Identity Management: Unicorn or Gorgon?

LOPSA Columbus

Josh Preston
Solutions Architect, Red Hat, Inc.
10-27-2016

Josh Preston mrjoshuap@redhat.com

Twitter @MrJoshuaP
XBox Live MrJoshuaP

https://people.redhat.com/jpreston
https://github.com/mrjoshuap
https://www.linkedin.com/in/mrjoshuap

... and coming soon: mrjoshuap.com

Agenda

Goals of the Presentation

- Introduce the identity management problem space
- Provide an overview of the identity management components
- Review a few real world use cases
  - Centralized user and group management
  - Password policy management
  - Host based access controls (HBAC)
  - User SELinux context mappings
  - SUDOers management
  - SSL certificate management

Identity Management Problem Space

Wikipedia definition:

Identity Management – (noun)

“Identity management (IdM) describes the management of individual principals, their authentication, authorization, and privileges within or across system and enterprise boundaries with the goal of increasing security and productivity while decreasing cost, downtime and repetitive tasks.”

Identity Management Problem Space

Main aspects

- **Identities**
  - Where are my users stored? What properties do they have? How is this data made available to systems and applications?
- **Authentication**
  - What credentials do my users use to authenticate? Passwords? Smart Cards? Special devices? Is there SSO? How can the same user access file stores and web applications without requiring re-authentication?

Identity Management Problem Space

Main aspects, continued

- **Access control**
  - Which users have access to which systems, services, applications? What commands can they run on those systems? What SELinux context is a user is mapped to?
- **Policies**
  - What is the strength of the password? What are the automount rules? What are Kerberos ticket policies?

Introducing IdM (FreeIPA)

- IdM – Identity Management in Red Hat Enterprise Linux
- Based on FreeIPA open source technology
- IPA stands for Identity, Policy, Audit
    - So far we have focused on identities and related policies
    - A separate project is ongoing in the audit space

Problems IdM Solves

- Central management of authentication and identities for Linux clients better than standalone LDAP/Kerberos/NIS - based solutions
- Central management of internal SSL certificates with built-in PKI infrastructure easier than standalone OpenSSL CAs
- Acts as a gateway between the Linux infrastructure and AD environment making infrastructure more manageable and more cost effective
    - Paths to AD integration
        - Cross forest trusts (recommended)
        - User and password synchronization (not recommended)

Identity Management: Unicorn or Gorgon?

Wikipedia definition:

Unicorn – (noun)

“The unicorn is a legendary creature that has been described since antiquity as a beast with a large, pointed, spiraling horn projecting from its forehead. The unicorn was depicted in ancient seals of the Indus Valley Civilization and was mentioned by the ancient Greeks in accounts of natural history by various writers, including Ctesias, Strabo, Pliny the Younger, and Aelian.[1] The Bible also describes an animal, the re'em, which some translations have erroneously rendered with the word unicorn.”

Examples of Unicorns

Examples of Unicorns

Examples of Unicorns

Wikipedia definition:

Gorgon – (noun)

“In Greek mythology, a Gorgon is a female creature. The name derives from the ancient Greek word gorgós, which means "dreadful", and appears to come from the same root as the Sanskrit word "garğ" which is defined as a guttural sound, similar to the growling of a beast,[1] thus possibly originating as an onomatopoeia. While descriptions of Gorgons vary across Greek literature and occur in the earliest examples of Greek literature, the term commonly refers to any of three sisters who had hair made of living, venomous snakes, as well as a horrifying visage that turned those who beheld her to stone.”

Examples of Gorgons

Examples of Gorgons

Examples of Gorgons

Identity Management: The Truth

Examples of the Truth

Examples of the Truth

Examples of the Truth

Examples of the Truth

Overview of the Identity Management Components

Components of the Portfolio

Identity Management in FreeIPA

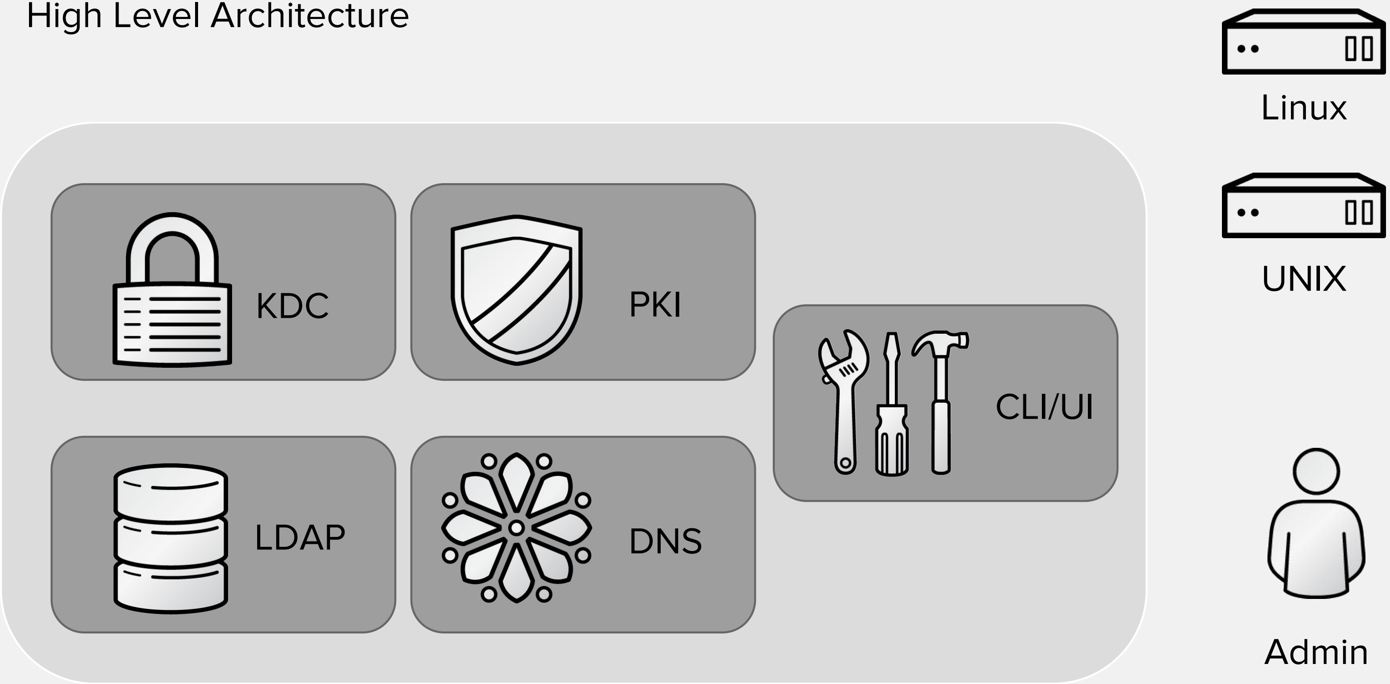
- SSSD
- Certmonger
- Keycloak IdP
- Apache modules

Identity Management (IdM)

- Domain controller for Linux/UNIX environments
- Combines LDAP, Kerberos, DNS and certificate management capabilities
- Provides centralized authentication, authorization and identity information for Linux/UNIX infrastructure
- Enables centralized policy and privilege escalation management
- Integrates with Active Directory on the server-to-server level

Identity Management

High Level Architecture



KDC

PKI

CLI/UI

LDAP

DNS

Linux

UNIX

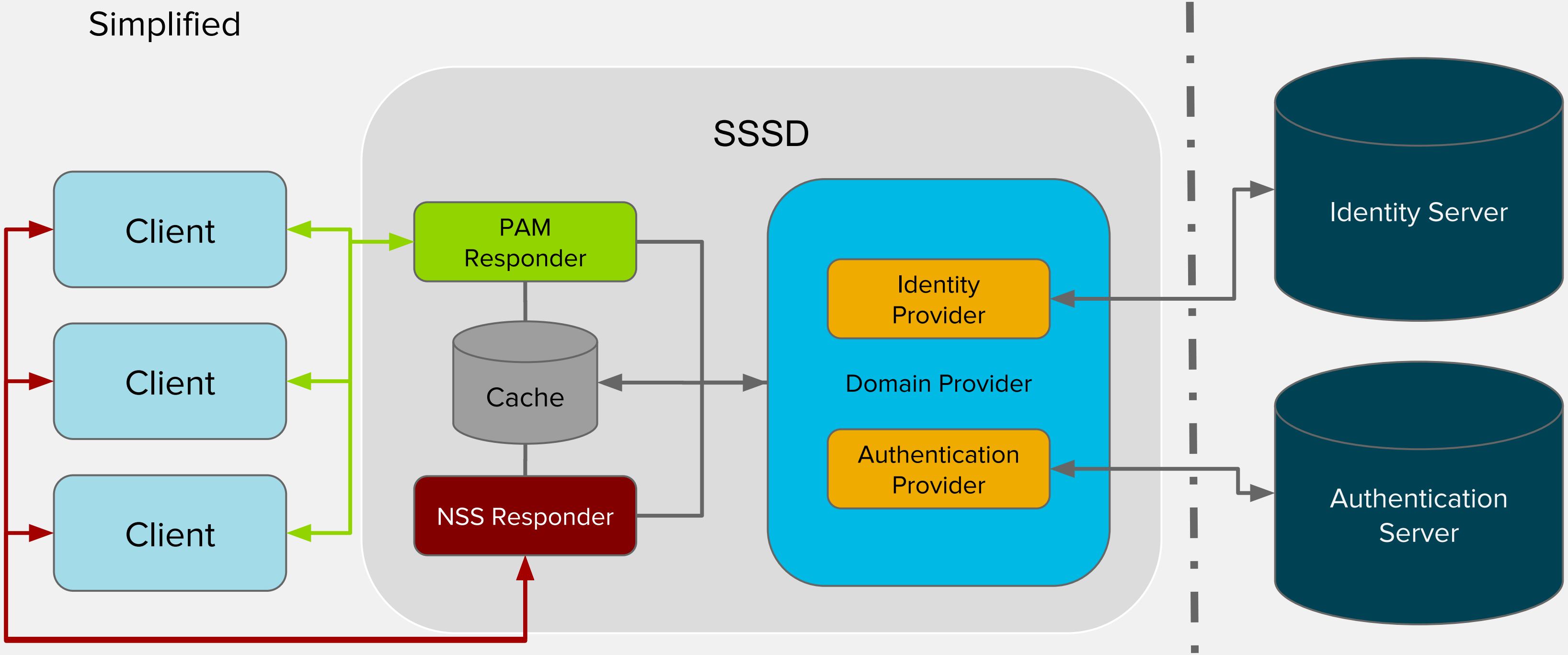
Admin

SSSD (System Security Services Daemon)

- Client-side component
- Part of Red Hat Enterprise Linux and many other Linux distributions
- Allows connecting a system to the identity and authentication source of your choice
- Caches identity and policy information for offline use
- Capable of connecting to different sources of identity data at the same time

SSSD Architecture

Simplified



Certmonger

- Client side component
- Connects to central Certificate Server and requests certificates
- Tracks and auto renews the certificates it is tracking

Keycloak IdP

- Identity Provider implementation
- Allows federation between different applications using SAML, OIDC based SSO

Apache Modules

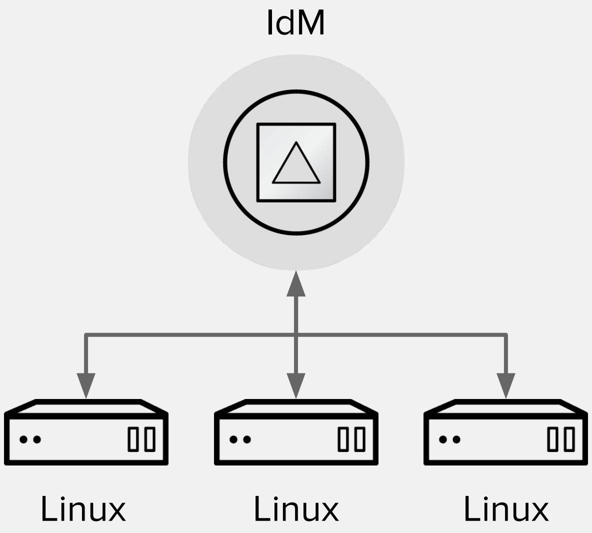
- Modules that can be integrated with Apache server
- Modules that support forms-based, Kerberos, certificate-based or SAML authentication
  - We are working on OIDC authentication
- Authorization and identity data lookups are also possible using corresponding modules

Solving Real World Identity Management Challenges

Use Cases and Challenges

- **How can I provide centralized authentication?**
- Can I provide centralized password policies?
- Can I define access control to hosts without copying configuration files?
- Can I manage SSH keys for users and hosts?
- Can I provide centralized SUDO, automount, SELinux user mappings?
- Is there a cost effective solution that provides strong authentication using OTP?
- Can I simplify SSL certificate management?

Centralized Authentication



Steps:

- Consolidate your user accounts
- Load your user data into a IdM
- Connect your Linux/UNIX systems to IdM
  - ipa-client-install

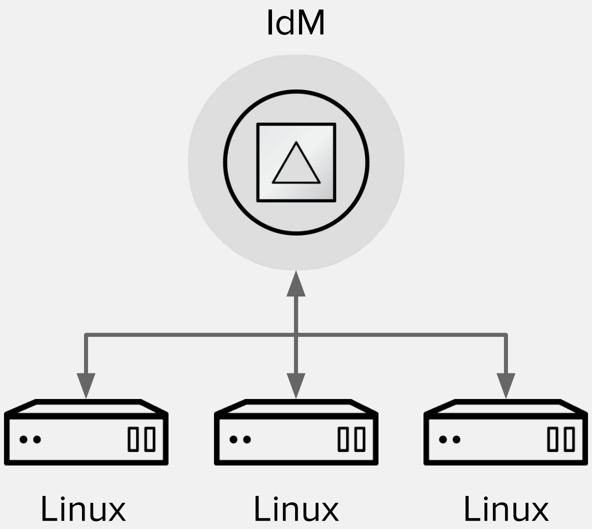
Why would I use IdM?

- Different authentication methods:
  - LDAP, Kerberos, OTP, Certificates
- Integrated solution
  - Easy to install and manage
- Integrates with AD
- Has a lot of other valuable capabilities

Use Cases and Challenges

- How can I provide centralized authentication?
- **Can I provide centralized password policies?**
- Can I define access control to hosts without copying configuration files?
- Can I manage SSH keys for users and hosts?
- Can I provide centralized SUDO, automount, SELinux user mappings?
- Is there a cost effective solution that provides strong authentication using OTP?
- Can I simplify SSL certificate management?

Password Policies



Steps:

- Create/update a global password policy
- Create logical user groups
- Organize your user accounts into groups
- Define group based password policies
- Define priorities for password policies

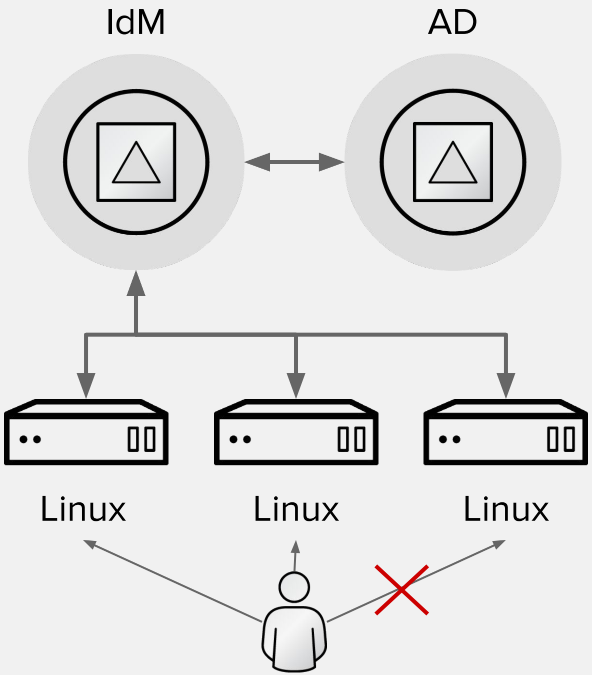
Why would I use IdM?

- Different password policy priorities
  - Global -> Group -> User
- Integrated solution
  - Easy to install, manage and audit

Use Cases and Challenges

- How can I provide centralized authentication?
- Can I provide centralized password policies?
- **Can I define access control to hosts without copying configuration files?**
- Can I manage SSH keys for users and hosts?
- Can I provide centralized SUDO, automount, SELinux user mappings?
- Is there a cost effective solution that provides strong authentication using OTP?
- Can I simplify SSL certificate management?

Host Based Access Control

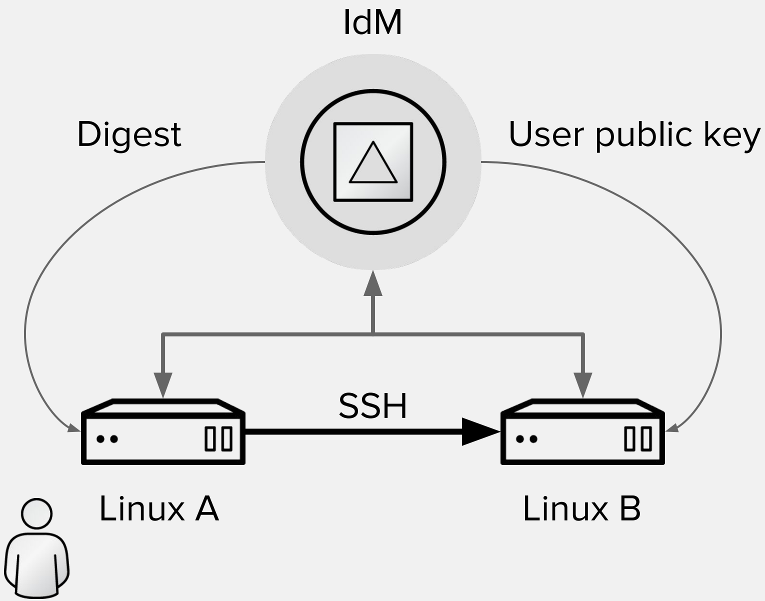


- Host based access control:
  - Which users or group of users can access
  - Which hosts or groups of hosts
  - Using which login services:
    - console, ssh, sudo, ftp, sftp, etc.
- You define rules centrally
- Works with trusted AD users

Use Cases and Challenges

- How can I provide centralized authentication?
- Can I provide centralized password policies?
- Can I define access control to hosts without copying configuration files?
- **Can I manage SSH keys for users and hosts?**
- Can I provide centralized SUDO, automount, SELinux user mappings?
- Is there a cost effective solution that provides strong authentication using OTP?
- Can I simplify SSL certificate management?

SSH Key Management

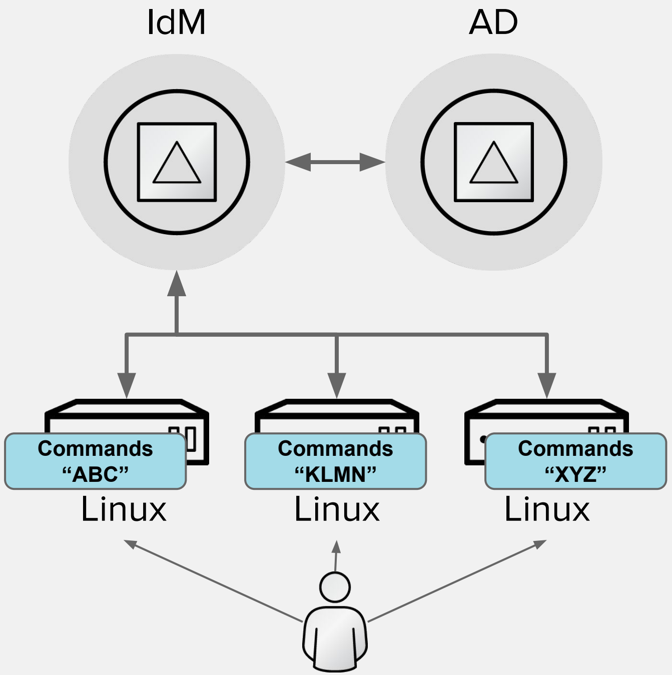


- Host public keys uploaded at the client installation time
- User can upload his public key to IdM manually
- When user SSHs from a system A the public key of to the target system B is delivered to system A (no need to validate digest)
- User public key is automatically delivered to system B
- Works with trusted AD users

Use Cases and Challenges

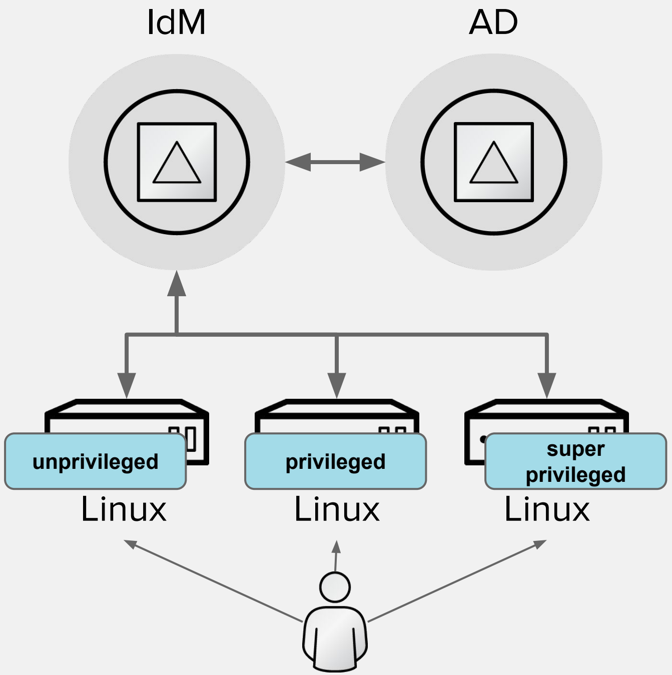
- How can I provide centralized authentication?
- Can I provide centralized password policies?
- Can I define access control to hosts without copying configuration files?
- Can I manage SSH keys for users and hosts?
- **Can I provide centralized SUDO, automount, SELinux user mappings?**
- Is there a cost effective solution that provides strong authentication using OTP?
- Can I simplify SSL certificate management?

SUDO Integration



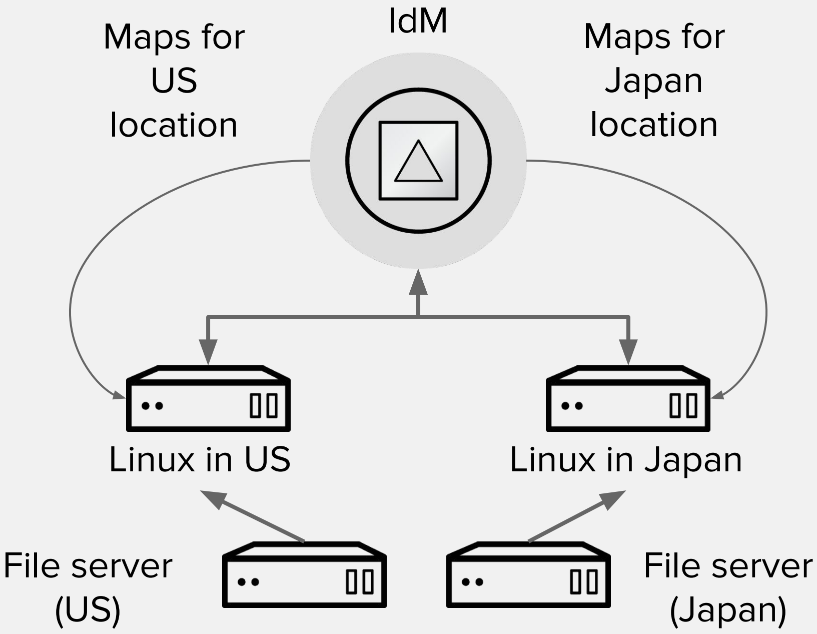
- Centrally define commands and groups of commands
- Define which groups of users can run these commands or groups of commands on which hosts or groups of hosts
- Rules are enforced on client
- Rules are cached
- Capability is integrated into the sudo utility
- Works with trusted AD users

SELinux Integration



- Mappings can be defined centrally
- Allow different users on different systems have different SELinux context
- Default SELinux labels are available in IdM configuration
- Mappings are enforced on the client
- Mappings are cached
- Works with trusted AD users

Automount

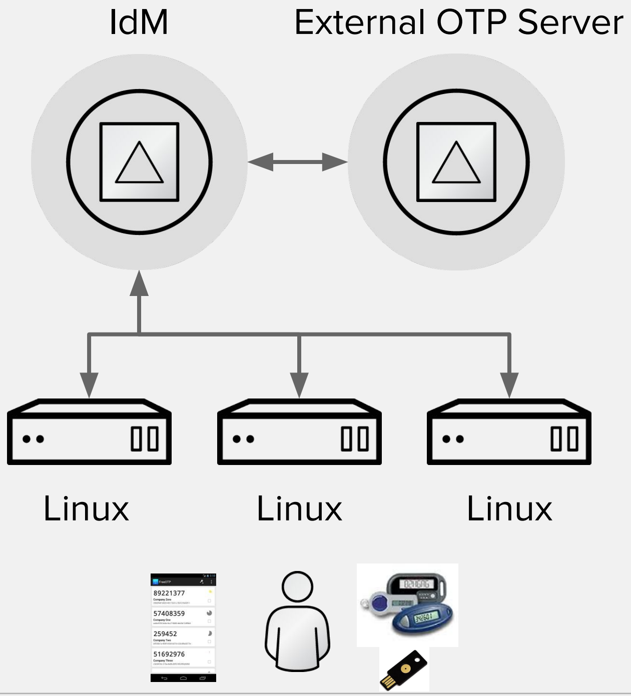


- Define direct or indirect maps
- Associate maps with a particular location
- Configure clients to pull data from that location (part of the LDAP tree)
- Maps are defined centrally
- Maps are applied on the client
- Maps are cached
- Maps are integrated with autofs

Use Cases and Challenges

- How can I provide centralized authentication?
- Can I provide centralized password policies?
- Can I define access control to hosts without copying configuration files?
- Can I manage SSH keys for users and hosts?
- Can I provide centralized SUDO, automount, SELinux user mappings?
- **Is there a cost effective solution that provides strong authentication using OTP?**
- Can I simplify SSL certificate management?

Two-factor Authentication



IdM

External OTP Server

Linux

Linux

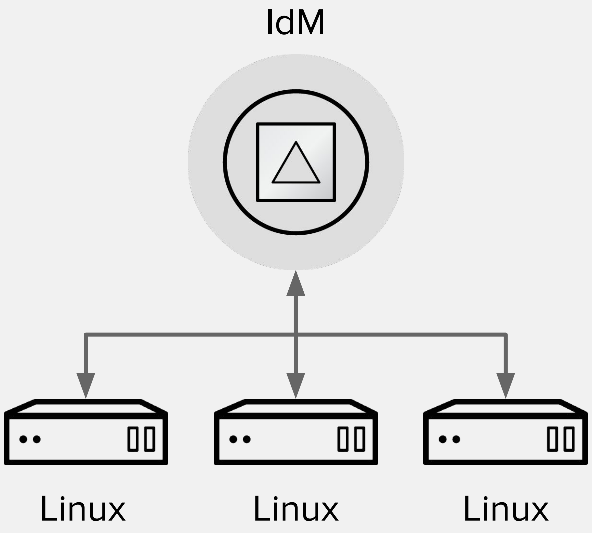
Linux

- Native 2FA
  - Yubikey, FreeOTP, Google authenticator
  - HOTP/TOTP compatible
  - Over LDAP or Kerberos
- Proxied over RADIUS
  - Any third party that has RADIUS support
  - Kerberos only
- Easy migration

Use Cases and Challenges

- How can I provide centralized authentication?
- Can I provide centralized password policies?
- Can I define access control to hosts without copying configuration files?
- Can I manage SSH keys for users and hosts?
- Can I provide centralized SUDO, automount, SELinux user mappings?
- Is there a cost effective solution that provides strong authentication using OTP?
- **Can I simplify SSL certificate management?**

SSL Certificate Management



Steps:

- Install and configure IPA client
- Request new certificate
  - *ipa-getcert request ...*

Why would I use IdM?

- Centrally manage SSL certificates
  - Sign, issue and revoke
- Automate SSL certificate deployments
- Auto-renewal of certificates

Wrap-up

Resources

Summary

- Linux Domain Identity, Authentication, and Policy Guide
  - https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Linux_Domain_Identity_Authentication_and_Policy_Guide/index.html
- Windows Integration Guide
  - https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/Windows_Integration_Guide/index.html
- System-Level Authentication Guide
  - https://access.redhat.com/site/documentation/en-US/Red_Hat_Enterprise_Linux/7/html/System-Level_Authentication_Guide/index.html

Resources

Summary

- FreeIPA
  - Project wiki: www.freeipa.org
  - Project trac: https://fedorahosted.org/freeipa/
  - Code: http://git.fedorahosted.org/git/?p=freeipa.git
  - Mailing lists:
    - freeipa-users@redhat.com
    - freeipa-devel@redhat.com
    - freeipa-interest@redhat.com
- SSSD: https://fedorahosted.org/sssd/
  - Mailing lists:
    - sssd-devel@lists.fedorahosted.org
    - sssd-users@lists.fedorahosted.org

Training Materials and Blogs

- Training
  - http://www.freeipa.org/page/Documentation#FreeIPA_Training_Series
- Blog aggregation
  - http://planet.freeipa.org/
- FreeIPA demo instance in the cloud
  - http://www.freeipa.org/page/Demo

Questions?

Finally

Identity Management (IdM)

IdM

Features

- Centralized authentication via Kerberos or LDAP
- Identity management:
  - users, groups, hosts, host groups, netgroups, services
  - user lifecycle management
- Manageability:
  - Simple installation scripts for server and client
  - Rich CLI and web-based user interface
  - Pluggable and extensible framework for UI/CLI
  - Flexible delegation and administrative model
    - Self, delegated, role based; read permissions

IdM

Features Continued

- Host-based access control
- Centrally-managed SUDO
- SSH key management
- Group-based password policies
- Automatic management of private groups
- Can act as NIS server for legacy systems
- Painless password migration
- SELinux user mapping
- Auto-membership for hosts and users
- Serving sets of automount maps to different clients
- Different POSIX data and SSH keys for different sets of hosts

IdM

Features DNS

- DNS is optional but convenient
- Advantages (automation and security):
    - The SRV records get created automatically
    - Host records get created automatically when hosts are added
    - The clients can update their DNS records in a secure way (GSS-TSIG)
    - The admin can delegate management of the zones to whomever he likes
    - Built in DNSSEC support (Tech Preview in RHEL 7.2)
- Disadvantages:
    - You need to delegate a zone

IdM

More Features

- Replication:
  - Supports multi-server deployment based on the multi-master replication (up to 20 replicas)
  - Recommended deployment 2K-3K clients per replica
  - Details depend on the number of data centers and their geo location
- 2FA
  - Native HOTP/TOTP support with FreeOTP and Yubikey
  - Proxied 2FA authentication over RADIUS for other solutions
  - 2FA for AD users (in works)
- Backup and Restore
- Compatibility with broad set of clients (LINUX/UNIX)

IdM

Features PKI

- CA related capabilities
  - Certificate provisioning for users (new in RHEL-7.2), hosts and services
  - Multiple certificate profiles (new in RHEL-7.2)
  - Sub CAs (in works)
- CA deployment types
  - CA-less
  - Chained to other CA
  - Self-signed root
- Tool to change deployment type and rotate CA keys
  - Flexibility in deploying CAs on different replicas
- Key store (Vault) - new in RHEL-7.2

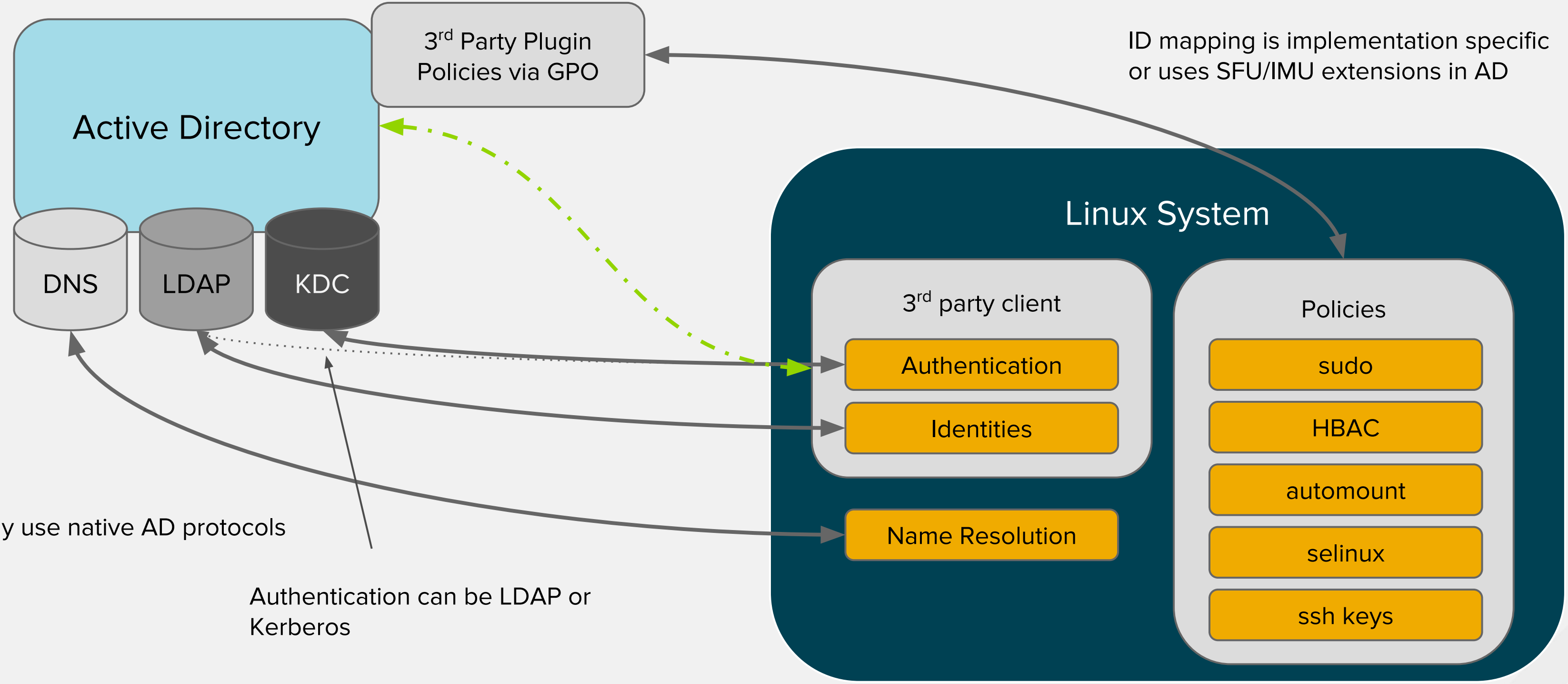
Active Directory Integration

Direct Integration Options

Outline

- 3rd party
- Legacy (pam_krb5/pam_ldap, nss_ldap, nslcd)
- Traditional - winbind
- Contemporary - SSSD (with realmd)

Third Party Direct Integration



ID mapping is implementation specific or uses SFU/IMU extensions in AD

Client may use native AD protocols

Authentication can be LDAP or Kerberos

Third Party Direct Integration

Pros and Cons

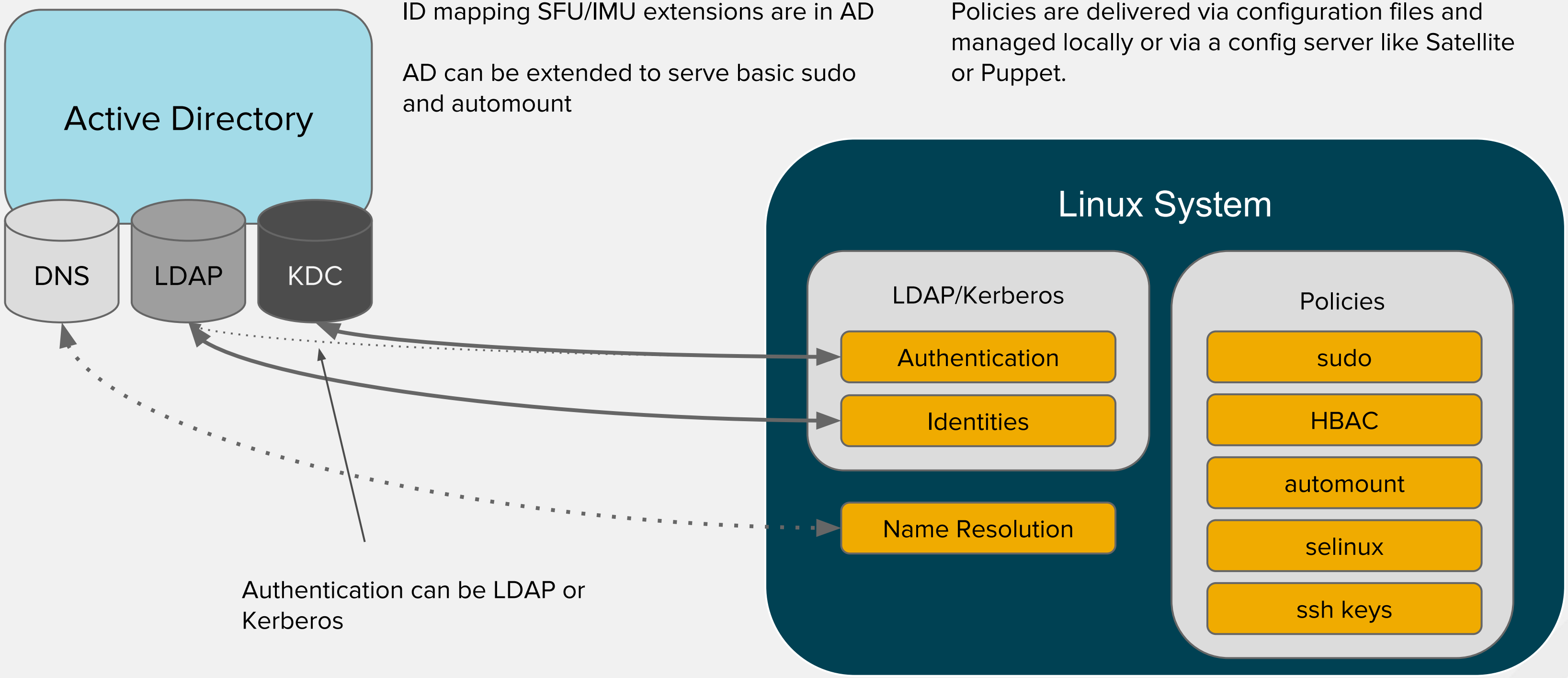
- Pros
  - Everything is managed in one place including policies
  - SSO can be accomplished via Kerberos
- Cons
  - Requires third party vendor
  - Extra cost per system (adds up)
  - Limits UNIX/Linux environment independence
  - Requires software on AD side
  - OTP support unclear (Azure)

Legacy Direct Integration

ID mapping SFU/IMU extensions are in AD

AD can be extended to serve basic sudo and automount

Policies are delivered via configuration files and managed locally or via a config server like Satellite or Puppet.



Authentication can be LDAP or Kerberos

Legacy Direct Integration

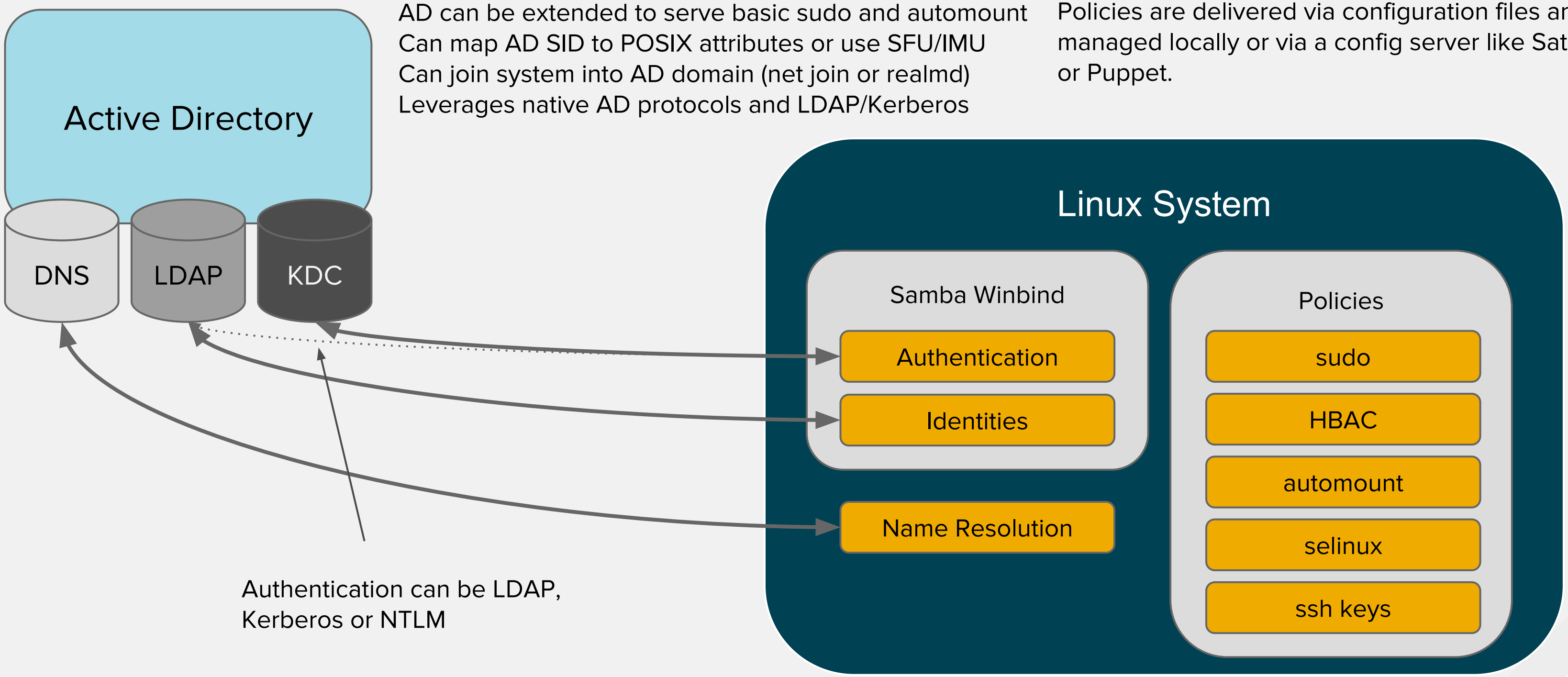
Pros and Cons

- Pros:
  - Free
  - No third party vendor is needed
  - Intuitive
  - LDAP OTP authentication in Azure (have not tried)
  - Available on UNIXes
- Cons:
  - Requires SFU/IMU AD extension (which are deprecated as of fall 2014)
  - Policies are not centrally managed
  - Hard to configure securely
  - No SSO with OTP

Traditional Direct Integration

AD can be extended to serve basic sudo and automount
Can map AD SID to POSIX attributes or use SFU/IMU
Can join system into AD domain (net join or realmd)
Leverages native AD protocols and LDAP/Kerberos

Policies are delivered via configuration files and managed locally or via a config server like Satellite or Puppet.



Authentication can be LDAP, Kerberos or NTLM

Traditional Direct Integration

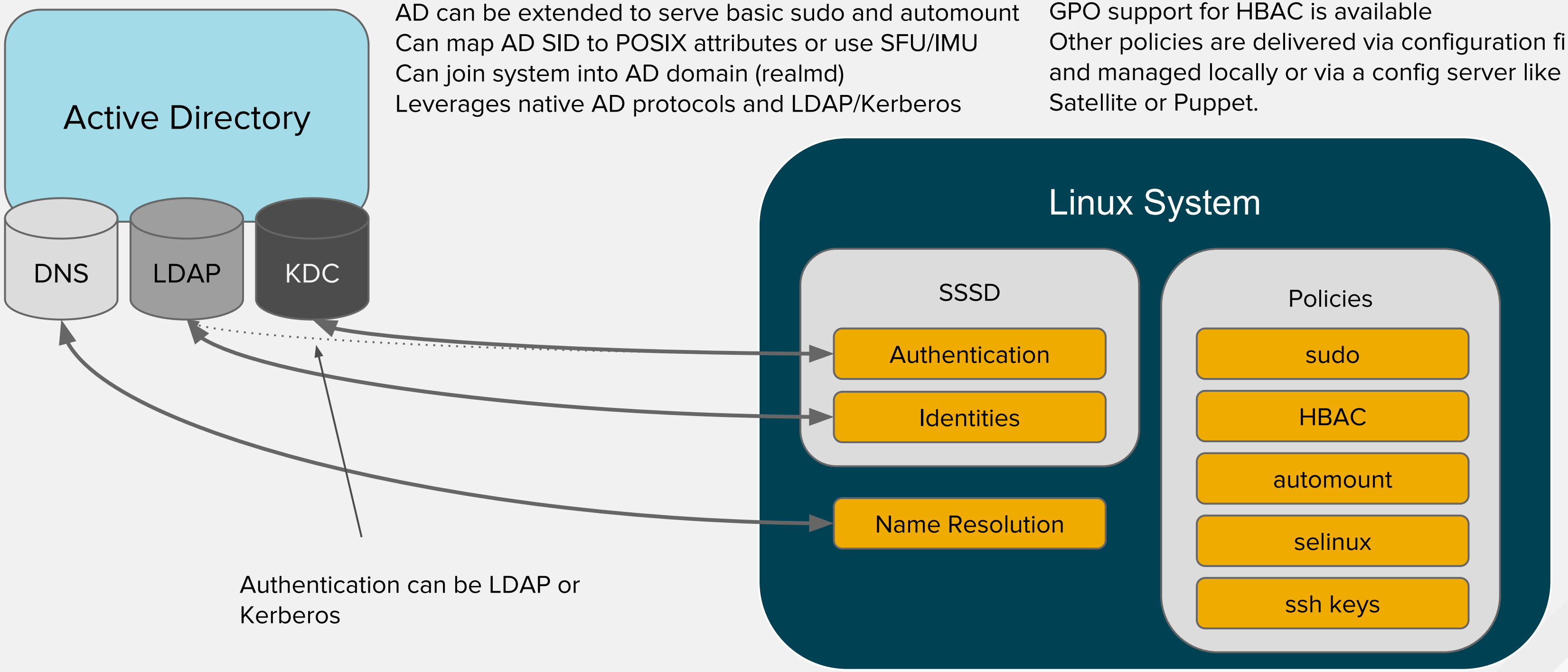
Pros and Cons

- Pros:
    - Well known
    - Does not require third party
    - Does not require SFU/IMU
    - Supports trusted domains
    - Supports CIFS client and Samba FS integration
- Cons:
    - Can connect only to AD and very MSFT focused
    - Has some perceived stability issues
    - Policies are not centrally managed
    - No OTP support

SSSD Based Direct Integration

AD can be extended to serve basic sudo and automount
Can map AD SID to POSIX attributes or use SFU/IMU
Can join system into AD domain (realmd)
Leverages native AD protocols and LDAP/Kerberos

GPO support for HBAC is available
Other policies are delivered via configuration files and managed locally or via a config server like Satellite or Puppet.



Authentication can be LDAP or Kerberos

SSSD Based Direct Integration

Pros and Cons

- Pros:
  - Does not require SFU/IMU but can use them
  - Can be used with different identity sources
  - Support transitive trusts in AD domains and forest trusts with FreeIPA
  - Supports CIFS client and Samba FS integration
  - GPO for Windows based HBAC
- Cons:
  - No NTLM support, no support for AD forest trusts (yet)
  - No SSO with OTP
  - Not all policies are centrally managed

Direct Integration

Option Summary

Please read my blog :-)

http://rhelblog.redhat.com/author/dpalsecam/

Comparison:

http://rhelblog.redhat.com/2015/02/04/overview-of-direct-integration-options/

Direct Integration

Bottom line

- SSSD is the way to go
- Winbind is the fallback option:
  - if you rely on NTLM (please do not, it is very insecure)
  - If you have multiple forests and need users from different forests to access the Linux system
- Policy management is still not fully central
- Might require deprecated extensions on the AD side
- Per system CALs add to cost
- Linux/UNIX administrators do not have control over the environment

redhat.

THANK YOU

- plus.google.com/+RedHat
- facebook.com/redhatinc
- linkedin.com/company/red-hat
- twitter.com/RedHatNews
- youtube.com/user/RedHatVideos