NAME

sssd-krb5 − SSSD Kerberos provider

DESCRIPTION

This manual page describes the configuration of the Kerberos 5 authentication backend for **sssd**(8). For a detailed syntax reference, please refer to the “FILE FORMAT” section of the **sssd.conf**(5) manual page.

The Kerberos 5 authentication backend contains auth and chpass providers. It must be paired with an identity provider in order to function properly (for example, id_provider = ldap). Some information required by the Kerberos 5 authentication backend must be provided by the identity provider, such as the user´s Kerberos Principal Name (UPN). The configuration of the identity provider should have an entry to specify the UPN. Please refer to the man page for the applicable identity provider for details on how to configure this.

This backend also provides access control based on the .k5login file in the home directory of the user. See **.k5login**(5) for more details. Please note that an empty .k5login file will deny all access to this user. To activate this feature, use ´access_provider = krb5´ in your SSSD configuration.

In the case where the UPN is not available in the identity backend, **sssd** will construct a UPN using the format *username@krb5_realm*.

CONFIGURATION OPTIONS

If the auth−module krb5 is used in an SSSD domain, the following options must be used. See the **sssd.conf**(5) manual page, section “DOMAIN SECTIONS”, for details on the configuration of an SSSD domain.

krb5_server, krb5_backup_server (string)

Specifies the comma−separated list of IP addresses or hostnames of the Kerberos servers to which SSSD should connect, in the order of preference. For more information on failover and server redundancy, see the “FAILOVER” section. An optional port number (preceded by a colon) may be appended to the addresses or hostnames. If empty, service discovery is enabled; for more information, refer to the “SERVICE DISCOVERY” section.

When using service discovery for KDC or kpasswd servers, SSSD first searches for DNS entries that specify _udp as the protocol and falls back to _tcp if none are found.

This option was named “krb5_kdcip” in earlier releases of SSSD. While the legacy name is recognized for the time being, users are advised to migrate their config files to use “krb5_server” instead.

krb5_realm (string)

The name of the Kerberos realm. This option is required and must be specified.

krb5_kpasswd, krb5_backup_kpasswd (string)

If the change password service is not running on the KDC, alternative servers can be defined here. An optional port number (preceded by a colon) may be appended to the addresses or hostnames.

For more information on failover and server redundancy, see the “FAILOVER” section. NOTE: Even if there are no more kpasswd servers to try, the backend is not switched to operate offline if authentication against the KDC is still possible.

Default: Use the KDC

krb5_ccachedir (string)

Directory to store credential caches. All the substitution sequences of krb5_ccname_template can be used here, too, except %d and %P. The directory is created as private and owned by the user, with permissions set to 0700.

Default: /tmp

krb5_ccname_template (string)

Location of the user´s credential cache. Three credential cache types are currently supported: “FILE”, “DIR” and “KEYRING:persistent”. The cache can be specified either as *TYPE:RESIDUAL*, or as an absolute path, which implies the “FILE” type. In the template, the following sequences are substituted:

%u
: login name

%U
: login UID

%p
: principal name

%r
: realm name

%h
: home directory

%d
: value of krb5_ccachedir

%P
: the process ID of the SSSD client

%%
: a literal ´%´

If the template ends with ´XXXXXX´ mkstemp(3) is used to create a unique filename in a safe way.

When using KEYRING types, the only supported mechanism is “KEYRING:persistent:%U”, which uses the Linux kernel keyring to store credentials on a per−UID basis. This is also the recommended choice, as it is the most secure and predictable method.

The default value for the credential cache name is sourced from the profile stored in the system wide krb5.conf configuration file in the [libdefaults] section. The option name is default_ccache_name. See krb5.conf(5)´s PARAMETER EXPANSION paragraph for additional information on the expansion format defined by krb5.conf.

NOTE: Please be aware that libkrb5 ccache expansion template from **krb5.conf**(5) uses different expansion sequences than SSSD.

Default: (from libkrb5)

krb5_auth_timeout (integer)
: Timeout in seconds after an online authentication request or change password request is aborted. If possible, the authentication request is continued offline.

  Default: 6

krb5_validate (boolean)
: Verify with the help of krb5_keytab that the TGT obtained has not been spoofed. The keytab is checked for entries sequentially, and the first entry with a matching realm is used for validation. If no entry matches the realm, the last entry in the keytab is used. This process can be used to validate environments using cross−realm trust by placing the appropriate keytab entry as the last entry or the only entry in the keytab file.

  Default: false

krb5_keytab (string)
: The location of the keytab to use when validating credentials obtained from KDCs.

Default: /etc/krb5.keytab

krb5_store_password_if_offline (boolean)
Store the password of the user if the provider is offline and use it to request a TGT when the provider comes online again.

NOTE: this feature is only available on Linux. Passwords stored in this way are kept in plaintext in the kernel keyring and are potentially accessible by the root user (with difficulty).

Default: false

krb5_renewable_lifetime (string)
Request a renewable ticket with a total lifetime, given as an integer immediately followed by a time unit:

s for seconds

m for minutes

h for hours

d for days.

If there is no unit given, *s* is assumed.

NOTE: It is not possible to mix units. To set the renewable lifetime to one and a half hours, use ´90m´ instead of ´1h30m´.

Default: not set, i.e. the TGT is not renewable

krb5_lifetime (string)
Request ticket with a lifetime, given as an integer immediately followed by a time unit:

s for seconds

m for minutes

h for hours

d for days.

If there is no unit given *s* is assumed.

NOTE: It is not possible to mix units. To set the lifetime to one and a half hours please use ´90m´ instead of ´1h30m´.

Default: not set, i.e. the default ticket lifetime configured on the KDC.

krb5_renew_interval (string)

The time in seconds between two checks if the TGT should be renewed. TGTs are renewed if about half of their lifetime is exceeded, given as an integer immediately followed by a time unit:

s for seconds

m for minutes

h for hours

d for days.

If there is no unit given, *s* is assumed.

NOTE: It is not possible to mix units. To set the renewable lifetime to one and a half hours, use ´90m´ instead of ´1h30m´.

If this option is not set or is 0 the automatic renewal is disabled.

Default: not set

krb5_use_fast (string)
Enables flexible authentication secure tunneling (FAST) for Kerberos pre−authentication. The following options are supported:

never use FAST. This is equivalent to not setting this option at all.

try to use FAST. If the server does not support FAST, continue the authentication without it.

demand to use FAST. The authentication fails if the server does not require fast.

Default: not set, i.e. FAST is not used.

NOTE: a keytab is required to use FAST.

NOTE: SSSD supports FAST only with MIT Kerberos version 1.8 and later. If SSSD is used with an older version of MIT Kerberos, using this option is a configuration error.

krb5_fast_principal (string)
Specifies the server principal to use for FAST.

krb5_canonicalize (boolean)
Specifies if the host and user principal should be canonicalized. This feature is available with MIT Kerberos 1.7 and later versions.

Default: false

krb5_use_kdcinfo (boolean)
Specifies if the SSSD should instruct the Kerberos libraries what realm and which KDCs to use. This option is on by default, if you disable it, you need to configure the Kerberos library using the **krb5.conf**(5) configuration file.

See the **sssd_krb5_locator_plugin**(8) manual page for more information on the locator plugin.

Default: true

krb5_use_enterprise_principal (boolean)
Specifies if the user principal should be treated as enterprise principal. See section 5 of RFC 6806 for more details about enterprise principals.

Default: false (AD provider: true)

FAILOVER

The failover feature allows back ends to automatically switch to a different server if the current server fails.

Failover Syntax

The list of servers is given as a comma−separated list; any number of spaces is allowed around the comma. The servers are listed in order of preference. The list can contain any number of servers.

For each failover−enabled config option, two variants exist: *primary* and *backup*. The idea is that servers in the primary list are preferred and backup servers are only searched if no primary servers can be reached. If a backup server is selected, a timeout of 31 seconds is set. After this timeout SSSD will periodically try to reconnect to one of the primary servers. If it succeeds, it will replace the current active (backup) server.

The Failover Mechanism

The failover mechanism distinguishes between a machine and a service. The back end first tries to resolve the hostname of a given machine; if this resolution attempt fails, the machine is considered offline. No further attempts are made to connect to this machine for any other service. If the resolution attempt succeeds, the back end tries to connect to a service on this machine. If the service connection attempt fails, then only this particular service is considered offline and the back end automatically switches over to the next service. The machine is still considered online and might still be tried for another service.

Further connection attempts are made to machines or services marked as offline after a specified period of time; this is currently hard coded to 30 seconds.

If there are no more machines to try, the back end as a whole switches to offline mode, and then attempts to reconnect every 30 seconds.

SERVICE DISCOVERY

The service discovery feature allows back ends to automatically find the appropriate servers to connect to using a special DNS query. This feature is not supported for backup servers.

Configuration

If no servers are specified, the back end automatically uses service discovery to try to find a server. Optionally, the user may choose to use both fixed server addresses and service discovery by inserting a special keyword, "_srv_", in the list of servers. The order of preference is maintained. This feature is useful if, for example, the user prefers to use service discovery whenever possible, and fall back to a specific server when no servers can be discovered using DNS.

The domain name

Please refer to the "dns_discovery_domain" parameter in the **sssd.conf**(5) manual page for more details.

The protocol

The queries usually specify _tcp as the protocol. Exceptions are documented in respective option description.

See Also

For more information on the service discovery mechanism, refer to RFC 2782.

EXAMPLE

The following example assumes that SSSD is correctly configured and FOO is one of the domains in the *[sssd]* section. This example shows only configuration of Kerberos authentication; it does not include any identity provider.

```
[domain/FOO]
auth_provider = krb5
krb5_server = 192.168.1.1
krb5_realm = EXAMPLE.COM
```

SEE ALSO

sssd(8), **sssd.conf**(5), **sssd-ldap**(5), **sssd-krb5**(5), **sssd-simple**(5), **sssd-ipa**(5), **sssd-ad**(5), **sssd-sudo**(5), **sss_cache**(8), **sss_debuglevel**(8), **sss_groupadd**(8), **sss_groupdel**(8), **sss_groupshow**(8), **sss_groupmod**(8), **sss_useradd**(8), **sss_userdel**(8), **sss_usermod**(8), **sss_obfuscate**(8), **sss_seed**(8), **sssd_krb5_locator_plugin**(8), **sss_ssh_authorizedkeys**(8), **sss_ssh_knownhostsproxy**(8), **sssd-ifp**(5), **pam_sss**(8). **sss_rpcidmapd**(5)

AUTHORS

The SSSD upstream − http://fedorahosted.org/sssd