redhat.

SSSD 1.9

Overview Session

SSSD Team

10-17-2012

RED HAT® ENTERPRISE LINUX®

Agenda

- Purpose and use cases
- Architecture and capabilities
- Future direction
- Resources

SSSD Purpose

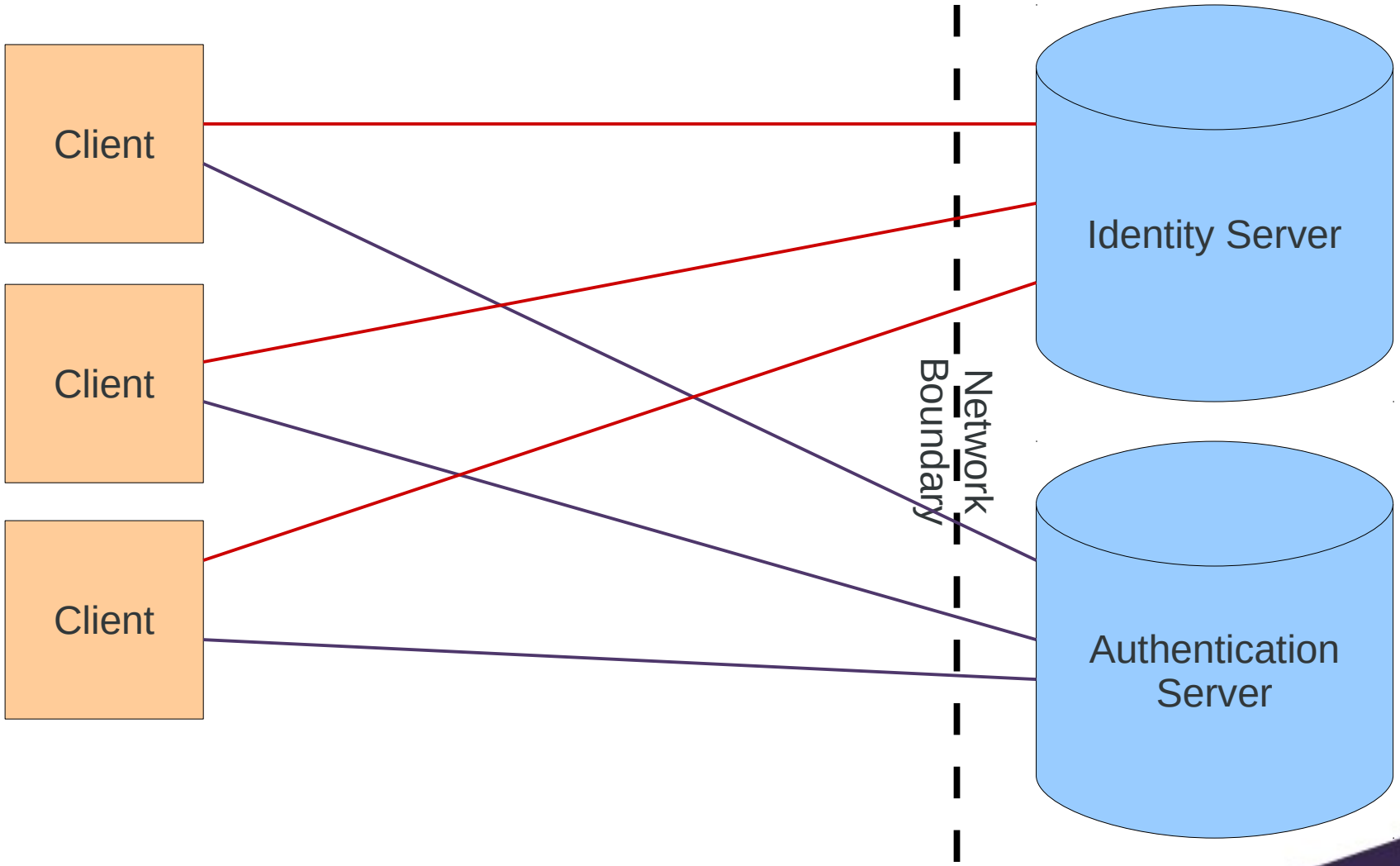
SSSD stands for: System Security Services Daemon

- Manages communication with centralized identity and authentication stores
- Provides robust, predictable caching for network accounts
- Can cache authentication credentials locally to allow local updates
- Can handle multiple domains of user data and authentication

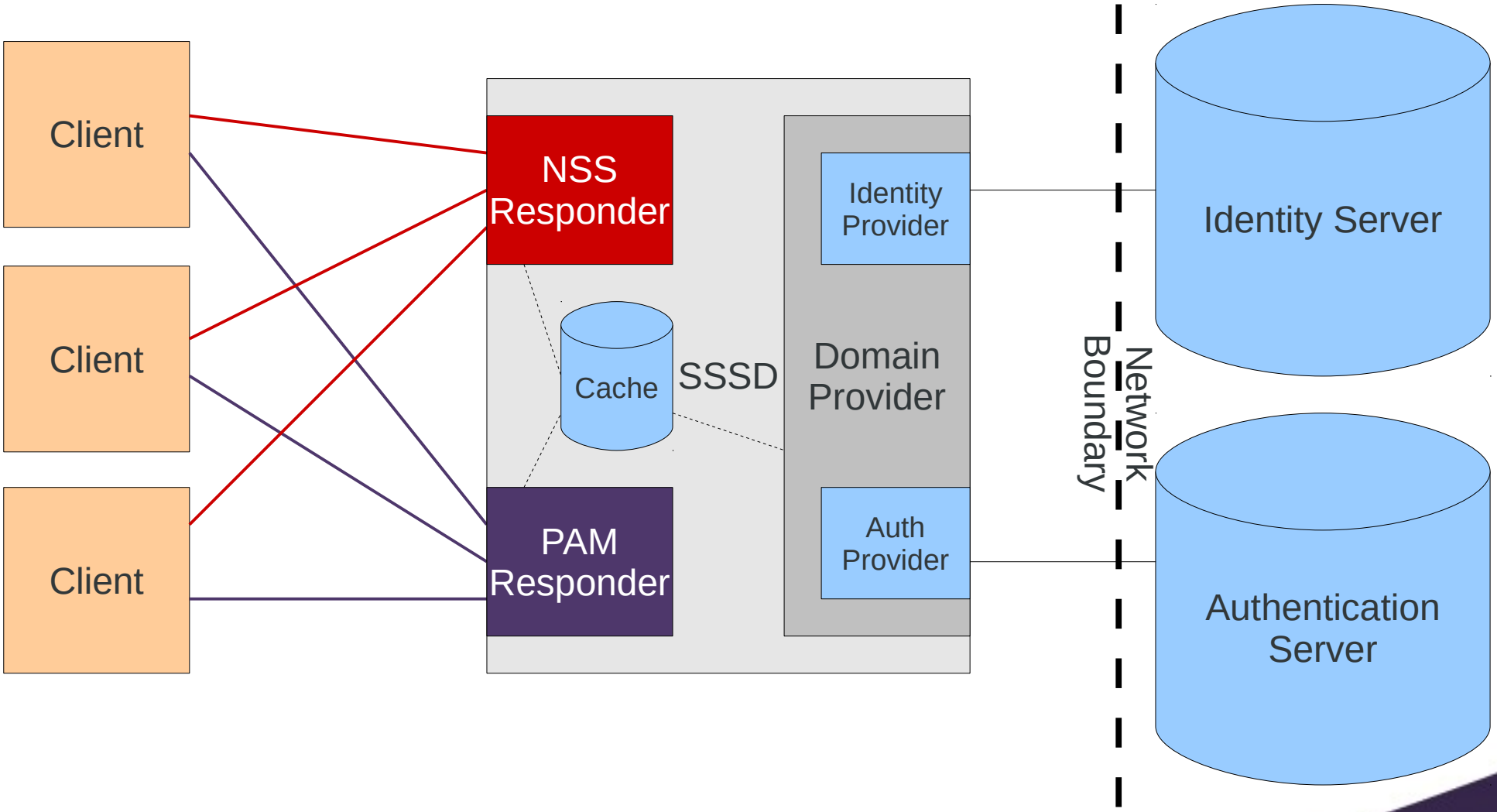
Use Cases

- Datacenter
  - Datacenters that require highly-available authentication can take advantage of SSSDs caching to ride out temporary internal service outages (such as an LDAP or Kerberos server outage)
- Corporate Laptop
  - Traditional problem: users maintain a separate local account on the laptop to log into when out of the office
  - With SSSD providing cached credentials, the user can keep the same account (UID and all) when logging in remotely

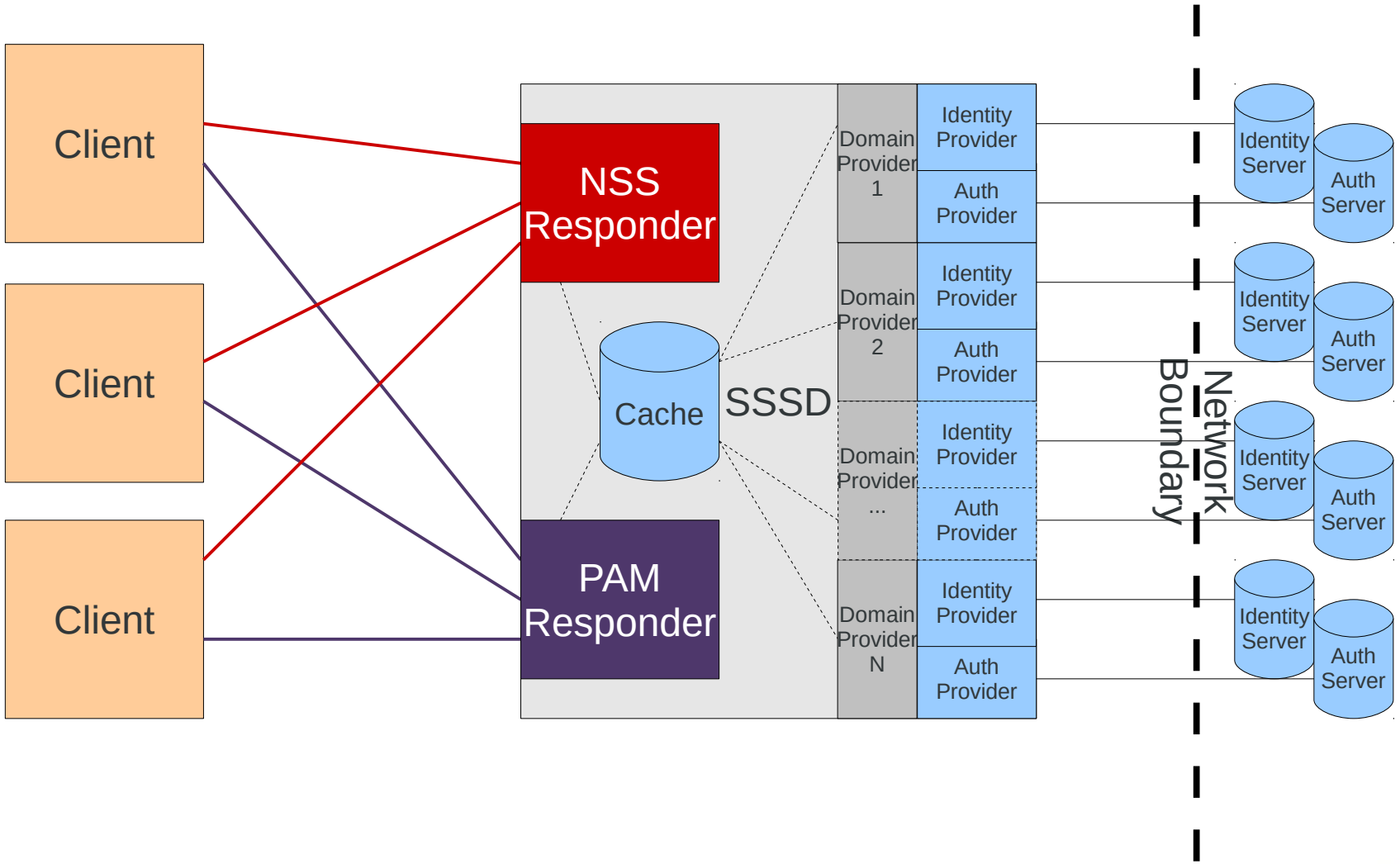
Identity Source Integration without SSSD



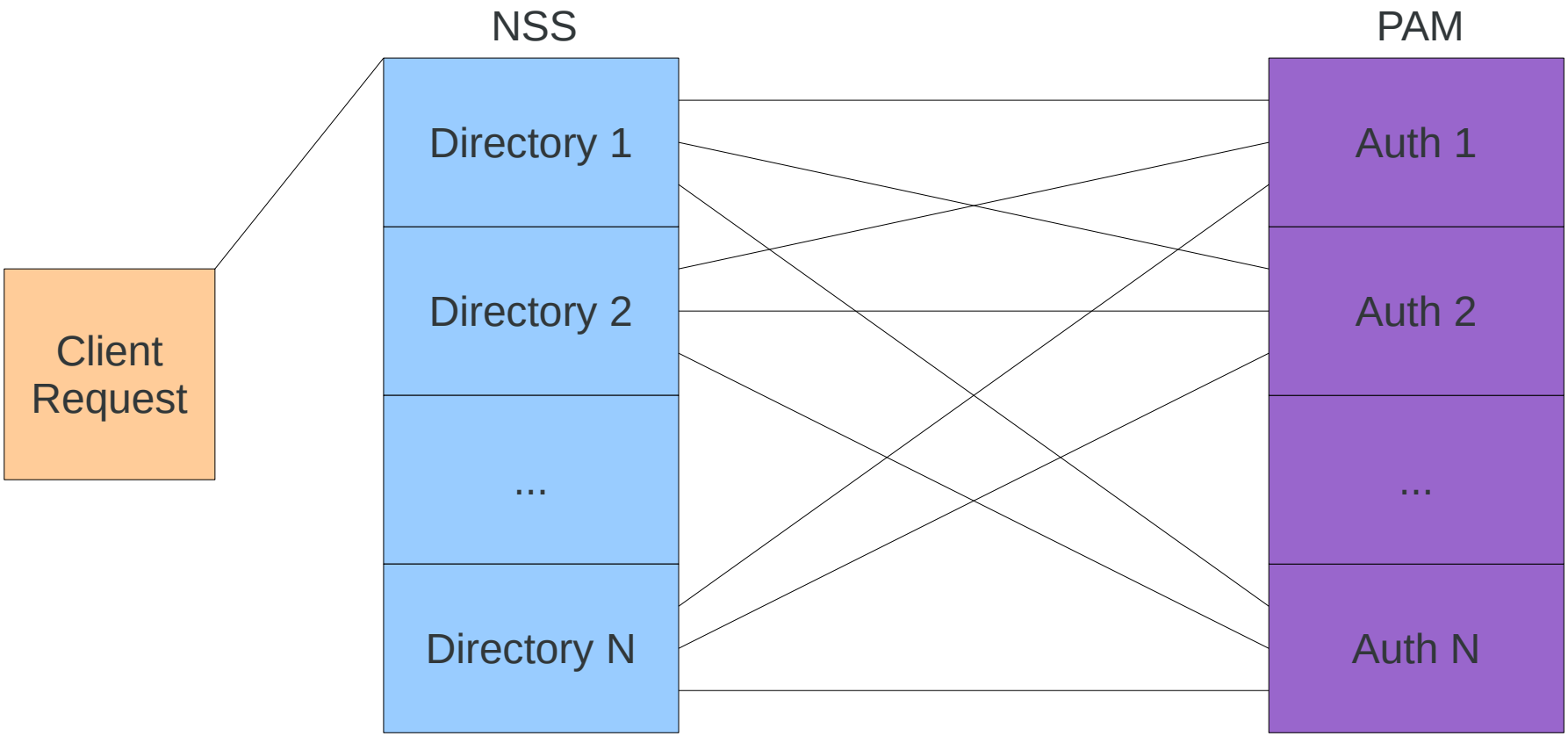
Identity Source Integration with SSSD



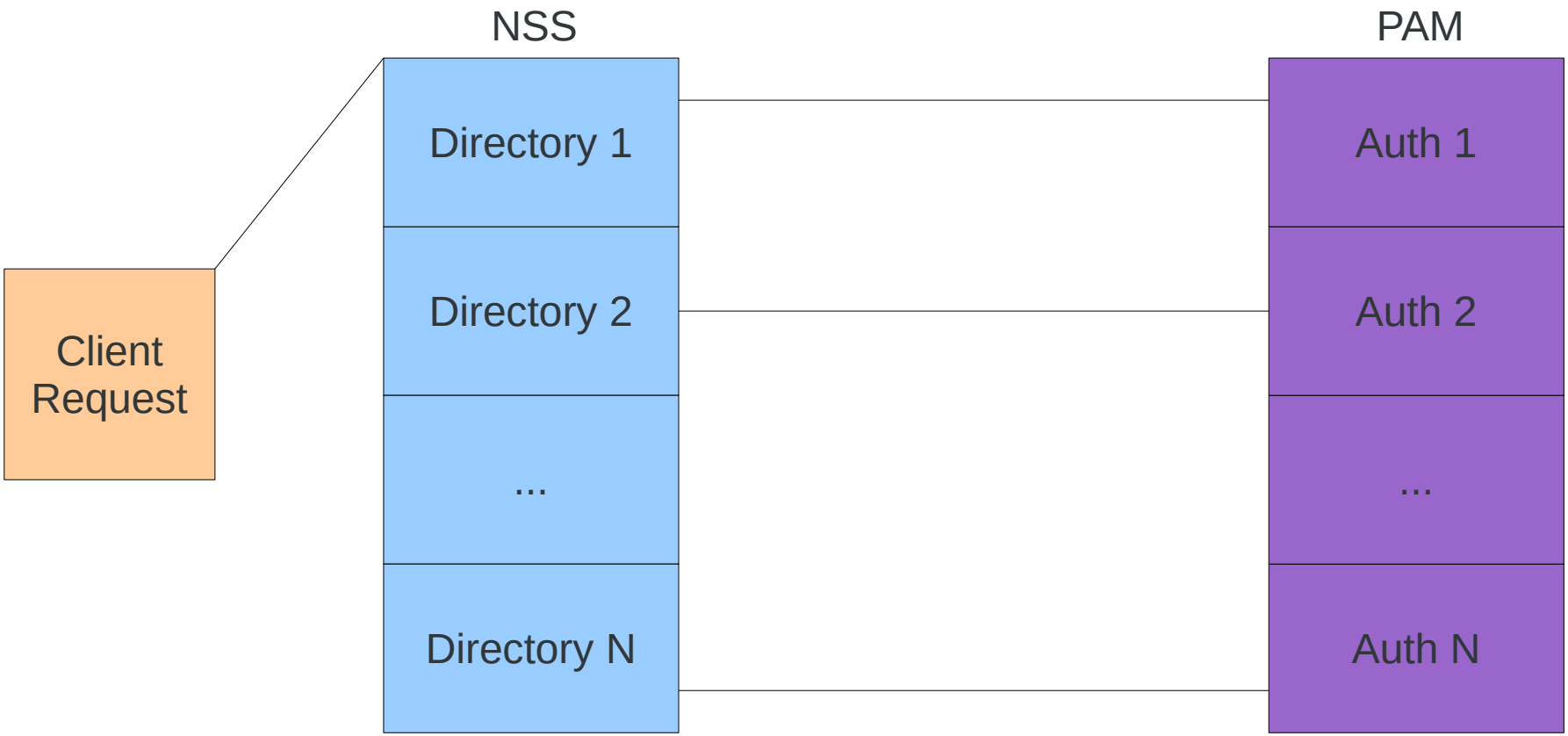
SSSD with Multiple Identity Sources



Traditional Authentication



Authentication with SSSD



Supported Servers

- Active Directory or FreeIPA (IdM)
  - LDAP identity lookups and authentication
  - Kerberos authentication
- LDAP Servers: 389 DS, OpenLDAP
  - LDAP identity lookups and authentication
- MIT Kerberos KDC for authentication, usually with LDAP for identity

Advanced Caching Capabilities

- Over nscd
  - SSSD user and group cache expiration is more predictable
  - When cached in the SSSD, user identity entries will not expire while offline
  - SSSD operates closer to the backends, so it can be aware of backend-specific temporary failures that nscd would report as missing entries
- Over pam_ccreds
  - SSSD can be configured to perform offline expiration of cached credentials (requiring clients to 'check in' with the central server regularly)
  - SSSD will inform the user when authenticating with cached credentials, and will warn of approaching offline expiration

Differences from traditional authentication

- SSSD requires the use of transport layer encryption when performing simple bind authentication against LDAP
  - LDAPS, TLS or GSSAPI
- SSSD enforces a one-to-one relationship between user identities and authentication services
- Offline authentication against a Kerberos server can be configured to automatically perform a kinit when the server becomes available
- User tickets can be automatically renewed based on policy

Supported NSS Maps

- Users (passwd)
- Groups
- Netgroups
- Services – since 1.8

Integration with 3rd party Applications

- Automount
  - Starting with version 1.8, SSSD can cache autofs maps
- SUDO
  - Starting with version 1.9, SSSD can cache SUDO rules
- OpenSSH
  - Starting with version 1.8, SSSD can cache SSH host keys. Currently implemented for IPA provider only, LDAP provider implementation is planned.

Specific FreeIPA (IdM) Integration Features

- System joins FreeIPA domain
- Smooth password migration when environment transitions from LDAP to FreeIPA
- Centrally managed by FreeIPA HBAC (Host Based Access Control) rules
- Centrally managed SELinux user mappings

Authentication Providers

- LDAP
  - Password authentication through LDAP simple bind
- KRB5
  - Password authentication through the Kerberos protocol
  - Authentication through this backend will perform a kinit and acquire a Kerberos ticket-granting ticket for network single-sign-on
- IPA
  - Password authentication to FreeIPA through the Kerberos protocol or LDAP simple bind (during password migration only)
  - Handles all advanced IPA integration features

Authentication Providers (continued)

- AD
  - Password authentication to AD through the Kerberos protocol
  - Handles many advanced Active Directory integration features
- Proxy
  - Invokes a custom PAM stack to perform authentication against a traditional PAM module (or series of modules)

Identity Providers

- LDAP
  - Supports LDAP servers using RFC2307 or RFC2307bis schema
- IPA
  - Support for the FreeIPA identity store
- AD
  - Support for the Active Directory identity store
- Proxy
  - Can support identity data from an existing nameservice library

Access control Providers

- Permit
  - Always allows access to any user that succeeded at authentication
  - Default if no access_provider is specified
- Deny
  - Always denies access, regardless of authentication success
- Simple
  - Grants access to users in a list

Access Control Providers (continued)

- LDAP
  - Grants access to users whose user entry matches a particular LDAP search query
  - Support access control based on expiration policy
  - Able to limit login based on the “host” or “authorizedService” attributes
- IPA
  - Grants access based on complex host-based access control (HBAC) rules configured on a FreeIPA server
  - Access control provider may be configured to respect account lock and account expiration status

Advanced Active Directory Features

- SID to UID/GID mapping
- De-reference control
- Nested groups resolution
- Retrieval of groups with large number of members using an AD-specific extension

Active Directory Integration Options

Feature	LDAP/KRB	Winbind	SSSD
Authenticate using Kerberos or LDAP	Yes	Yes	Yes
Identities are looked up in AD	Yes	Yes	Yes
Requires SFU/IMU	Yes	No	Yes until SSSD 1.9
ID mapping	None	Multiple ways	One way starting SSSD 1.9
System is joined into AD	Manual	Has join utility	Solved by realmd
Supports multiple AD domains	No	Yes	Will in SSSD 1.10
Supports heterogeneous domains	No	No	Yes
Support advanced AD features	No	Yes	Some
Reliability	High	Medium	High

Other Features

- Support of the Kerberos DIR cache to store multiple credential caches tied with different identities
- Support of ticket cache in common location
- Support of the cross realm Kerberos trusts between FreeIPA and AD

Further Direction

- Further AD integration improvements
- Support of Smart Cards
- Winbind replacement for CIFS client and server use cases
- Desktop integration to support 2FA authentication via Kerberos
- Monitoring of the ticket expiration
- D-BUS interface for authentication and identity lookups
- RADIUS authentication provider

Resources

- man sssd
  - Many detailed man pages about sssd configuration
- Project source: git://git.fedorahosted.org/git/sssd.git
- Project wiki and trac: https://fedorahosted.org/sssd
- IRC on freenode.net: #sssd
- Mailing lists:
  - Developer list: sssd-devel@lists.fedorahosted.org
  - User list: sssd-users@lists.fedorahosted.org