User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

How Do I Integrate Bright With AD provider from SSSD?

There are many ways to enroll a UNIX client to an Active Directory domain. The other KB articles have generally followed a few of the 10 or so options in table 4.2 of (http://goo.gl/ON9hnx). Some of the procedures are quite long.

This article discusses the AD provider of SSSD. The AD provider provided by MS Windows provides compatibility with LDAP/Kerberos as well as AD. The AD provider provided by SSSD is a protocol-compatible version that does much of the same. It also helps avoid many of the long procedures that would otherwise be followed.

SSSD's AD provider introduces these unique features in terms of security, speed and ease of configuration:

- logins are faster as the AD provider can leverage the special tokenGroups feature
- the client machine is able to update or refresh its DNS records
- the NetBIOS domain name can be autodiscovered and used in both lookups and output format (`getent passwd AD\\Administrator` now works)
- clients are able to automatically discover the closest AD server to connect to using the 'sites' feature of AD
- the AD provider automatically discovers trusted domains in the same forest, allowing all users from the same forest to log in to the machine
- expressing access control with an LDAP filter was made much simpler with a new configuration option
- custom UPN suffixes, also known as Enterprise Principals are supported by default (in RHEL7 only. See workaround for RHEL5/6 below)

(Source: Jakub Hrozek http://goo.gl/7wCrkH)

Also note that the AD provider uses GSSAPI binds to AD LDAP for increased security.

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

How to test if AD service discovery is working?

AD clients use SRV records to find the domain controller for a given service. Dynamic DNS is an integral part of Active Directory, because domain controllers register their network service types in DNS so that other computers in the Domain (or Forest) can access them.

To test AD service discovery you can query the DNS server for the AD-specific SRV records:

```
dig srv _ldap._tcp.dc._msdcs.<Domain_Name>
```

If service discovery is not working (e.g. the domain is in a non-public TLD), then you can configure the DNS server to forward queries for that zone to domain controller:

```
# cat /etc/named.conf.include
zone "bcm.local" {
 type forward;
 forwarders { 10.2.184.236; };
};
#
```

```
CMDAemon will respect any changes made to the file /etc/named.conf.inc
lude.
```

```
After that you will need to restart the DNS server on the headnode:
```

```
service named restart
```

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

Obtain a Kerberos keytab using `adcli`

In order to get started, a Kerberos keytab file is required. There are several ways to obtain a keytab file, but the simplest way to generate one on a Unix system is via the the `adcli` utility. `adcli` is one of the building blocks of realmd, AND unfortunately, it is the only component of realmd that works on RHEL 6. `adcli` is available from the EPEL repository (http://goo.gl/xGSTwF).

If AD service is working the following command should show some basic information about the domain:

```
#  adcli info  bmc.local
[domain]
domain-name = bcm.local
domain-short = BCM
domain-forest = bcm.local
domain-controller = win2008.bcm.local
domain-controller-site = Default-First-Site-Name
domain-controller-flags = pdc gc ldap ds kdc timeserv closest writable
 good-timeserv full-secret
domain-controller-usable = yes
[computer]
computer-site = Default-First-Site-Name
#
```

If AD service is not working, you will need to specify a few extra options:

```
adcli info bmc.local -S  10.2.184.236  -D BCM.LOCAL -R BCM.LOCAL
```

-S IP/Hostname of the Domain Controller

-D Domain name

-R Kerberos realm

To join the domain:

```
adcli join bmc.local
```

or

```
adcli join bmc.local -S  win2008.bcm.local  -D BCM.LOCAL -R BCM.LOCAL
```

This `adcli` join command would prompt the user for his password and if the password is correct, it will return to the prompt without further messages. The default user is the Administrator, but any user that has permission to join a new machine to the domain can be used. You can use the `-U` option to specify the user e.g.:

```
adcli join wind2008.bmc.local -U johndoe
```

If `adcli` succeeds a keytab file will be created in `/etc/krb5.keytab`.

You can use the following command to verify the file:

```
# klist -ke | head -n 10
Keytab name: FILE:/etc/krb5.keytab
KVNO Principal
---- --------------------------------------------------------------------
---------
  6 SERVERNAME$@BCM.LOCAL (des-cbc-crc)
  6 SERVERNAME$@BCM.LOCAL (des-cbc-md5)
  6 SERVERNAME$@BCM.LOCAL (arcfour-hmac)
  6 SERVERNAME$@BCM.LOCAL (aes128-cts-hmac-sha1-96)
  7SERVERNAME$@BCM.LOCAL (des-cbc-crc)
  7SERVERNAME$@BCM.LOCAL (des-cbc-md5)
  6 HOST/SERVERNAME@BCM.LOCAL (des-cbc-crc)
#
```

You do not need to install `adcli` on every node on the cluster. It needs to be installed on only one host.

A keytab can be generated in various other ways as well:

- Using SAMBA. This method is discussed in the following KB article:

http://goo.gl/YyUI4T - Sections 1 to 4

- On a Windows host:

```
ktpass -princ host/SERVERNAME@BCM.lOCAL -mapuser johndoe@BCM.LOCAL -pa
ss MY_SECRET_PASSWORD  -out unix.keytab
```

You will need to transfer `unix.keytab` to the Linux host as `/etc/krb5.keytab`.

Kerberos keytabs are machine-independent so it does not matter where and how the file was created.

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

Configure Kerberos:

```
# cat /etc/krb5.conf
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
default_realm = BCM.LOCAL
dns_lookup_realm = true
dns_lookup_kdc = true
ticket_lifetime = 24h
renew_lifetime = 7d
forwardable = true

[realms]
BCM.LOCAL = {
 kdc = win2008.brightcomputing.com
 admin_server = win2008.brightcomputing.com
}


[domain_realm]
.bcm.local = BCM.LOCAL
bcm.local = BCM.LOCAL
#
```

Configure SSSD:

```
# cat /etc/sssd/sssd.conf
[sssd]
config_file_version = 2
services = nss, pam, ssh, autofs
domains = BCM.LOCAL

[nss]
```

```
[pam]


[domain/bcm.local]
id_provider = ad
# Uncomment if service discovery is not working
ad_server = win2008.brightcomputing.com
ad_domain = bcm.local
#
```

or if service discovery is working:

```
# cat /etc/sssd/sssd.conf
[sssd]
config_file_version = 2
services = nss, pam, ssh, autofs
domains = BCM.LOCAL

[nss]

[pam]


[domain/bcm.local]
id_provider = ad
```

Configure the system to use SSSD for authentication:

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

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

Verify that everything is working

```
# kinit -k SERVERNAME$
#
```

The command above should return to the prompt without errors:

`getent` should be able to retrieve the user's information:

```
# getent passwd administrator@bcm.local
administrator:*:331400500:331400513:Administrator:/:
#
```

SSH should be working:

```
#ssh johndoe@localhost
```

or

```
# ssh johndoe@bcm.local@localhost
```

Querying AD LDAP directly:

```
ldapsearch -H ldap://win2008.bcm.local -Y GSSAPI  -b 'cn=John Doe,cn=U
```

```
sers,dc=bcm,dc=local'
```

should return the user information without errors.

```
# getent -s sss passwd johndoe
johndoe:*:10002:10000:John Doe:/home/johndoe:/bin/sh
#
```

Configure the compute nodes

You can perform the same changes to one of the nodes and then grab the software image or copy the following files:

- /etc/nsswitch.conf

- /etc/pam.d/{system,password}-auth

- /etc/krb5.{keytab, conf}

- /etc/sssd/sssd.conf

to the same relative locations inside the software image e.g. `/cm/images/default-image/etc/nsswitch.conf`

Modify the exclude lists

You will also need to modify the exclude lists for the node's category, in order to prevent update/synchronization operations from altering SSSD 's cache:

```
# cmsh;
% category use default
% set excludelistsyncinstall
(add the following line)
no-new-files: - /var/lib/sss

% set excludelistgrab
(add the following line)
- /var/lib/sss

% set excludelistgrabnew
(add the following line)
- /var/lib/sss

% set excludelistupdate
(add the following line)
no-new-files: - /var/lib/sss

% commit
```

SELinux

In some cases it might be necessary to restore the SELinux security context of the keytab file:

```
chown root:root 0600
system_u:object_r:krb5_keytab_t:s0
```

Time synchronization

The clocks of the Domain controller and the Linux clients need to be synchronized in order for AD to function properly.

It is therefore recommended to use the domain controller as an NTP server:

```
# cmsh
[SERVERNAME]% partition use base
[SERVERNAME->partition[base]]% set timeservers win2008.bcm.local
[SERVERNAME->partition*[base*]]% commit
```

POSIX attributes

If you want to use the POSIX attributes defined in AD instead of SSSD's automatic ID mapping you will need to add the following directives in the domain section of `sssd.conf`:

```
[domain/bcm.local]
...
ldap_id_mapping = False
ldap_user_uid_number = uidNumber
ldap_user_gid_number = gidNumber
ldap_user_home_directory = unixHomeDirectory
```

Notes for RHEL7 and clones

RHEL 7 uses includes realmd and the whole process of joining and generating the configuration files can be achieved by issuing a single coomd:

```
realm join bcm.local
```

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

Notes for RHEL5 and clones

The AD provider is not avaiiable in the version of SSSD shipped with RHEL5 (1.5.1).

You can either use the LDAP provider and configure it to make GSSAPI binds to AD LDAP:

```
[sssd]
domains = BCM.LOCAL
services = nss, pam
config_file_version = 2

[nss]

[pam]

[domain/BCM.LOCAL]
# changing or commenting this value will not allow sssd service to sta
rt
id_provider = ldap
ldap_sasl_mech = GSSAPI
##### See klist -ke for the value of ldap_sasl_authid
ldap_sasl_authid =  HOST/SERVERNAME@BCM.LOCAL
ldap_krb5_keytab = /etc/krb5.keytab
ldap_krb5_init_creds = false
# to find the AD server
ldap_uri = ldap://win2008.bcm.local
ldap_schema = ad

# to get user information (UID/GID) from the active directory
ldap_user_object_class = user
ldap_user_home_directory = unixHomeDirectory
ldap_group_object_class = group

# kerberos config
auth_provider = krb5
krb5_server = win2008.bcm.local
krb5_realm = BCM.LOCAL
```

Additional LDAP-related directives can be added as needed.

or install SSSD 1.9.x on RHEL5 from a third-party repository:

1) Add the RHEL5 SSSD 1.9.x repository:

```
# cd /etc/yum.repos.d/
# wget https://repos.fedorapeople.org/repos/plabrop/SSSD-rhel5/epel-SSSD-rhel5.repo
```

2) Install SSSD:

```
# yum install sssd sssd-tools sssd-cient
```

`adcli` is not available on RHEL5 so you will have to generate the keytab on an RHEL6 system or use one of the other, previously-mentioned methods to generate it.

Summary

If service discovery is functioning the only required steps are:

1) Obtain a keytab:

```
adcli join bcm.local
```

2) Configure SSSD:

Edit `/etc/sssd/sssd.conf`

```
[sssd]
….
domains = BCM.LOCAL
...
[domain/bcm.local]
id_provider = ad
```

3) Configure Kerberos

(Recommended but not strictly necessary)

```
...
[libdefaults]
default_realm = BCM.LOCAL
 ...
[realms]
BCM.LOCAL = {
 kdc = win2008.brightcomputing.com
 admin_server = win2008.brightcomputing.com
}
...
[domain_realm]
.bcm.local = BCM.LOCAL
bcm.local = BCM.LOCAL
```

4) Use SSSD for authentication

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

Other issues

Enterprise Principals

```
Aug 31 01:52:32 SERVERNAME sshd[37629]: pam_sss(sshd:auth): system inf
o: [Cannot resolve servers for KDC in realm "BCM.COM"]
Sun Aug 31 01:52:32 2014) [[sssd[krb5_child[37715]]]] [unpack_buffer]
(0x0100): cmd [241] uid [745256] gid [10000] validate [true] offline [
false] UPN [9000029529@BCM.COM]
```

SSSD obtains the UPN value from the userPrincipalName defined in AD LDAP. If enterprise principals are being used the domain part of the the UPN might differ from what is defined in the configuration (e.g. BCM.COM instead of BCM.LOCAL)

Unfortunately, the version of SSSD that comes with RHEL6 does not support enterprise principals (See https://fedorahosted.org/sssd/ticket/1842). The workaround is to force SSSD to use the default realm, as it is defined in the configuration files, instead of trying to get that value from Active Directory.

To achieve that you need to add a non-existing value for the UPN in the domain section of sssd.conf:

```
[domain/bcm.local]
...
ldap_user_principal = nosuchattribute
```

User Management: How Do I Integrate Bright With Active Directory using the native AD provider of SSSD?

Overriding settings defined in AD

You can override settings such the user home directory, shell etc. that are defined in AD, by adding appropriate directives in `sssd.conf` For example, to override the pre-defined home directory for a user, add the `override_homedir` directive in the domain section of `sssd.conf`:

```
override_homerdir =  /home/%d/%u
```

The home directory of user `foobar` will be `/home/bcm.local/foobar`.

The available sequences are:

```
            %u
                login name

            %U
                UID number

            %d
                domain name

            %f
                fully qualified user name (user@domain)

            %o
                The original home directory retrieved from the identity
 provider.

            %%
                a literal ´%´
```

Those sequences are expanded automatically. You can use the `override_home` directive together with the above sequences to achieve the desired result. Other useful directives are:

`override_shell`, `allowed_shells`, `create_homedir` (bool), etc.

Setting the User Portal Authentication to work with AD too

If user portal authentication is to work with AD, then `/etc/pam.d/php` must be set to authenticate against AD via SSSD too. This can be done by copying the contents of `/etc/pam.d/system-auth` to `/etc/pam.d/php`.

Unique solution ID: #1224
Author: Panos Labropoulos
Last update: 2015-06-01 11:15