NAME

pam_sss_gss − PAM module for SSSD GSSAPI authentication

SYNOPSIS

pam_sss_gss.so [*debug*]

DESCRIPTION

pam_sss_gss.so authenticates user over GSSAPI in cooperation with SSSD.

This module will try to authenticate the user using the GSSAPI hostbased service name host@hostname which translates to host/hostname@REALM Kerberos principal. The *REALM* part of the Kerberos principal name is derived by Kerberos internal mechanisms and it can be set explicitly in configuration of [domain_realm] section in /etc/krb5.conf.

SSSD is used to provide desired service name and to validate the user's credentials using GSSAPI calls. If the service ticket is already present in the Kerberos credentials cache or if user's ticket granting ticket can be used to get the correct service ticket then the user will be authenticated.

If **pam_gssapi_check_upn** is True (default) then SSSD requires that the credentials used to obtain the service tickets can be associated with the user. This means that the principal that owns the Kerberos credentials must match with the user principal name as defined in LDAP.

To enable GSSAPI authentication in SSSD, set **pam_gssapi_services** option in [pam] or domain section of sssd.conf. The service credentials need to be stored in SSSD's keytab (it is already present if you use ipa or ad provider). The keytab location can be set with **krb5_keytab** option. See **sssd.conf**(5) and **sssd-krb5**(5) for more details on these options.

Some Kerberos deployments allow to associate authentication indicators with a particular pre−authentication method used to obtain the ticket granting ticket by the user. **pam_sss_gss.so** allows to enforce presence of authentication indicators in the service tickets before a particular PAM service can be accessed.

If **pam_gssapi_indicators_map** is set in the [pam] or domain section of sssd.conf, then SSSD will perform a check of the presence of any configured indicators in the service ticket.

OPTIONS

debug
Print debugging information.

MODULE TYPES PROVIDED

Only the **auth** module type is provided.

RETURN VALUES

PAM_SUCCESS
The PAM operation finished successfully.

PAM_USER_UNKNOWN
The user is not known to the authentication service or the GSSAPI authentication is not supported.

PAM_AUTH_ERR
Authentication failure.

PAM_AUTHINFO_UNAVAIL
Unable to access the authentication information. This might be due to a network or hardware failure.

PAM_SYSTEM_ERR
A system error occurred. The SSSD log files may contain additional information about the error.

EXAMPLES

The main use case is to provide password−less authentication in sudo but without the need to disable authentication completely. To achieve this, first enable GSSAPI authentication for sudo in sssd.conf:

```
[domain/MYDOMAIN]
pam_gssapi_services = sudo, sudo−i
```

And then enable the module in desired PAM stack (e.g. /etc/pam.d/sudo and /etc/pam.d/sudo−i).

...
auth sufficient pam_sss_gss.so
...

TROUBLESHOOTING

SSSD logs, pam_sss_gss debug output and syslog may contain helpful information about the error. Here are some common issues:

1. I have KRB5CCNAME environment variable set and the authentication does not work: Depending on your sudo version, it is possible that sudo does not pass this variable to the PAM environment. Try adding KRB5CCNAME to **env_keep** in /etc/sudoers or in your LDAP sudo rules default options.

2. Authentication does not work and syslog contains "Server not found in Kerberos database": Kerberos is probably not able to resolve correct realm for the service ticket based on the hostname. Try adding the hostname directly to **[domain_realm]** in /etc/krb5.conf like so:

3. Authentication does not work and syslog contains "No Kerberos credentials available": You don't have any credentials that can be used to obtain the required service ticket. Use kinit or authenticate over SSSD to acquire those credentials.

4. Authentication does not work and SSSD sssd−pam log contains "User with UPN [$UPN] was not found." or "UPN [$UPN] does not match target user [$username].": You are using credentials that can not be mapped to the user that is being authenticated. Try to use kswitch to select different principal, make sure you authenticated with SSSD or consider disabling **pam_gssapi_check_upn**.

[domain_realm]
.myhostname = MYREALM

SEE ALSO

sssd(8), **sssd.conf**(5), **sssd-ldap**(5), **sssd-krb5**(5), **sssd-simple**(5), **sssd-ipa**(5), **sssd-ad**(5), **sssd-files**(5), **sssd-sudo**(5), **sssd-session-recording**(5), **sss_cache**(8), **sss_debuglevel**(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) **sssd-systemtap**(5)

AUTHORS

The SSSD upstream − https://github.com/SSSD/sssd/