Configure file at /etc/nsswitch.conf
2. Update the below properties accordinglyvi /e
   1. passwd: files sss
   2. group: files sss

Setting the passwd & group to the above properties causes resolution to occur first at the local filesystem and then to “sssd” through the “sss” module

1. Modify NSCD configuration file

If both ncsd and sss are caching passwd & group, the deamons can conflict and the AD users won’t be resolved.

* 1. Take a backup of the nscd.conf file before making the edits
  2. Configure file at /etc/nscd.conf
     1. Change the “enable-cache” option for **passwd** from “yes” to “no”
     2. Change the “enable-cache” option for **group** from “yes” to “no”
  3. Restart NSCD by below command
     1. systemctl restart nscd
  4. Start SSSD service
     1. Systemctl start sssd

## Configure PAM (Alternate to this process is one command “pam-config -a –sss”)

To enable authentication through SSSD to Active Directory the pam\_sss.so module need to be added to pam.d services files.

1. Modify the files in “/etc/pam.d/common files” add pam\_sss.so
2. Take backup and edit “/etc/pam.d/common-auth”
   1. Add the below property above “pam\_unix.so”
      1. auth    sufficient        pam\_sss.so     use\_first\_pass
3. Take backup and edit “/etc/pam.d/common-account”
   1. Add the below property above “pam\_unix.so”
      1. account   sufficient      pam\_sss.so    use\_first\_pass
4. Take backup and edit “/etc/pam.d/common-session”
   1. Add the below property above “pam\_unix.so”
      1. session    sufficient     pam\_sss.so     use\_first\_pass
   2. Add the below property below “pam\_unix.so”
      1. session    sufficient   pam\_mkhomedir.so
5. Take backup and edit “/etc/pam.d/common-password”
   1. Add the below property above “pam\_unix.so”
      1. password     sufficient     pam\_sss.so

### Resolution

Id <userid>

getent passwd <userid>

### Authentication

Ssh <userid>@localhost

Id, Getent & ssh can be used for name resolution and user authentication.

### Verifying

### Test if the user id is resolved by typing the below command

### Id jdoe

### We should be able to see the information about the user id and group id resolved.

### Getent passwd jdoe

### We should be able to see the information about the user id and group id resolved.

### ssh jdoe@localhost

### This will allow us to authenticate jdoe user and after submitting the password we should be able to login with user id Jdoe in his home directory.