redhat.

Identity Management for Modern Infrastructure

Worcester Linux Users' Group

Dmitri Pal
Director of Engineering, Red Hat, Inc.
12/10/2015

AGENDA

Worcester Linux Users' Group

7:00PM—7:20PM
Identity Management High Level View

7:20PM—7:35PM
Introduction to Kerberos

7:35PM—8:00PM
Active Directory Integration

8:00PM—8:25PM
FreeIPA/IdM Introduction

8:25PM—8:40PM
Indirect Integration using FreeIPA/IdM

8:40PM—8:50PM
Application Integration

8:50PM—9:00PM
Questions

redhat.

Identity Management High Level View

Modern Enterprise

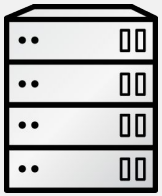
Simplified view

redhat.

Modern Enterprise

Simplified view

Servers



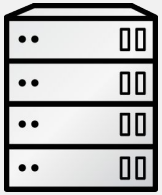
Windows
Linux
UNIX

redhat.

Modern Enterprise

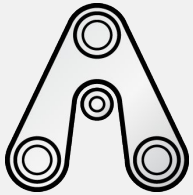
Simplified view

Servers



Windows
Linux
UNIX

Services



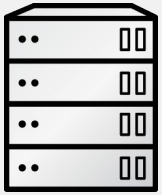
Internal
and
External

redhat.

Modern Enterprise

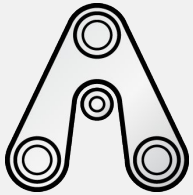
Simplified view

Servers



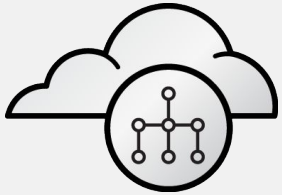
Windows
Linux
UNIX

Services



Internal
and
External

Clouds



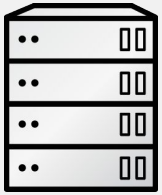
Private
and
Public

redhat.

Modern Enterprise

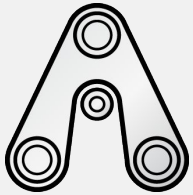
Simplified view

Servers



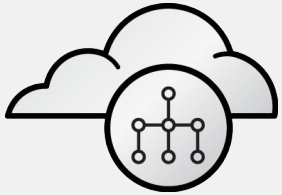
Windows
Linux
UNIX

Services



Internal
and
External

Clouds



Private
and
Public

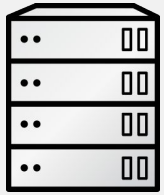
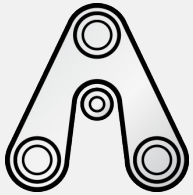
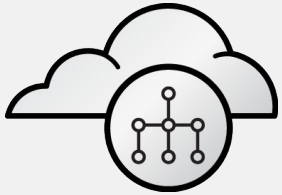
Applications

BM
VM
Container

redhat.

Modern Enterprise

Simplified view

Servers	Services	Clouds	Applications	Tools
				
Windows Linux UNIX	Internal and External	Private and Public	BM VM Container	Developer QE/QA DevOp/IT

redhat.

Modern Enterprise

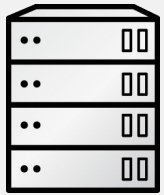
Identity View

redhat.

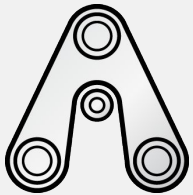
Modern Enterprise

Identity View

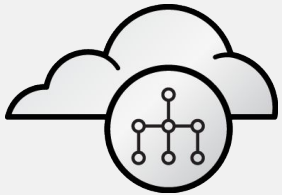
Servers



Services

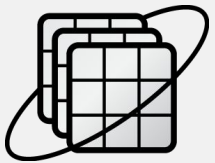


Clouds



Applications

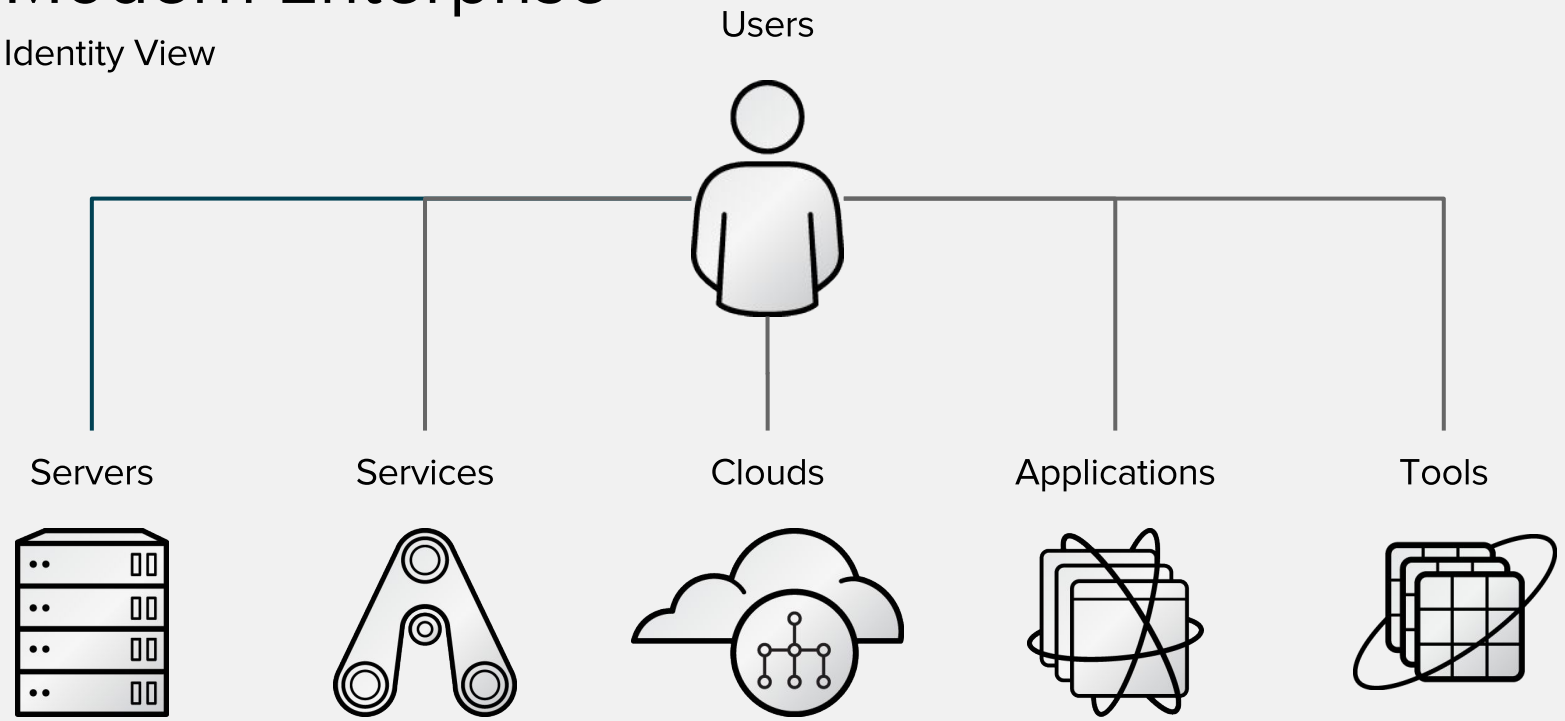
Tools



redhat.

Modern Enterprise

Identity View



redhat.

Users

In Modern Enterprise

Employees
Contractors

Customers
Partners

redhat.

Users

In Modern Enterprise

Internal Namespace

Employees
Contractors

External Namespace

Customers
Partners

redhat.

Internal Namespace

Focus of this presentation

Internal Namespace

Employees
Contractors

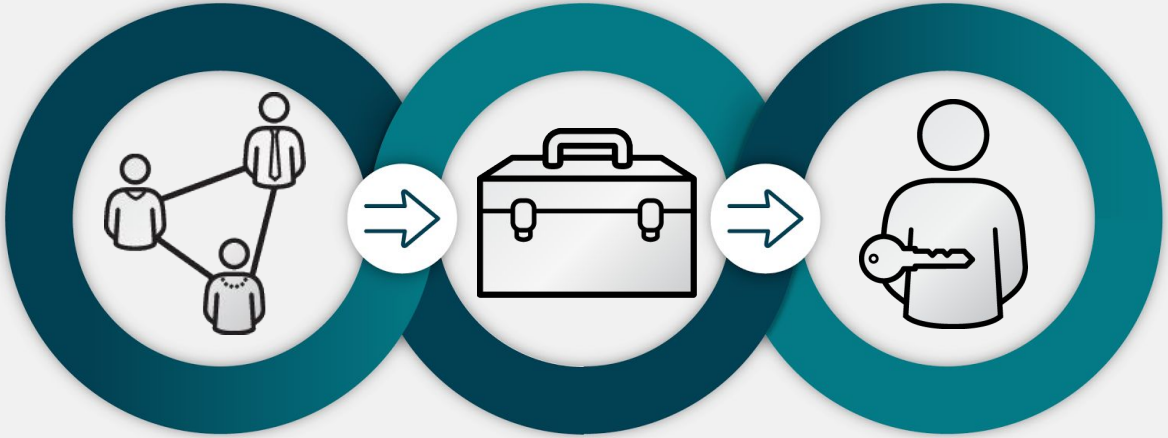
External Namespace

Customers
Partners

redhat.

Internal Namespace

Traditional Model



HR Database

Identity Management System

User Storage per Application

redhat.

Traditional Identity Model

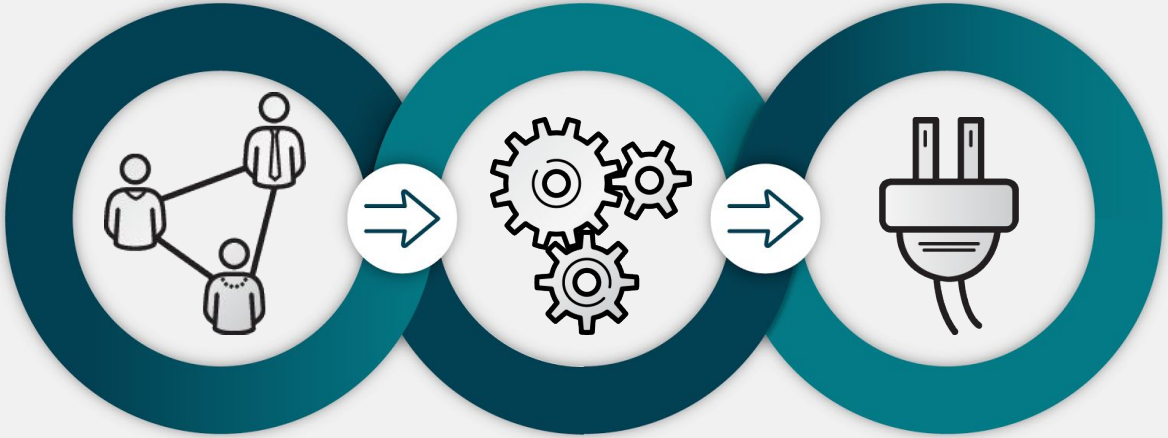
Weaknesses

- Complex
- Costly
- Applications are isolated
- Hard to manage and make sure that all systems are aligned
- Hard to be compliant with different regulations

redhat.

Internal Namespace

Modern Model



HR Database

Central Identity Store

Application Users from a single store

redhat.

Modern Identity Model

Pros and Cons

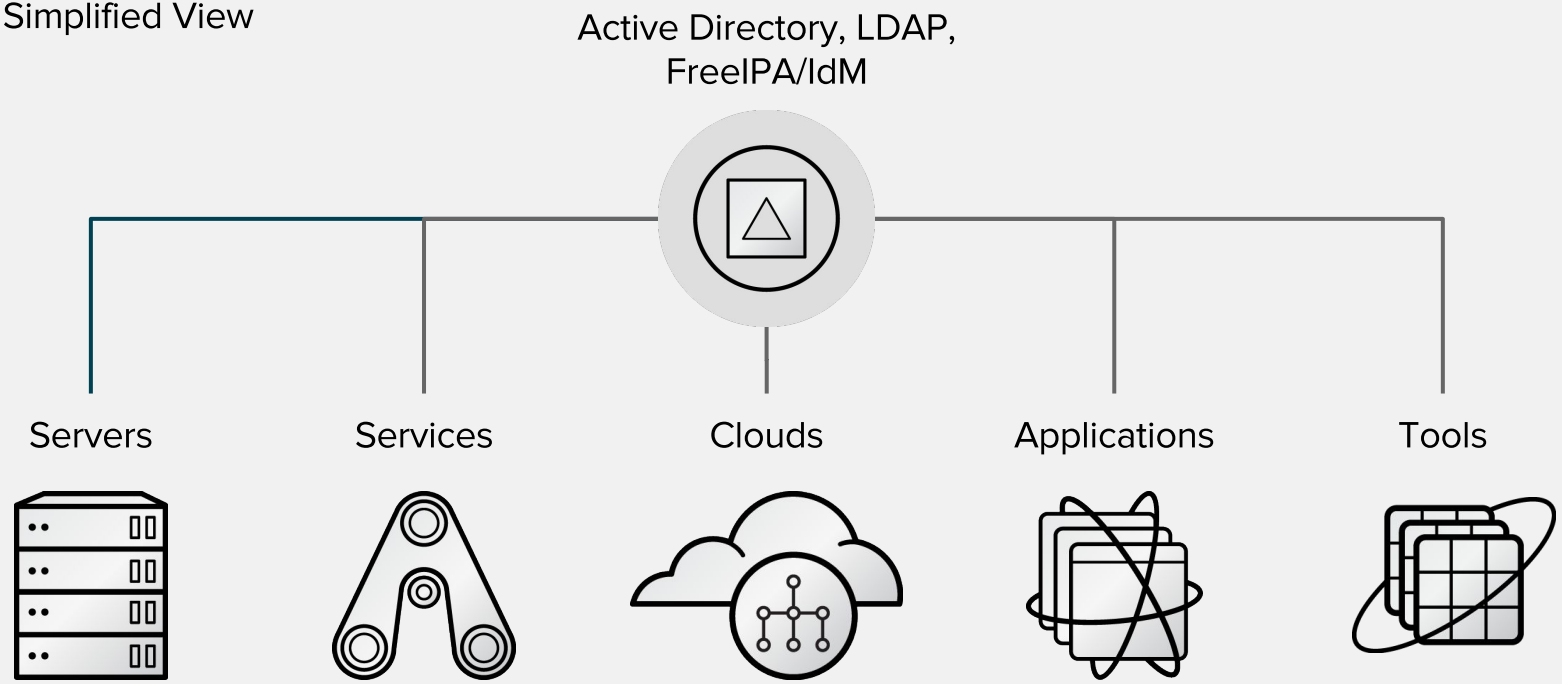
- Less complex, however, still has a fair amount of complexity
- Less costly
- Applications plug into a common identity source directly and indirectly but still need additional data
- Easier to manage but still has challenges

There is no silver bullet but the more control you have the better the result is.

redhat.

Modern Identity Model

Simplified View



redhat.

Modern Identity Model

Challenges if you use a pure LDAP solution

- Usually a home grown custom solution that is hard to support
  - A lot of craft and magic
  - Hard to modernize
  - Costly to maintain
- Windows client systems still require AD
- How you deal with SSO?

redhat.

How to deal with SSO

Overview

- Platform level:
  - ~~NTLM~~ - old, weak crypto, should not be used
  - **Kerberos** - old, went a long way, recommended
- Application level:
  - ~~OpenID~~ - old, has weaknesses, should not be used
  - **SAML** - old, proven, recommended, challenges with mobile
  - **OpenID Connect (OIDC)** - modern, proven, recommended for new applications

redhat.

How to deal with SSO

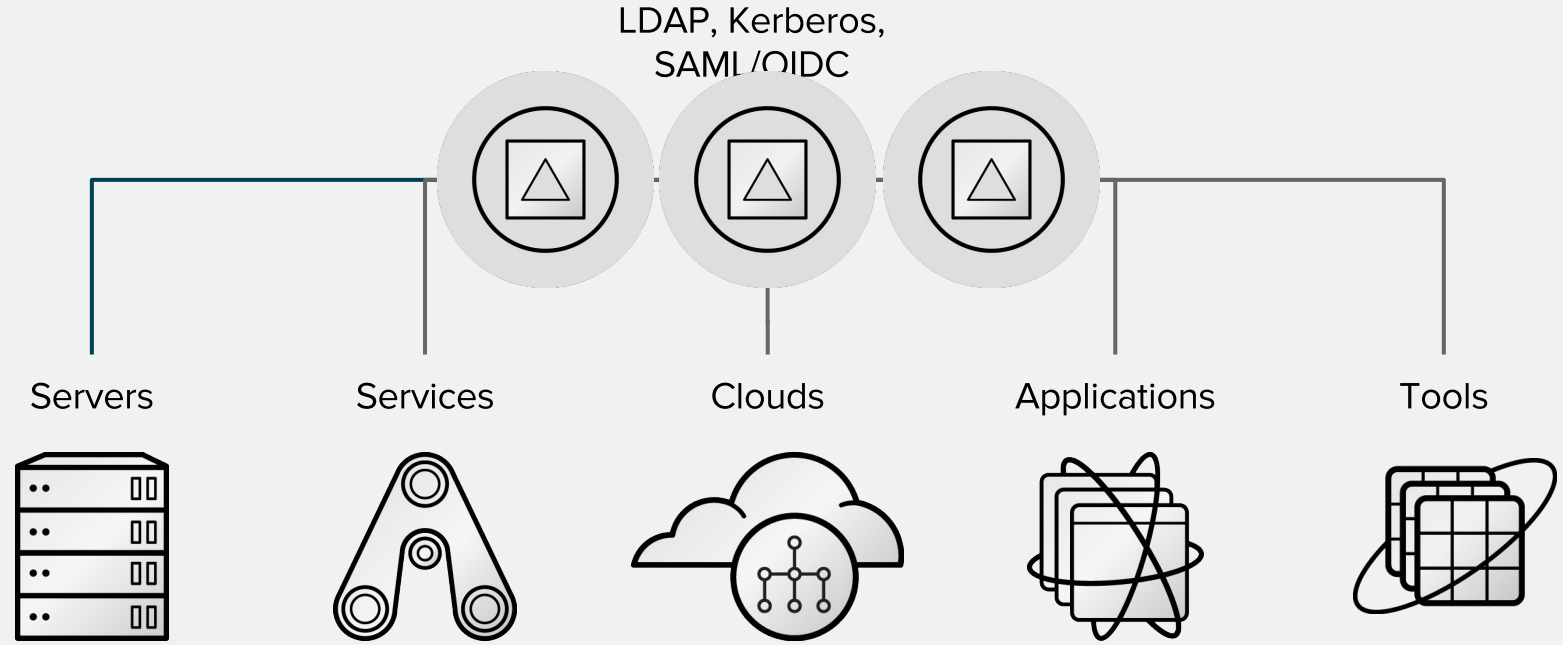
Bottom Line

Use Kerberos, SAML, OIDC and a combination of them based on the use case.

redhat.

Modern Identity Model

LDAP Solution



redhat.

LDAP Based Solution

Recommendations

- Since the solution has many limitations, costly and hard to maintain and migrate from the recommendation is not to embark on that path
- If you already have it, consider moving off it instead of continued investment into it. The more you invest, the harder it will be to modernize and move off in future.

redhat.

NIS

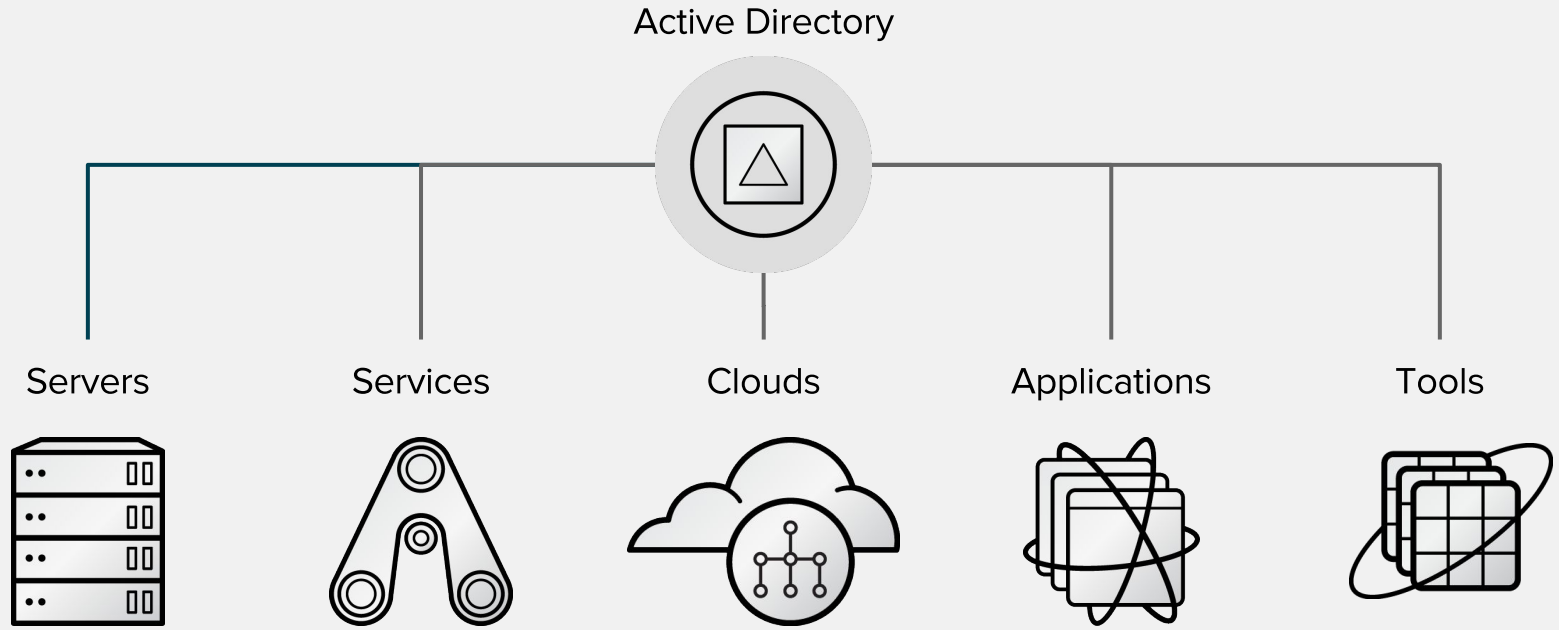
Added for completeness

- NIS is a UNIX focused solution of the past
- It is very limited and insecure
- It does not meet modern compliance requirements
- If you have it move off ASAP

redhat.

Modern Identity Model

Active Directory based solution



redhat.

Modern Identity Model

Challenges if you use an AD focused solution

- How to deal with policies and access control?
- How to handle different domains and forests?
- How to deal with lack of control over Active Directory?
- How to handle POSIX?
- How SSO works in this case?
- How deal with additional data that needs to be stored somewhere?

redhat.

Modern Identity Model

Challenges if you use an AD focused solution

- How to deal with policies and access control?
- How to handle different domains and forests?
- How to deal with lack of control over Active Directory?
- How to handle POSIX?
- How SSO works in this case?
- How deal with additional data that needs to be stored somewhere?

A section will be dedicated to this conversation

redhat.

Modern Identity Model

Challenges if use a FreeIPA/IdM solution

- Windows client systems still require AD… A non starter?

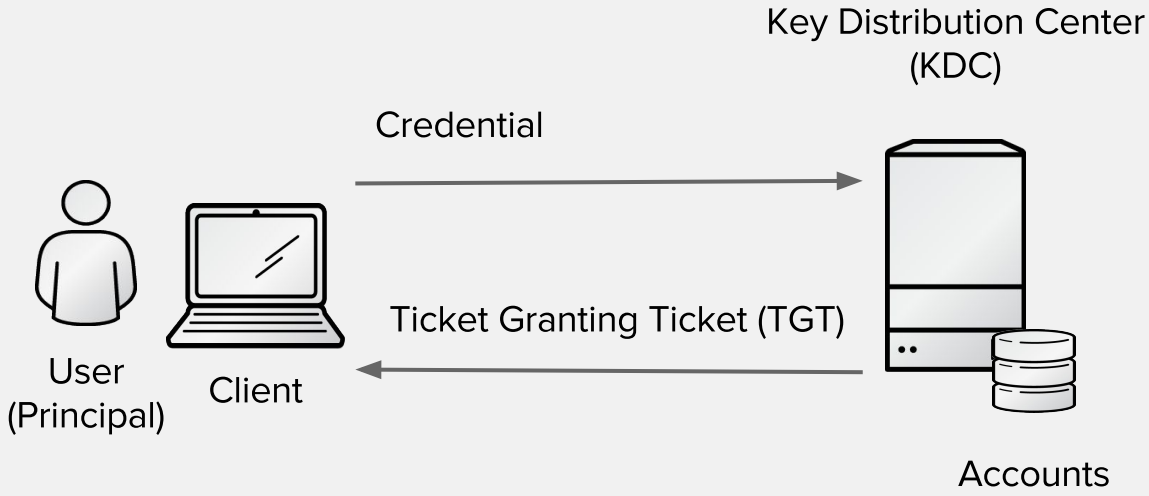
Another section will be dedicated to this conversation

redhat.


Introduction to Kerberos
redhat.

Kerberos Authentication

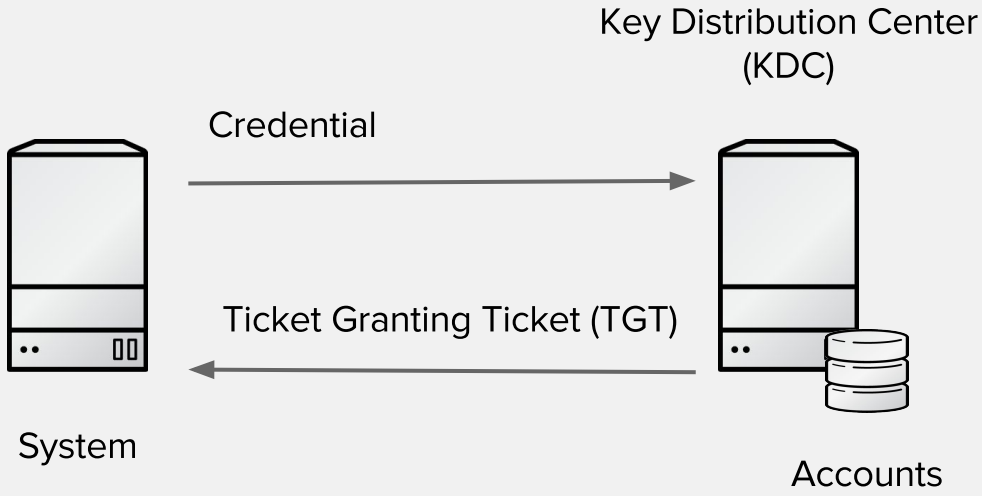
User authentication



redhat.

Kerberos Authentication

System authentication



redhat.

Credentials in Kerberos

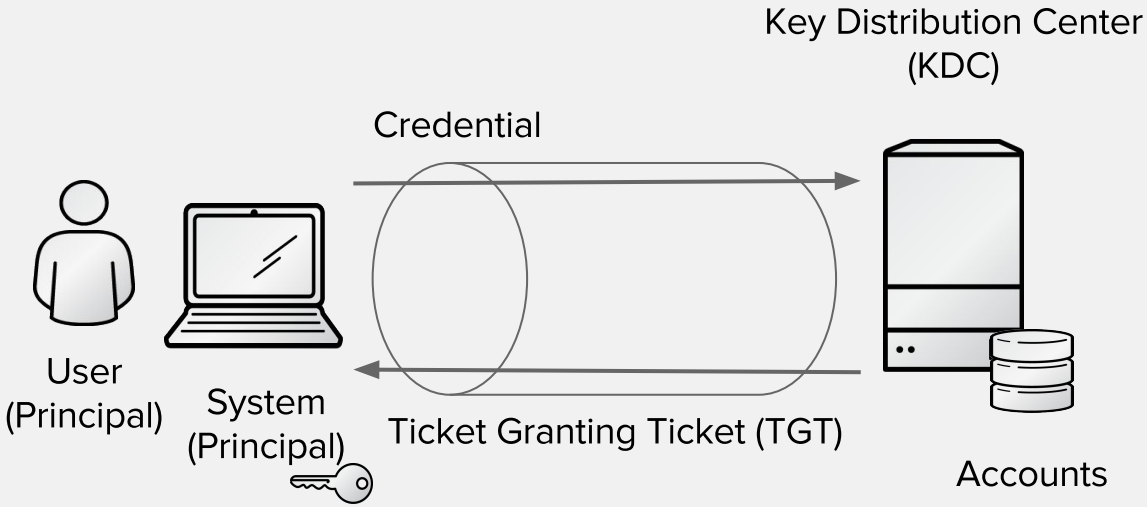
Overview

- Password/Key
  - User password - something that user remembers
  - System key - long and strong that can be used by a system
- Certificate (pkinit)
- OTP
  - Tunnelled
  - SPAKE

redhat.

Tunnelled Authentication

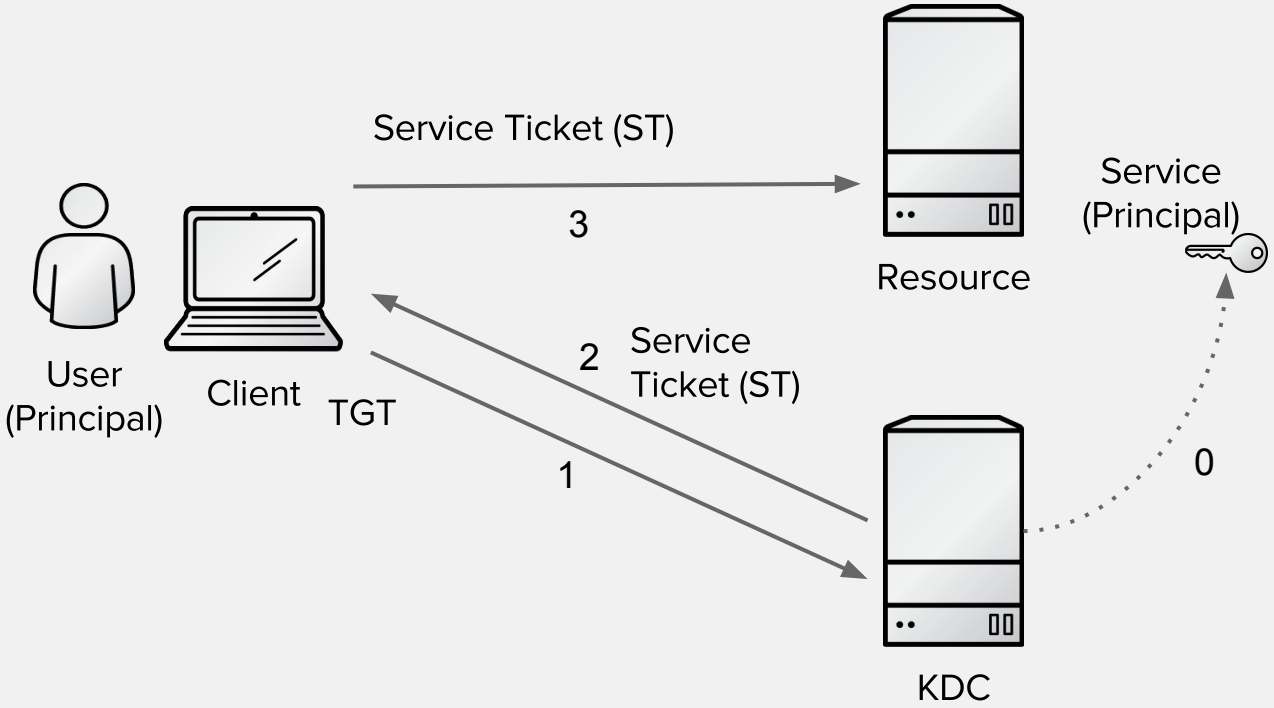
Secure Tunnel



redhat.

Kerberos SSO

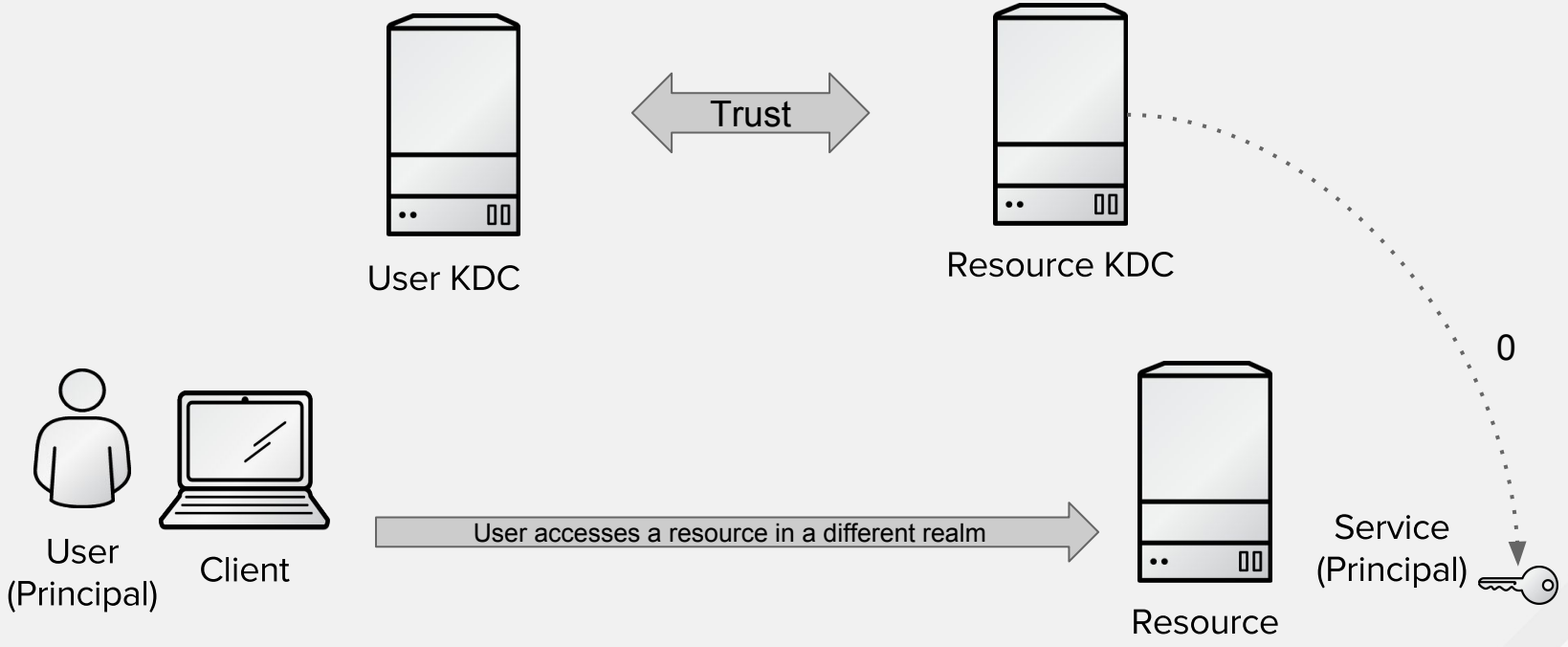
Accessing a resource



redhat.

Trust Case

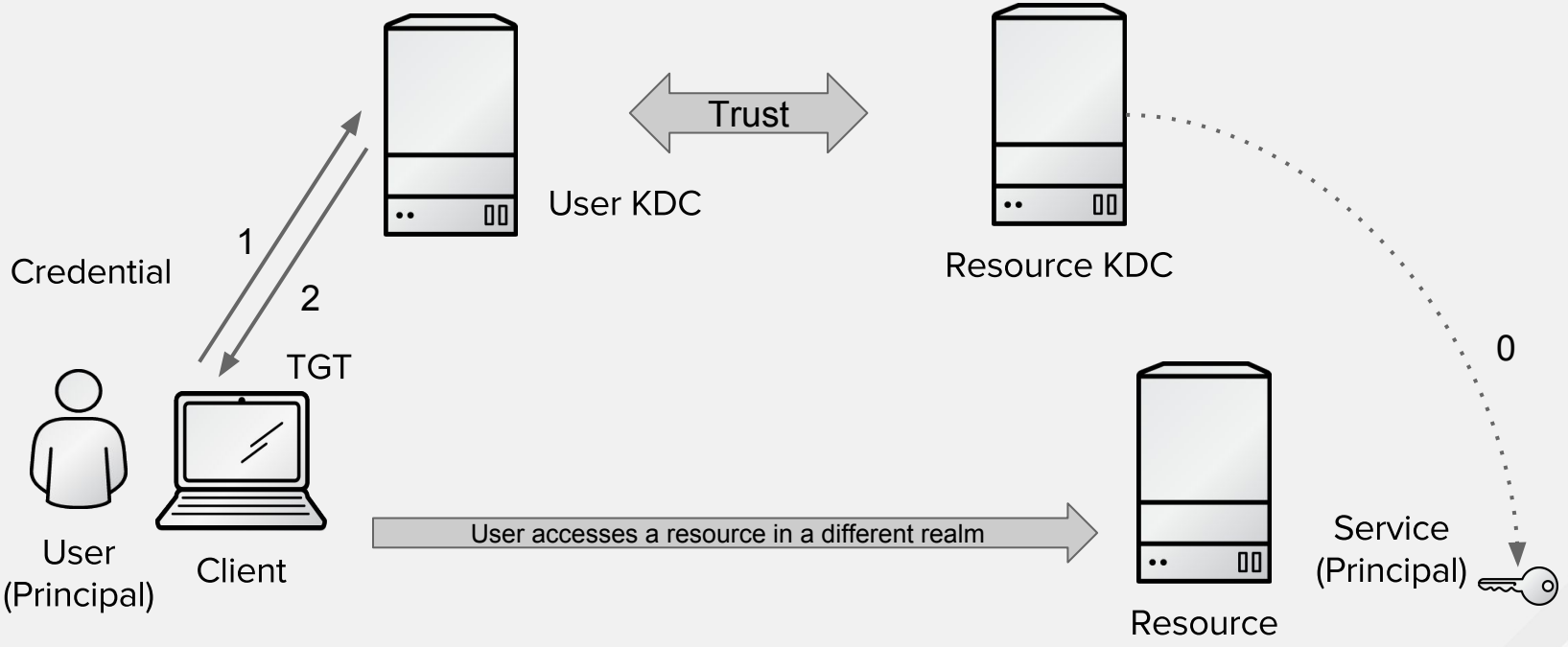
Accessing a resource in different Realm



redhat.

Trust Case

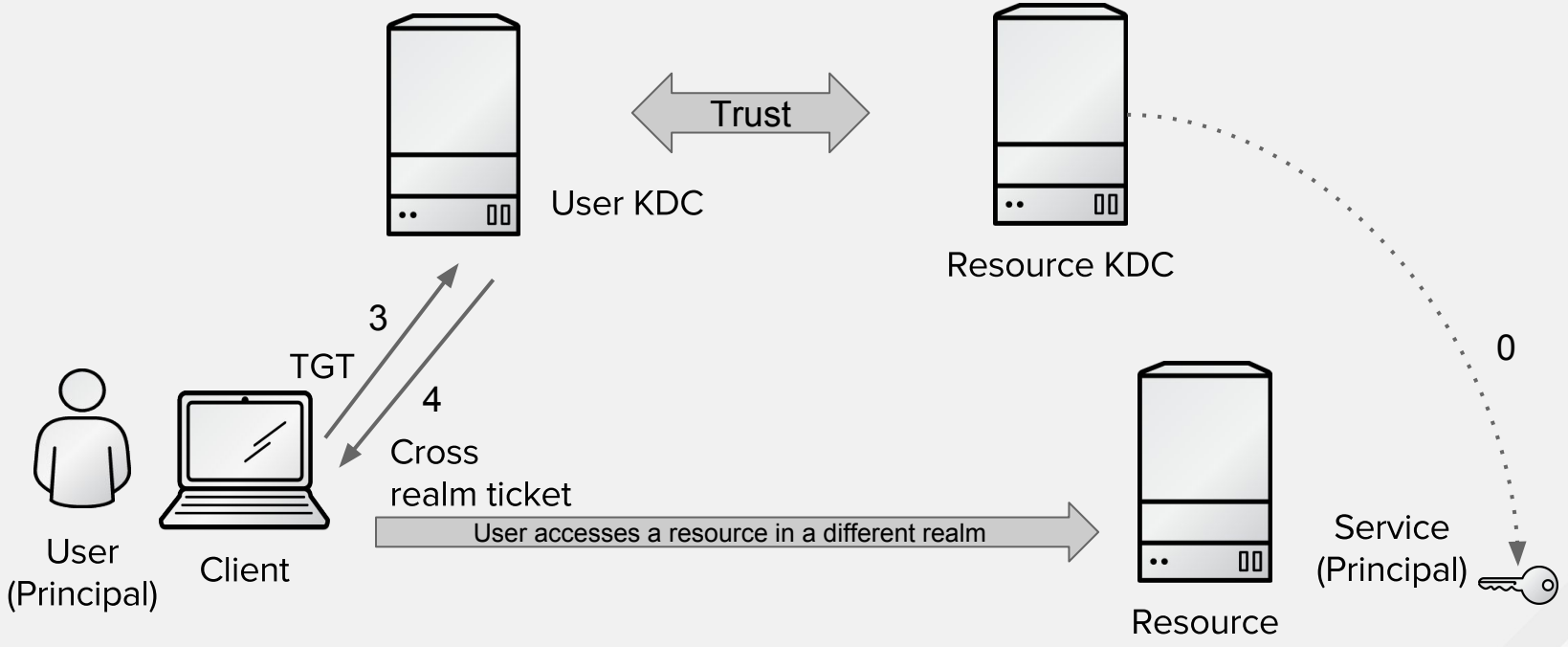
Accessing a resource in different Realm



redhat.

Trust Case

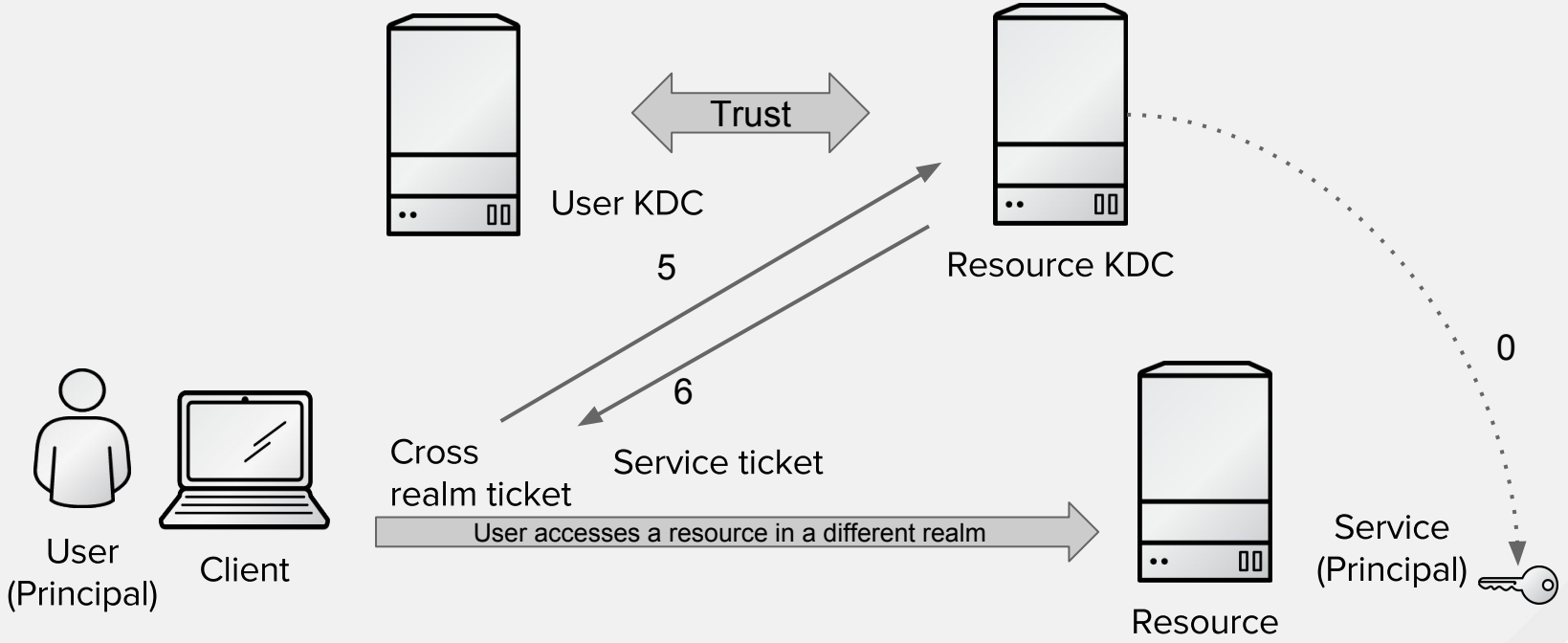
Accessing a resource in different Realm



redhat.

Trust Case

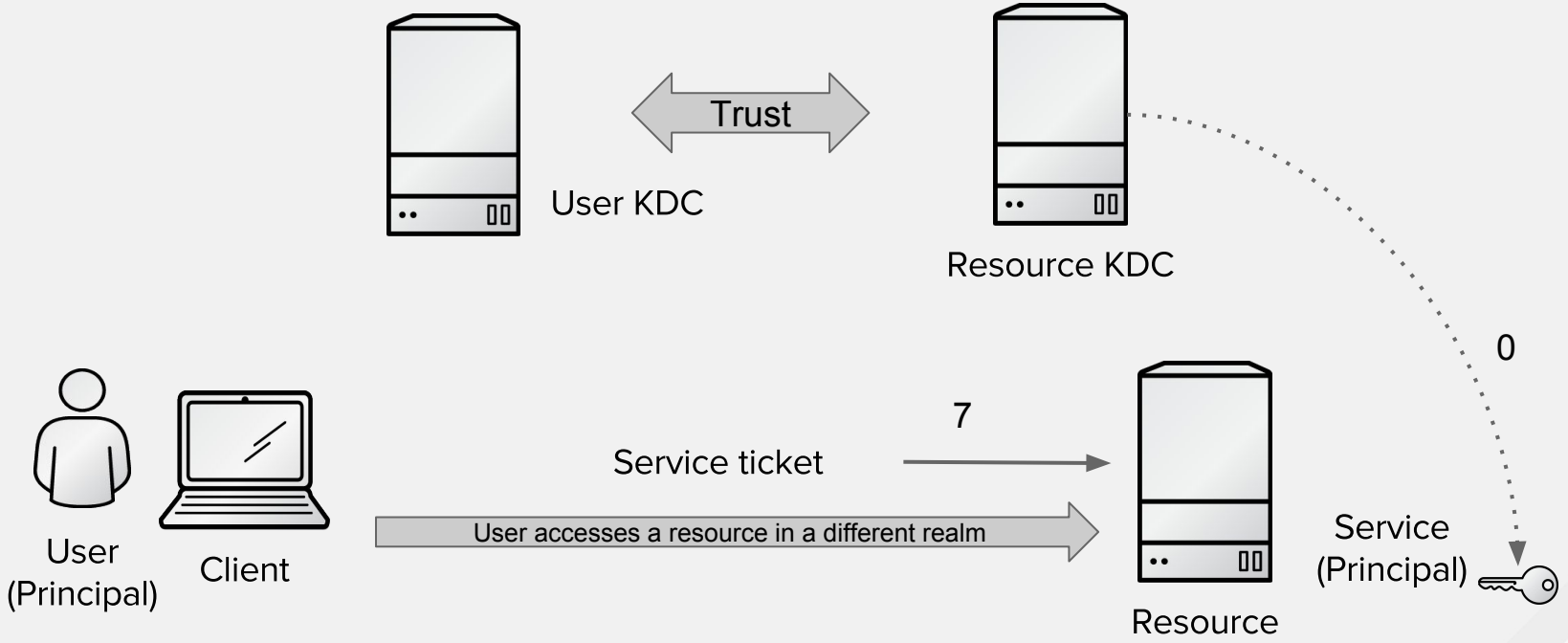
Accessing a resource in different Realm



redhat.

Trust Case

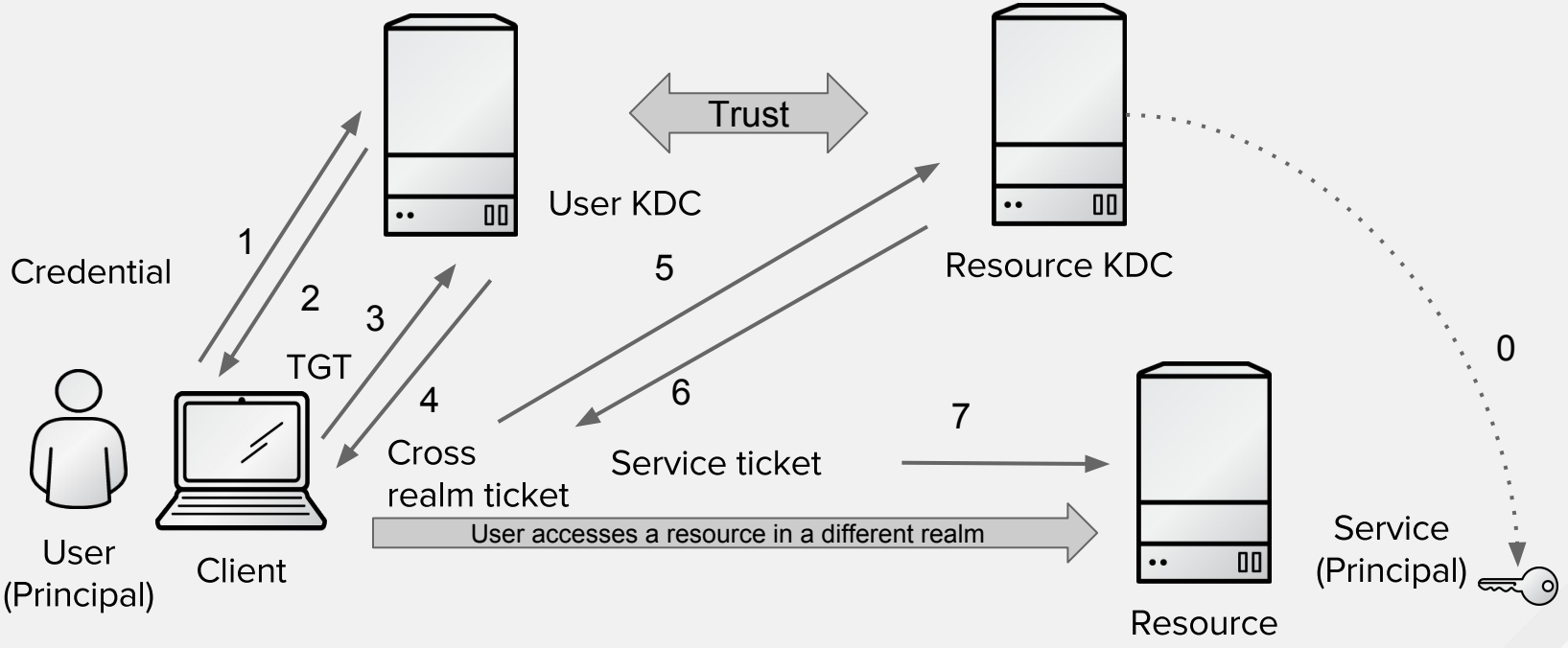
Accessing a resource in different Realm



redhat.

Trust Case

Accessing a resource in different Realm

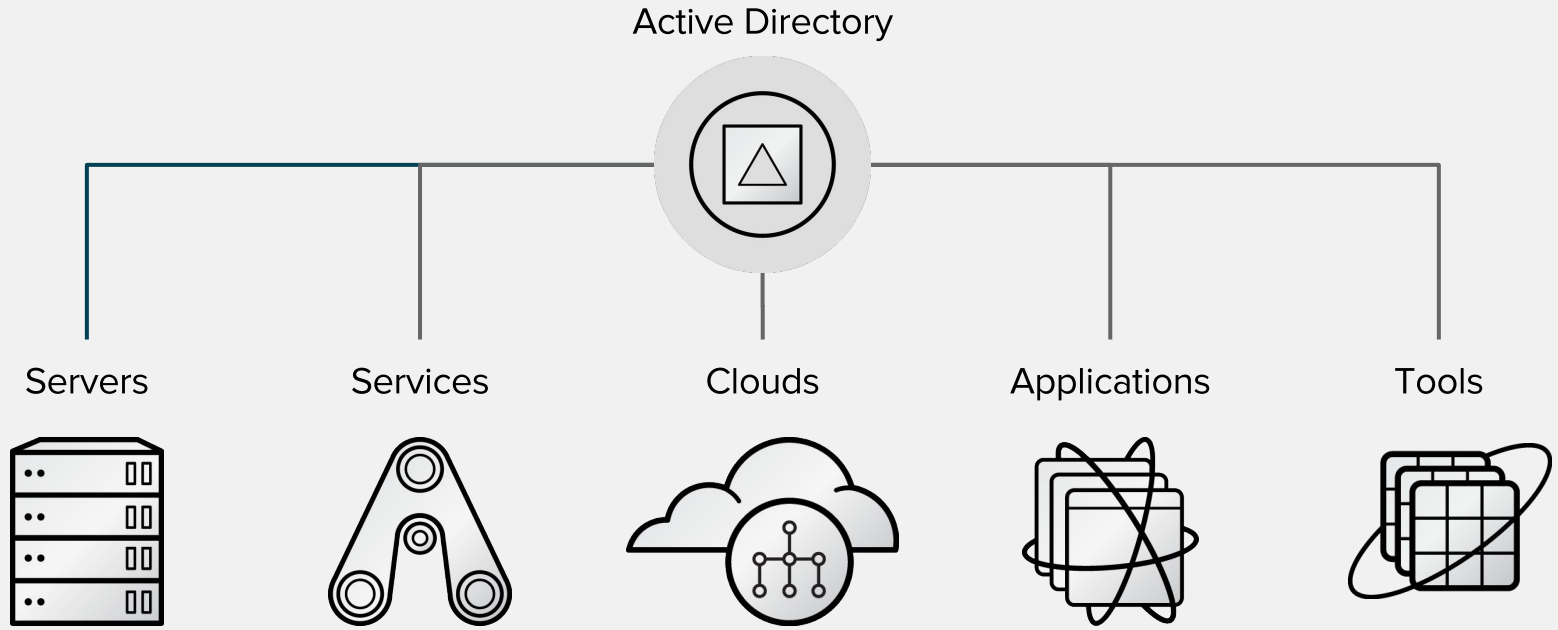


redhat.

Active Directory Integration

Modern Identity Model

Active Directory based solution



redhat.

Active Directory Integration

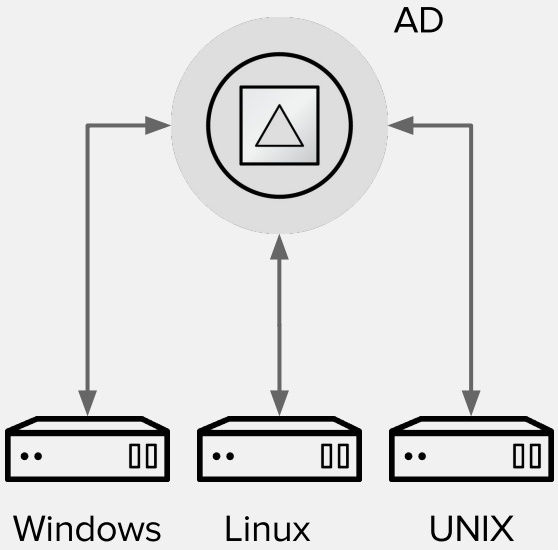
Aspects of Integration

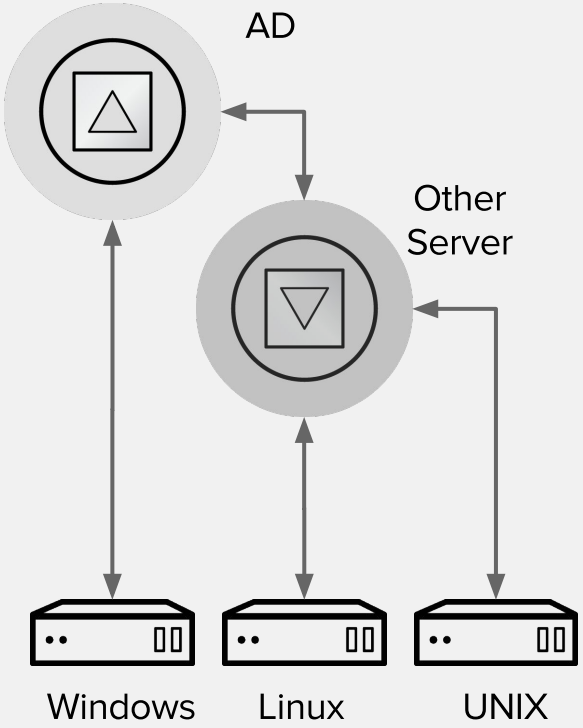
- Authentication
  - User logs into a Linux system or application/service, how is he authenticated?
- Identity lookup
  - How system/application/service knows about the right accounts?
  - How AD accounts are mapped to POSIX?
- Name resolution and service discovery
  - How system/service/application knows where is its authentication and identity server?
- Policy management
  - How other identity related policies are managed on the system?
  - How access control is determined for service or application?

redhat.

Connecting Systems

Integration Options

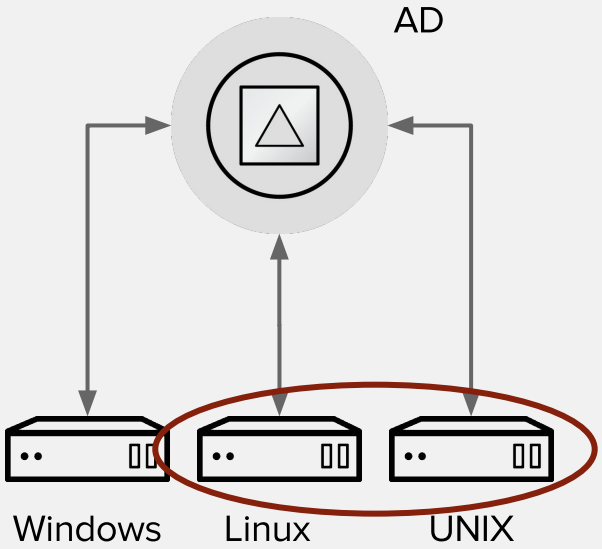


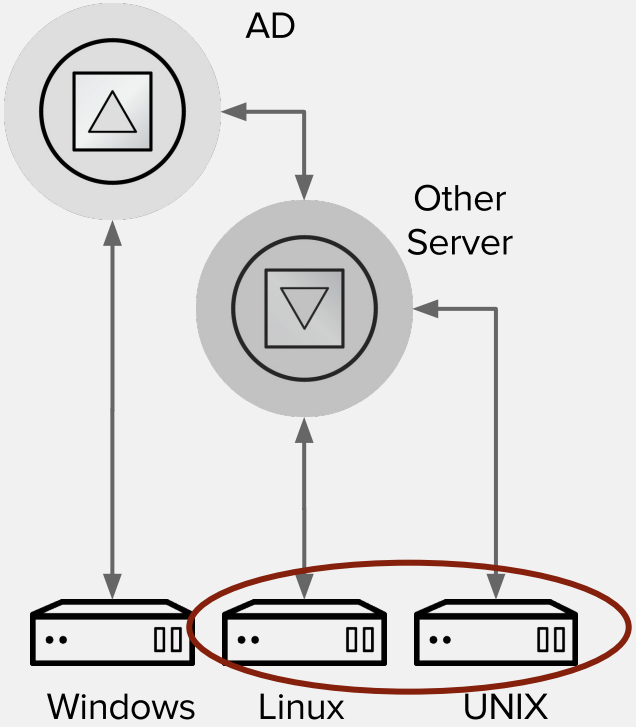


redhat.

Connecting Systems

Focus of the presentation





redhat.

Integration Concerns

Questions to ask

- How can I provide centralized authentication?
- 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 provide a smooth SSO experience for my users inside the enterprise?
- How can I integrate my applications into the same identity space?
- How to address Active Directory interoperability challenges?
- ...

redhat.

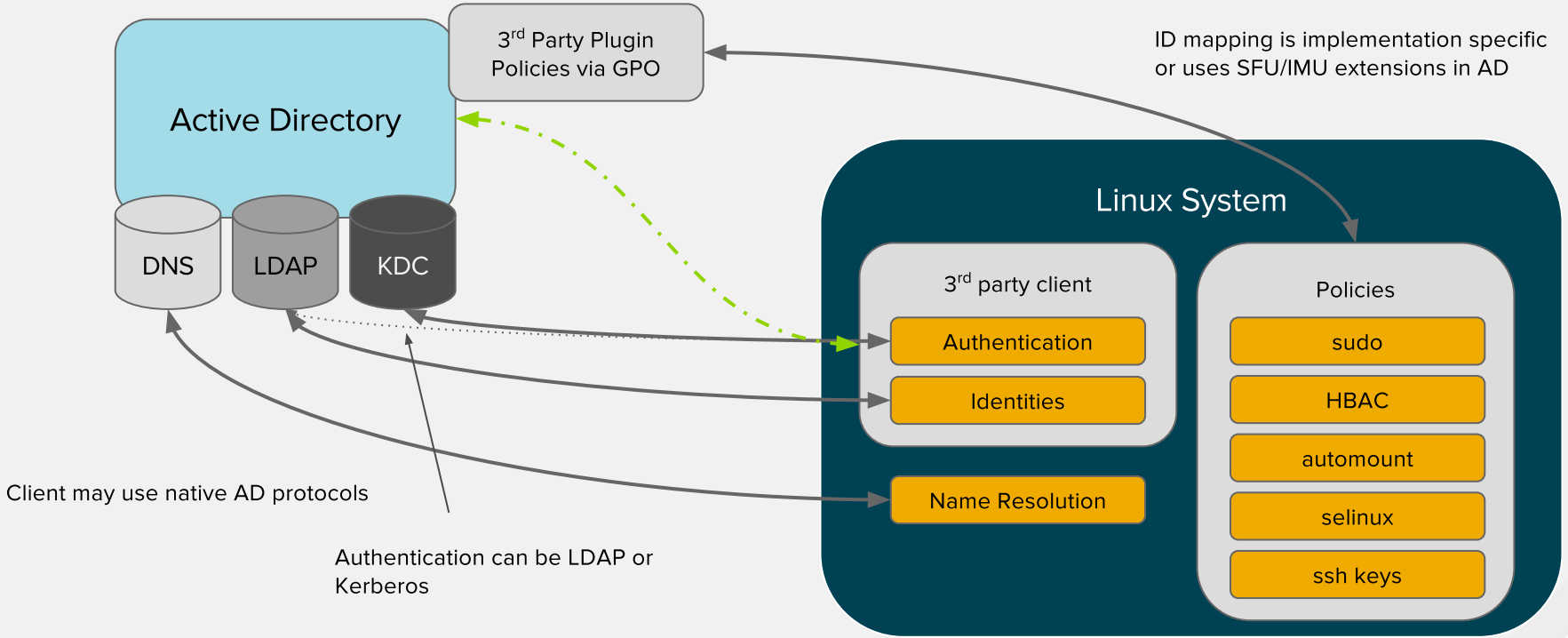
Direct Integration Options

Outline

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

redhat.

Third Party Direct Integration



redhat.

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)

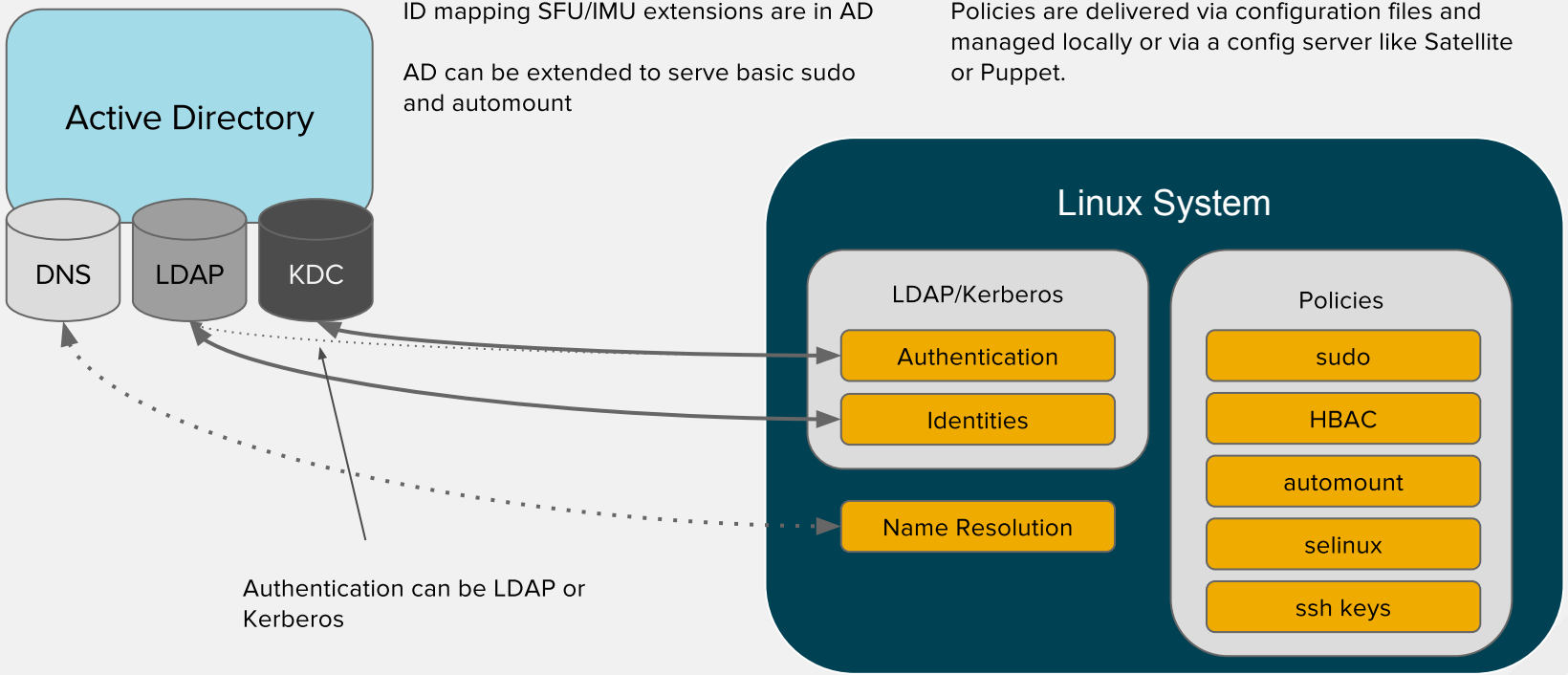
redhat.

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.



redhat.

Legacy Direct Integration

Pros and Cons

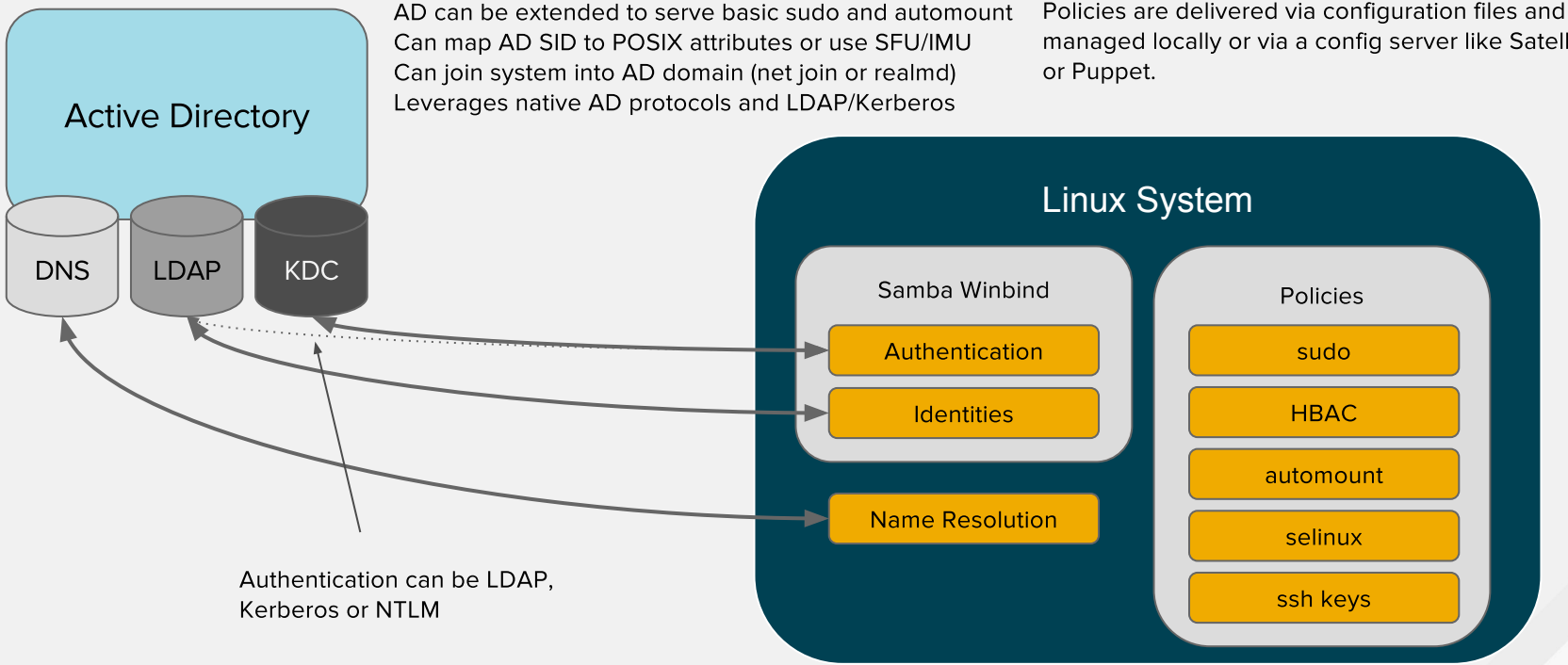
- Pros:
  - Free
  - No third party vendor is needed
  - Intuitive
  - LDAP OTP authentication in Azure (have not tried)
  - Available on UNIXes
- Cons:
  - Requires SFU/IMU AD extension (which are deprecated as of fall 2014)
  - Policies are not centrally managed
  - Hard to configure securely
  - No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

redhat.

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.



redhat.

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

redhat.

SSSD

Introduction

- SSSD = System Security Services Daemon
- SSSD is a service used to retrieve information from a central identity management system.
- SSSD connects a Linux system to a central identity store:
  - Active Directory
  - FreeIPA
  - Any other directory server
- Provides authentication and access control
- Top technology in the evolution chain of the client side IdM components

redhat.

SSSD

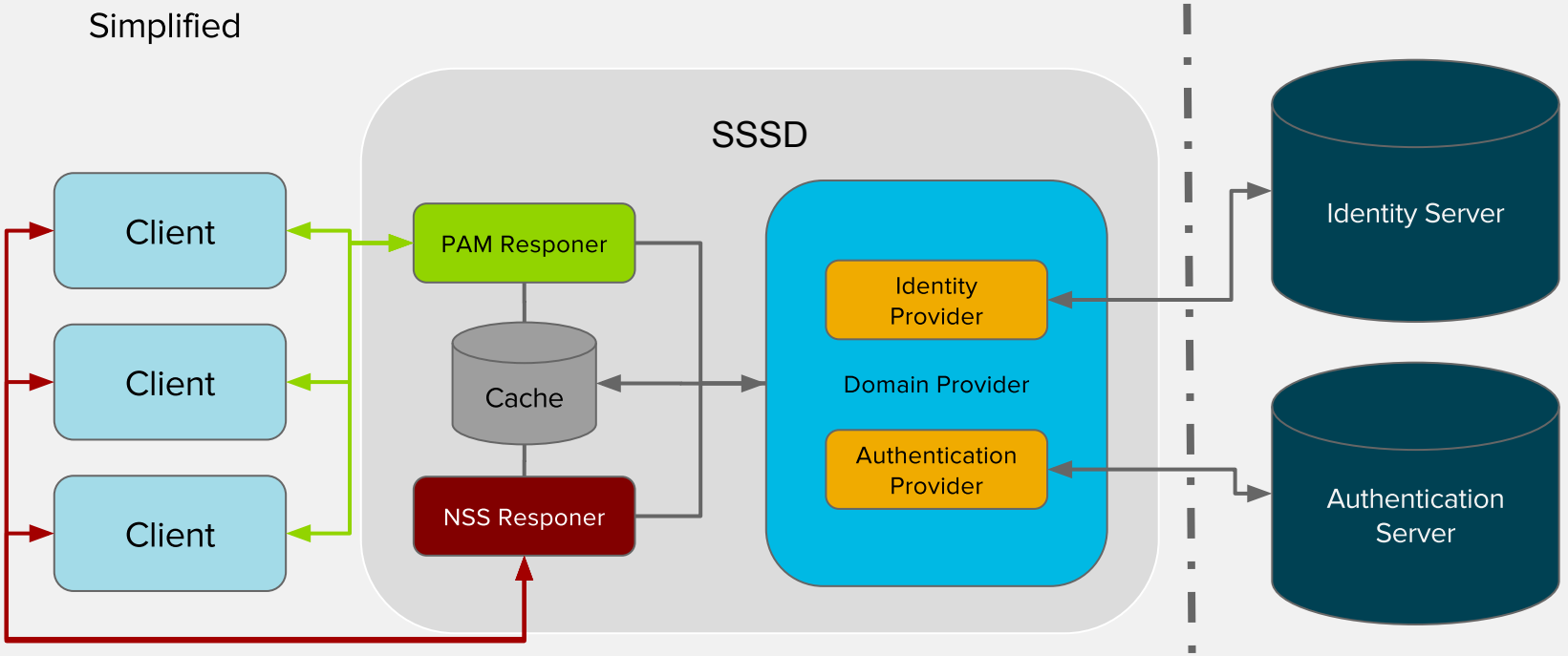
Capabilities

- Multiple parallel sources of identity and authentication – domains
- All information is cached locally for offline use
  - Remote data center use case
  - Laptop or branch office system use case
- Advanced features for:
  - FreeIPA integration
  - AD integration

redhat.

SSSD Architecture

Simplified



redhat.

SSSD

Why?

- Supports everything that previous UNIX solutions support and more
- Brings architecture to the next level
- Supports multiple sources – domains
- Supports IdM specific features
- Supports trusts between AD and IdM
- Has a feature parity with winbind in core areas and surpasses in some

redhat.

Realmd

Couple words

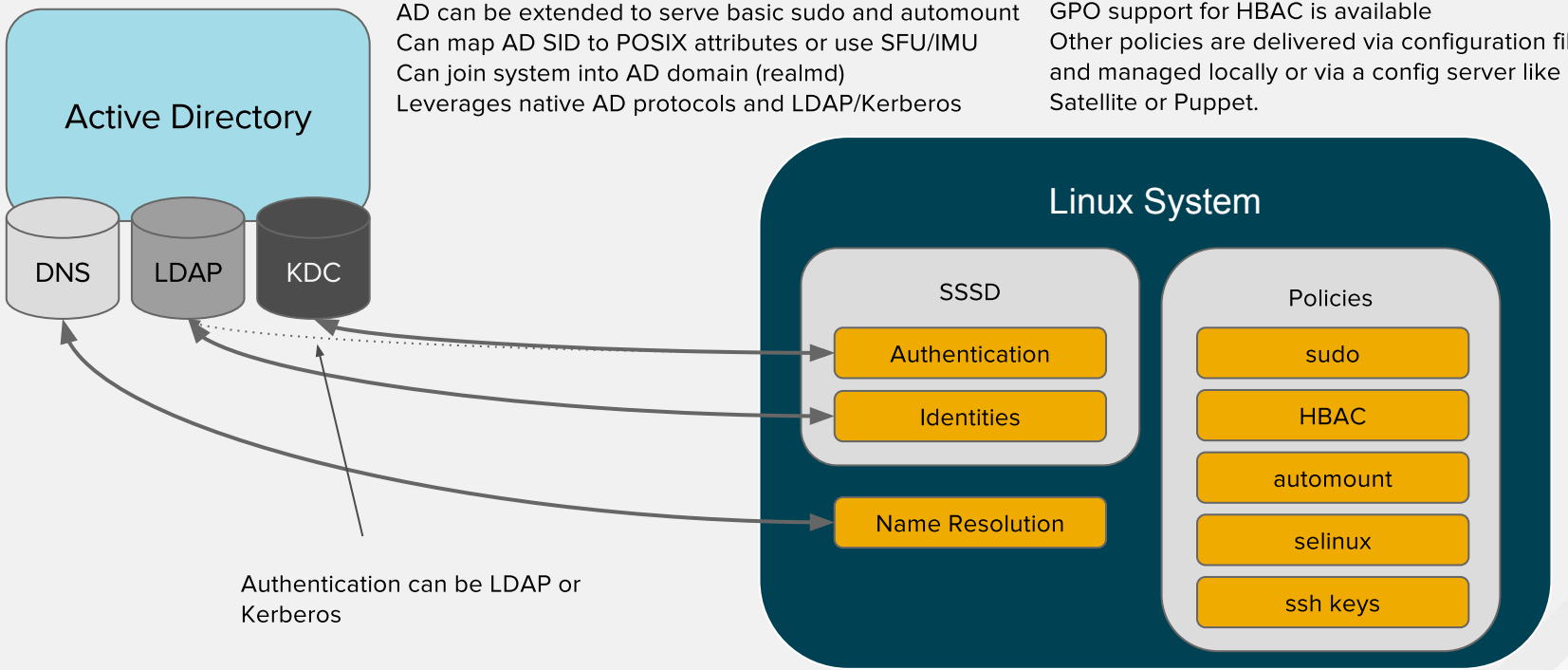
- Component of Linux
- Main goal is to detect domain environment using DNS (detection)
  - AD
  - FreeIPA
  - Kerberos
- Join system to the domain (using SSSD or Winbind)
- Do it in one command or click
- Availability: command line, D-BUS interface, system installer, desktop

redhat.

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.



redhat.

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

redhat.

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/

redhat.

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.


FreeIPA/IdM

FreeIPA/IdM

Introduction

- 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
  - Audit is coming but it is bigger than FreeIPA

redhat.

FreeIPA/IdM

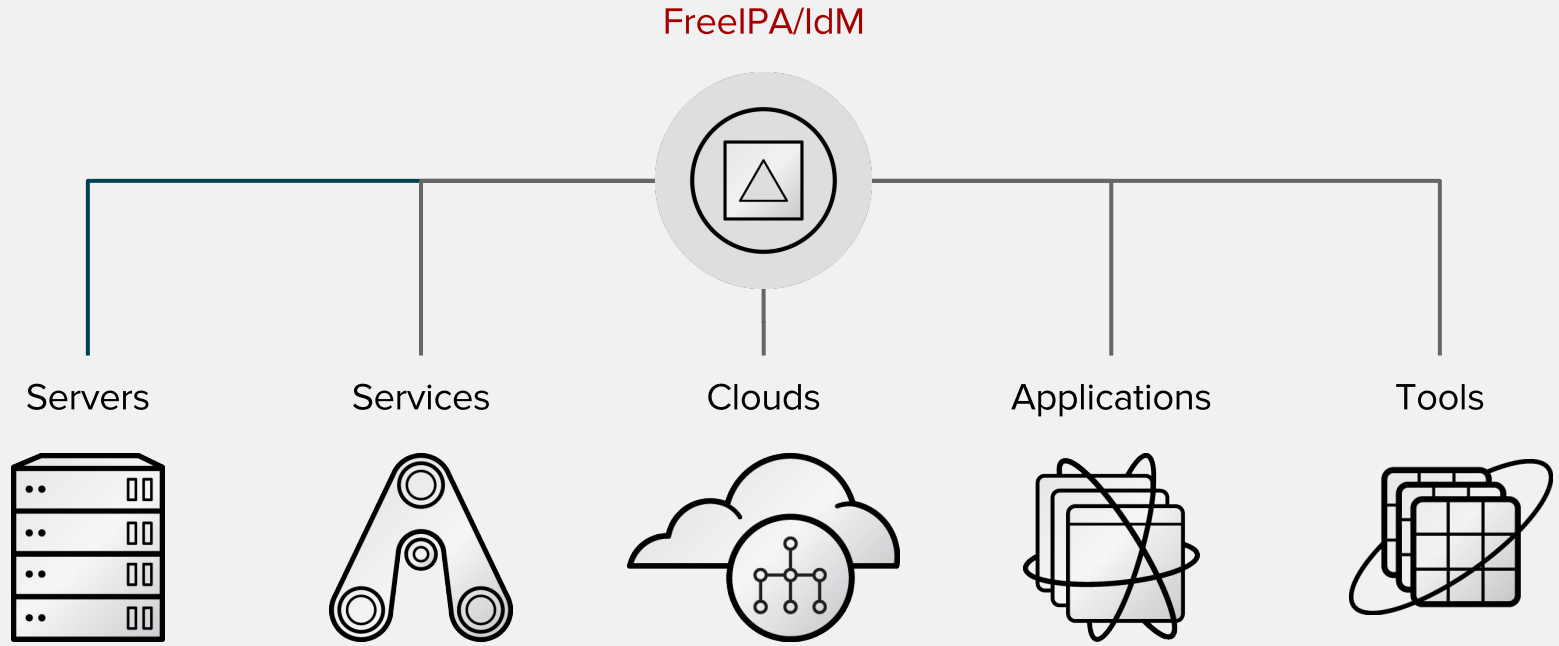
Problems it solves

- Central management of authentication and identities for Linux clients better than stand - alone LDAP/Kerberos/NIS - based solutions
- Acts as a gateway between the Linux infrastructure and AD environment making infrastructure more manageable and more cost effective

redhat.

Modern Identity Model

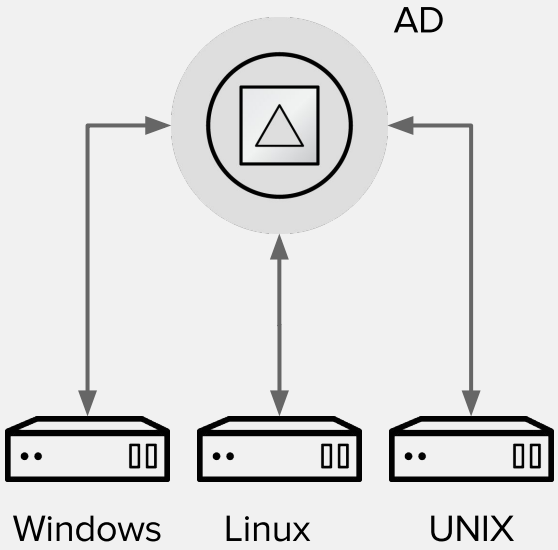
Simplified View

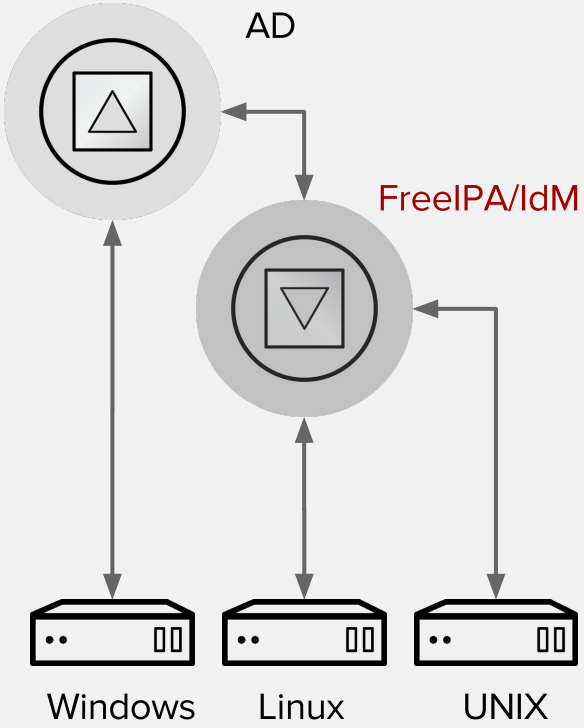


redhat.

Connecting Systems

Integration Options

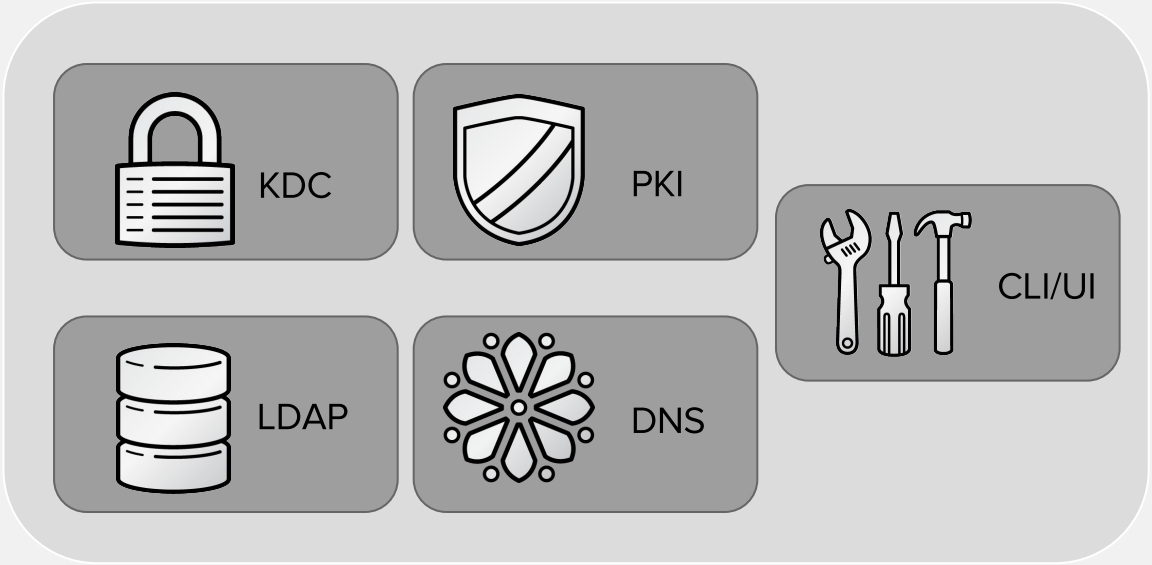




redhat.

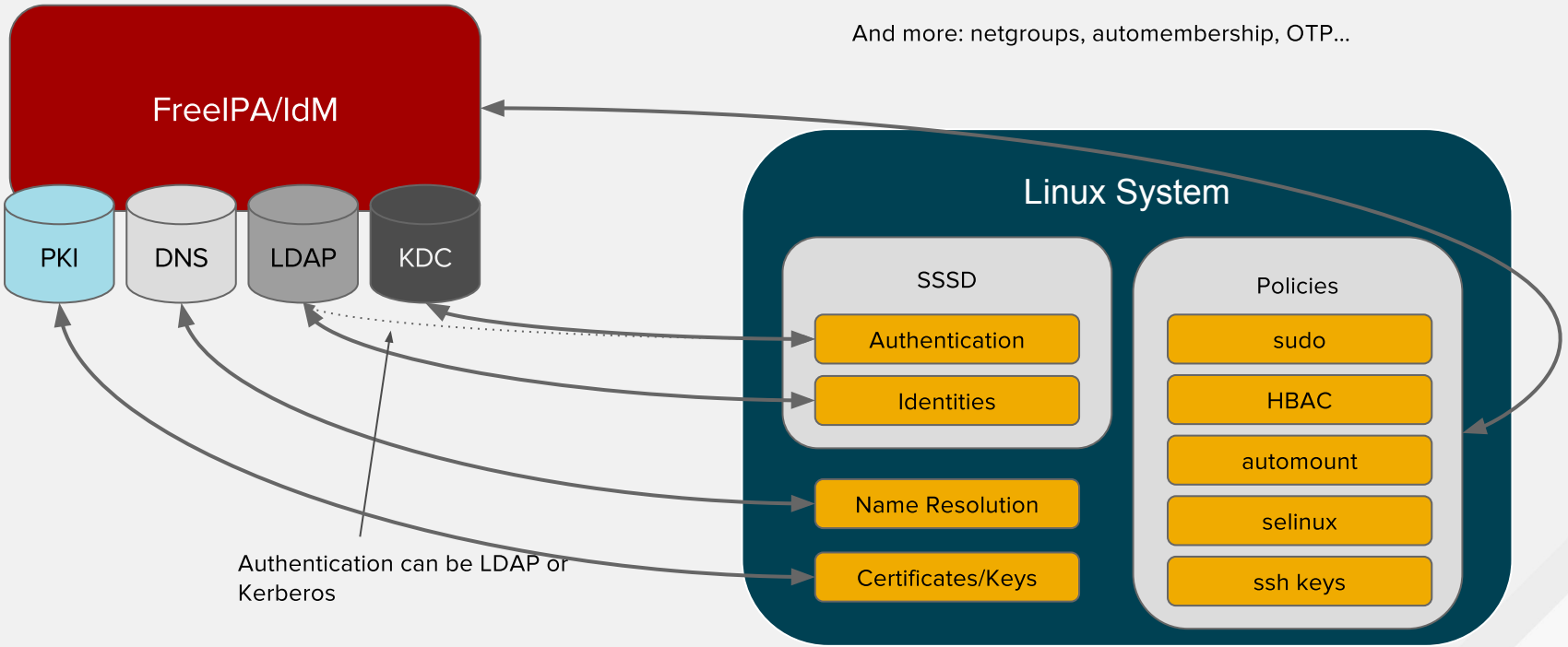
FreeIPA/IdM

High Level Architecture



redhat.

FreeIPA/IdM Integration



redhat.

FreeIPA/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

redhat.

FreeIPA/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

redhat.

FreeIPA/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 (planned as Tech Preview in RHEL 7.2)
- Disadvantages:
  - You need to delegate a zone

redhat.

FreeIPA/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)

redhat.

FreeIPA/IdM

Features PKI

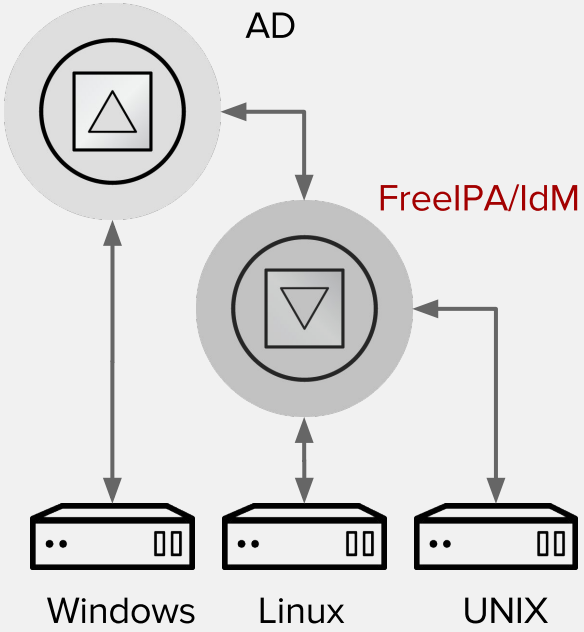
- CA related capabilities
  - Certificate provisioning for users (new), hosts and services
  - Multiple certificate profiles
  - Sub CAs (in works)
- CA deployment types
  - CA-less
  - Chained to other CA
  - Self signed root
- Tool to change deployment type and rotate CA keys
  - Flexibility in deploying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

redhat.

Indirect Integration using FreeIPA/IdM

Indirect Integration

Deep dive



redhat.

Integration Paths

Overview

- User and password synchronization (not recommended)
- Cross forest trusts (recommended)

redhat.

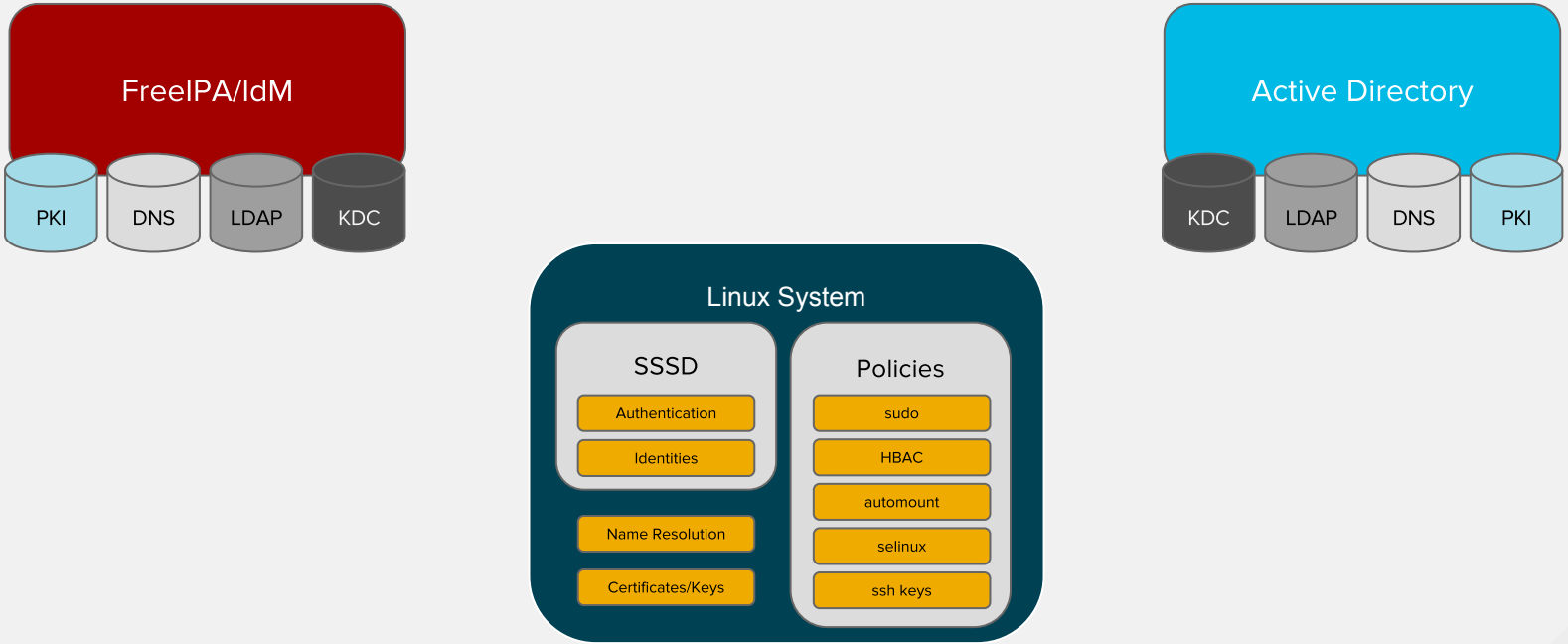
Synchronization Solution

Overview

- LDAP level synchronization
- AD is the authoritative source - one way sync
- No group synchronization, only users
- Only one domain can be synchronized
- Single point of failure - sync happens only on one replica
- Limited set of attributes is replicated
- Passwords need to captured and synced
    - Requires a plugin on every AD DC
    - Mismatch of password policies can lead to strange errors

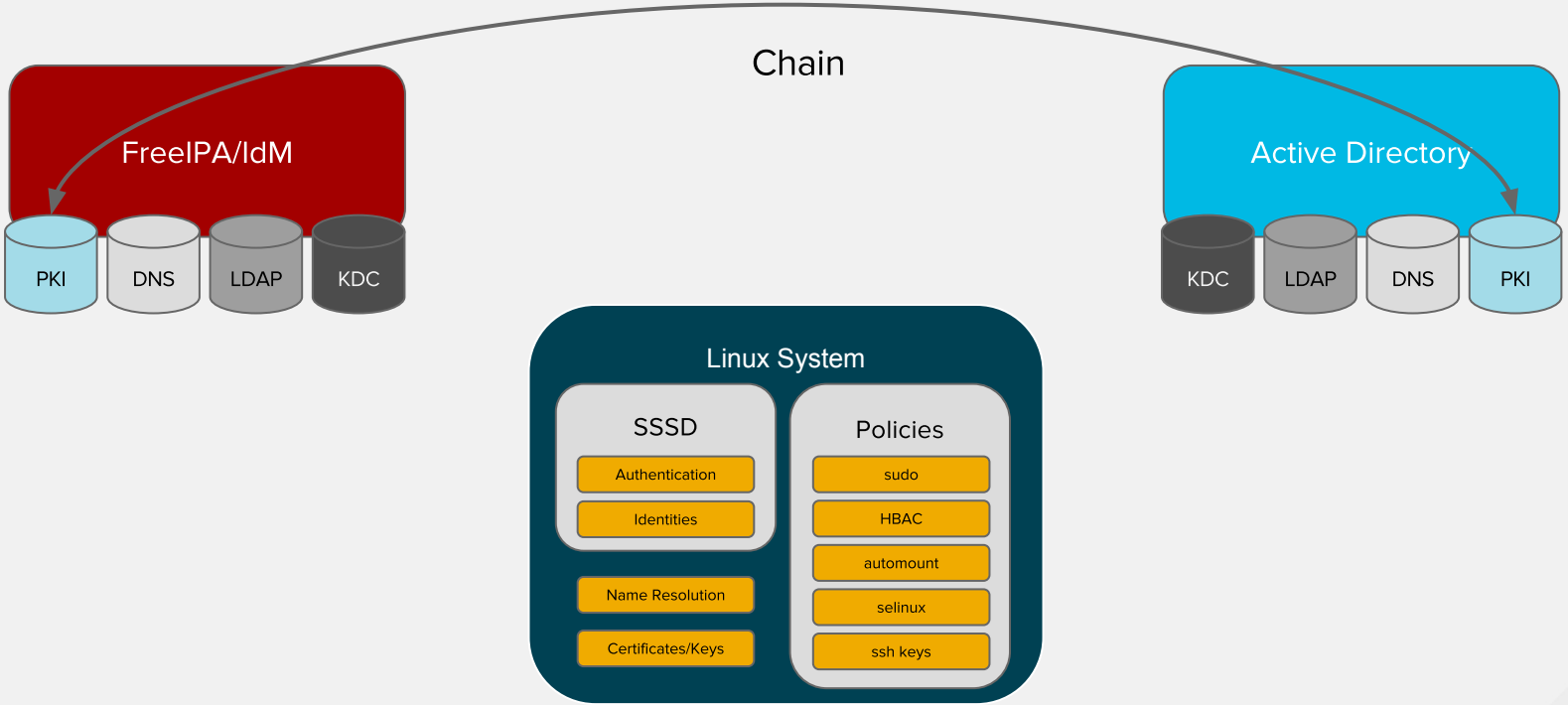
redhat.

FreeIPA/IdM AD Integration with Trust



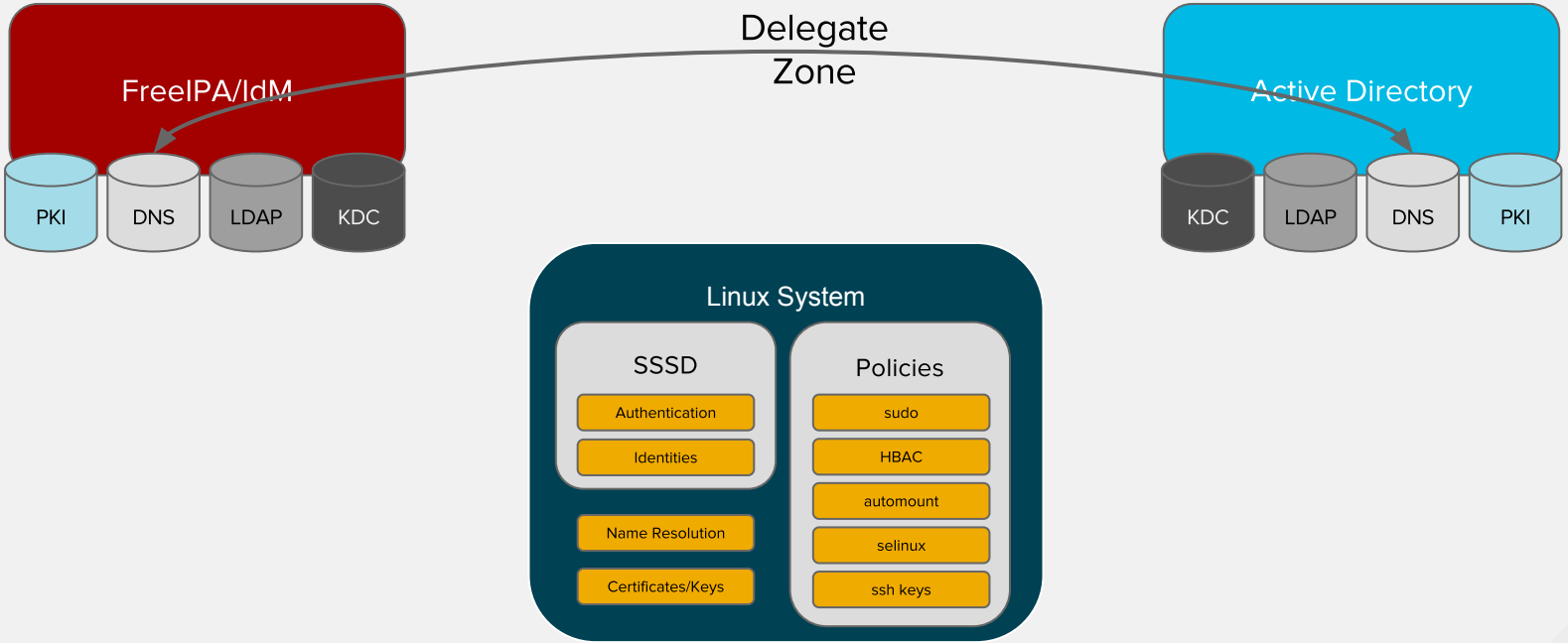
redhat.

FreeIPA/IdM AD Integration with Trust



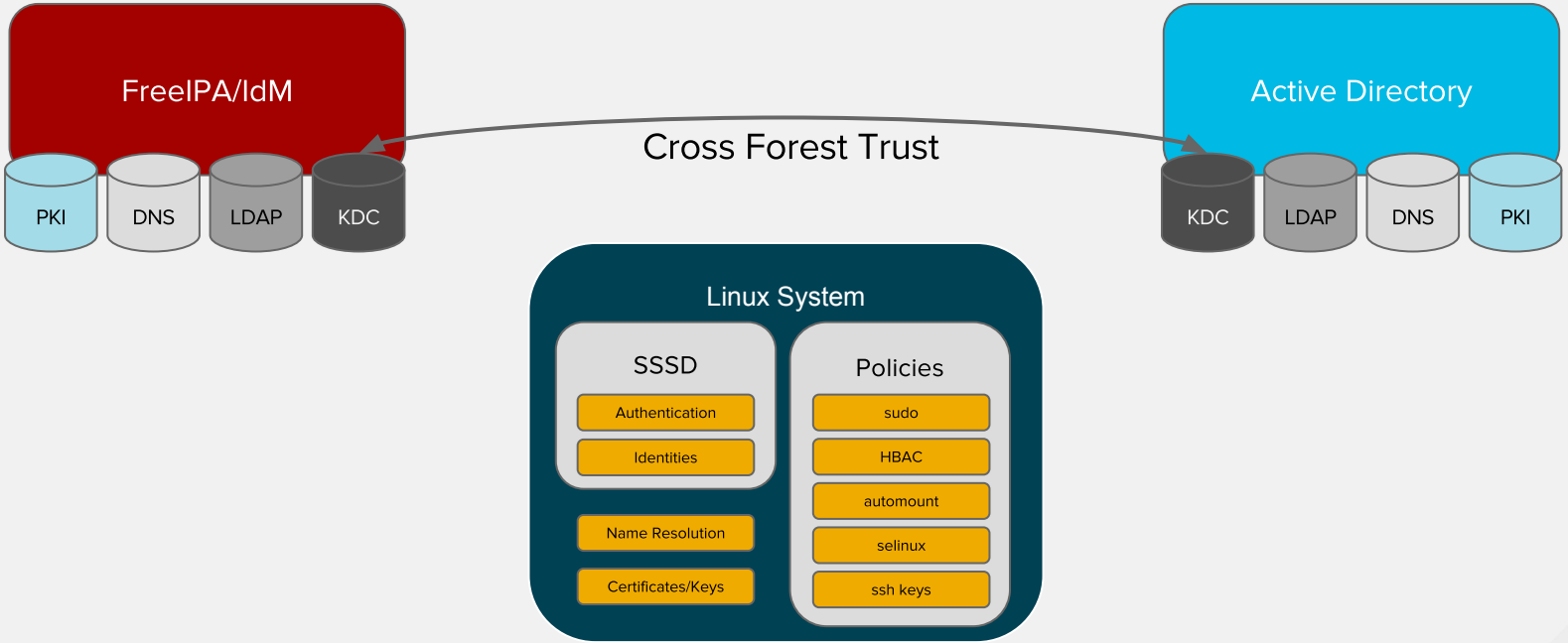
redhat.

FreeIPA/IdM AD Integration with Trust



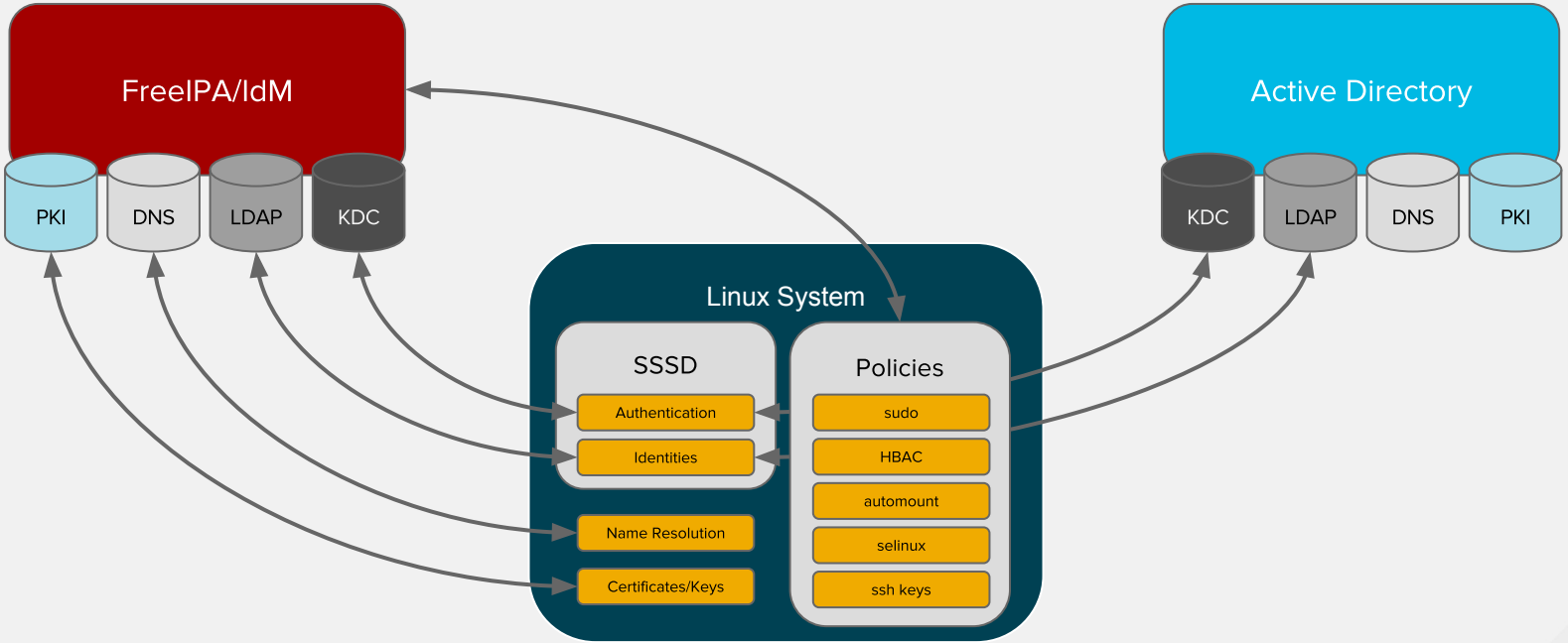
redhat.

FreeIPA/IdM AD Integration with Trust



redhat.

FreeIPA/IdM AD Integration with Trust



redhat.

Trust Based Solution

Pros and Cons

- Pros:
  - Reduces cost – no CALs or 3rd party
  - Policies are centrally managed
  - Gives control to Linux admins
  - Enabled independent growth of the Linux environment
  - No synchronization required
  - Authentication happens in AD
- Requirement:
  - Proper DNS setup

redhat.

User Mapping

Details

- Can leverage SFU/IMU for POSIX (brown field)
- Can do dynamic mapping of the SIDs to UIDs & GIDs (green field)
- Static override with ID views

redhat.

Trust

Details

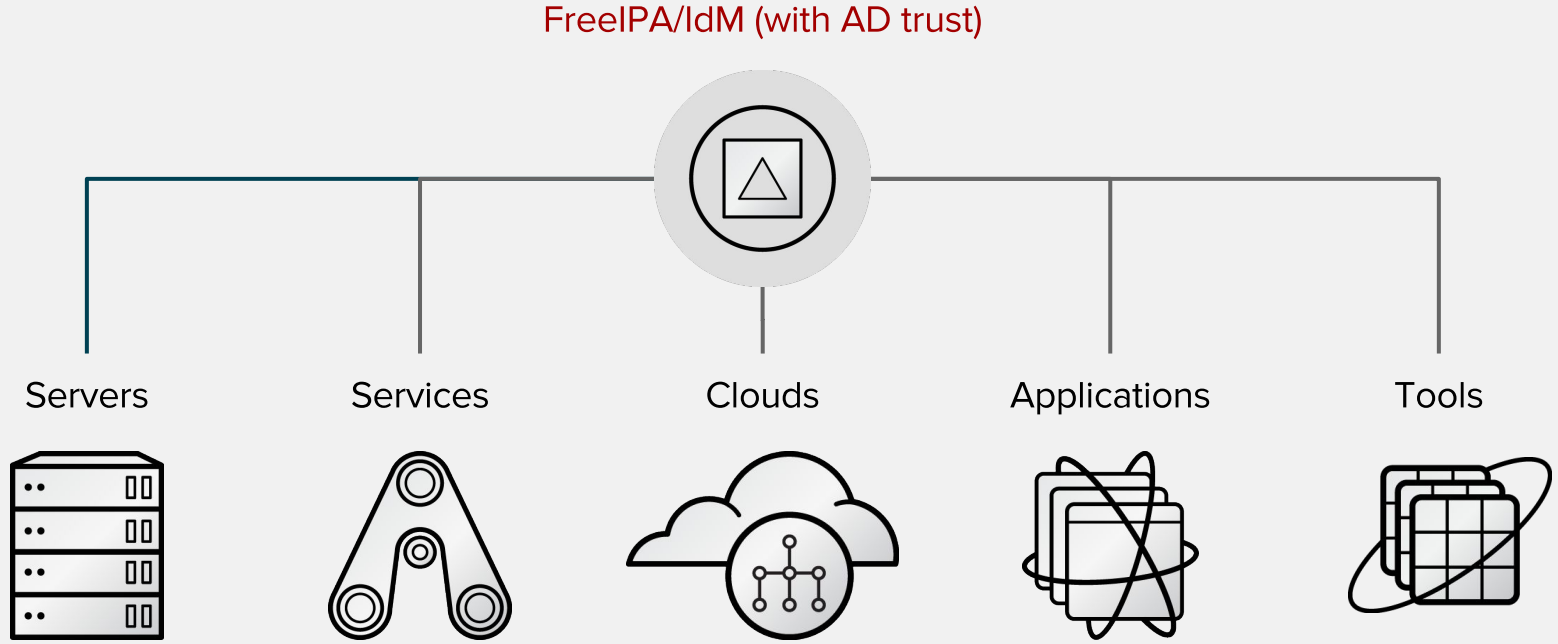
- Two-way and one-way trust (FreeIPA trusts AD)
  - AD/Samba DC trusting FreeIPA is on the roadmap
- Trust agents (different behavior of different replicas)
- Migration from the sync to trust

redhat.


Application Integration
redhat.

Modern Identity Model

Simplified View



redhat.

Application Integration

Current situation

- Applications are usually integrated using LDAP
- This approach has challenges:
  - Types of users
  - User sources
  - Connections to those sources
  - Deployment modes

redhat.

Types of Users

Overview

- End users:
  - Users coming from the internet (customers)
  - Users that are a part of the enterprise (internal customers)
  - Contractors, partners, providers, suppliers...
- Power users:
  - Enterprise sysadmins
  - Service provider
  - IT services subcontractor

redhat.

User Sources

Overview

- On Premise:
  - Users/admins come from:
    - Domain controllers (LDAP + Kerberos + more):
      - Active Directory, FreeIPA/IdM
    - LDAP directories:
      - 389, OpenLDAP, ApacheDS, SunDS, Oracle OID...
    - Databases
      - SQL, Non-SQL
    - Federation
      - SAML IdP, OpenID, OpenID Connect

redhat.

User Sources

Overview continued

- Managed Service (Cloud):
  - End users and some power users - from customer sources
  - Power users (those who manage tenants) - from managed service provider

redhat.

Connections to Sources

Overview

- Security of connections
  - Identities, passwords, keys, certs
- Multiples of identity sources in the same environment
  - Different LDAPs, Domains, Forests
- Failover
  - Each directory consists of multiple servers
- Offline situation (connection to all servers lost)

redhat.

Deployment Modes

Overview

- Development – simple, no central server
- Demo – emulated users with roles, single box
- POC – using a dummy directory
- Production – all sorts of different identity sources as mentioned above

redhat.

Complexity Matrix

Welcome to identity management nightmare!

- Evolution of the application leaves a lot of cruft that is hard to maintain or clean
- Different modes of operation dictate different identity sources
- Different production use cases and requirements create a lot of complexity
- Add compliance requirements and audits…
- One has to be an expert to sort all this out…

redhat.

What if?

- Can we offload all this complexity somewhere?

redhat.

What if?

- Can we offload all this complexity somewhere?
- How about externalizing the authentication, some access control and collecting identity information for the user?

redhat.

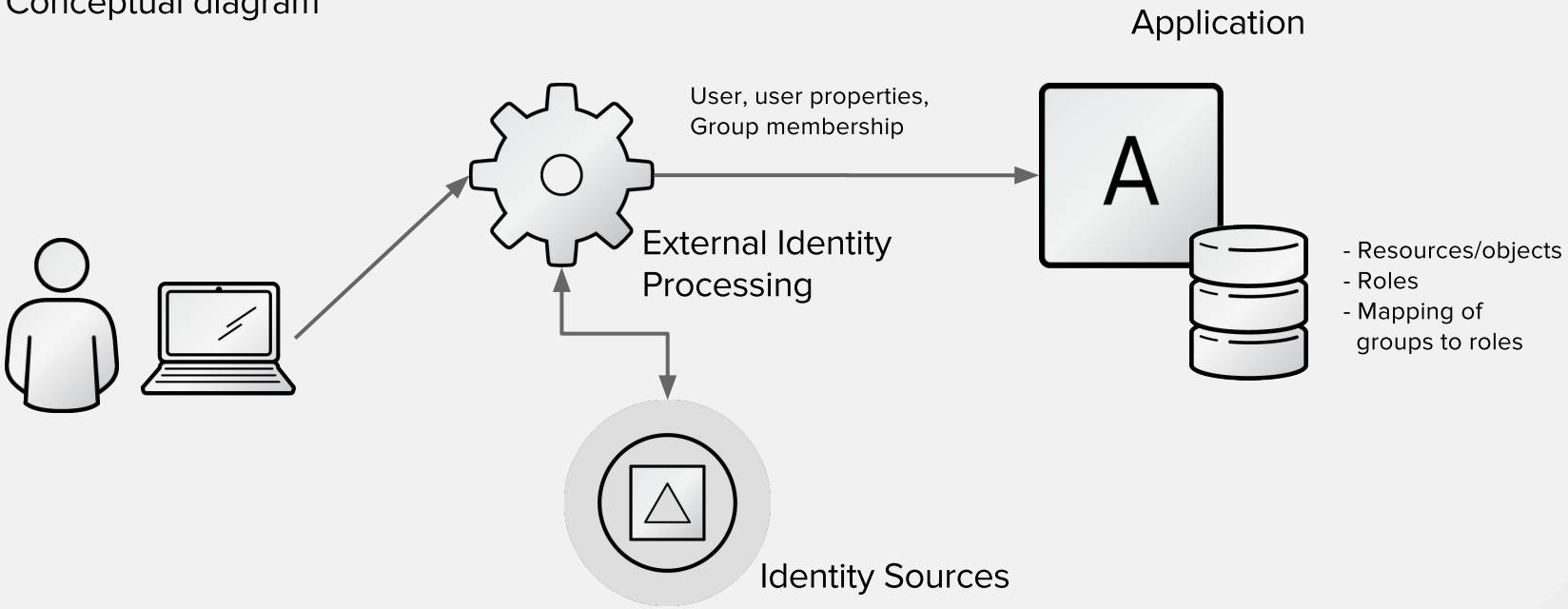
What belongs to application and what not?

- Application manages resources and defines roles. Those belong to the application
- Users with properties and group membership come from external sources
- Mapping user groups and properties to roles is the responsibility of the application
- This mapping is independent of the source the user came from

redhat.

Architecture

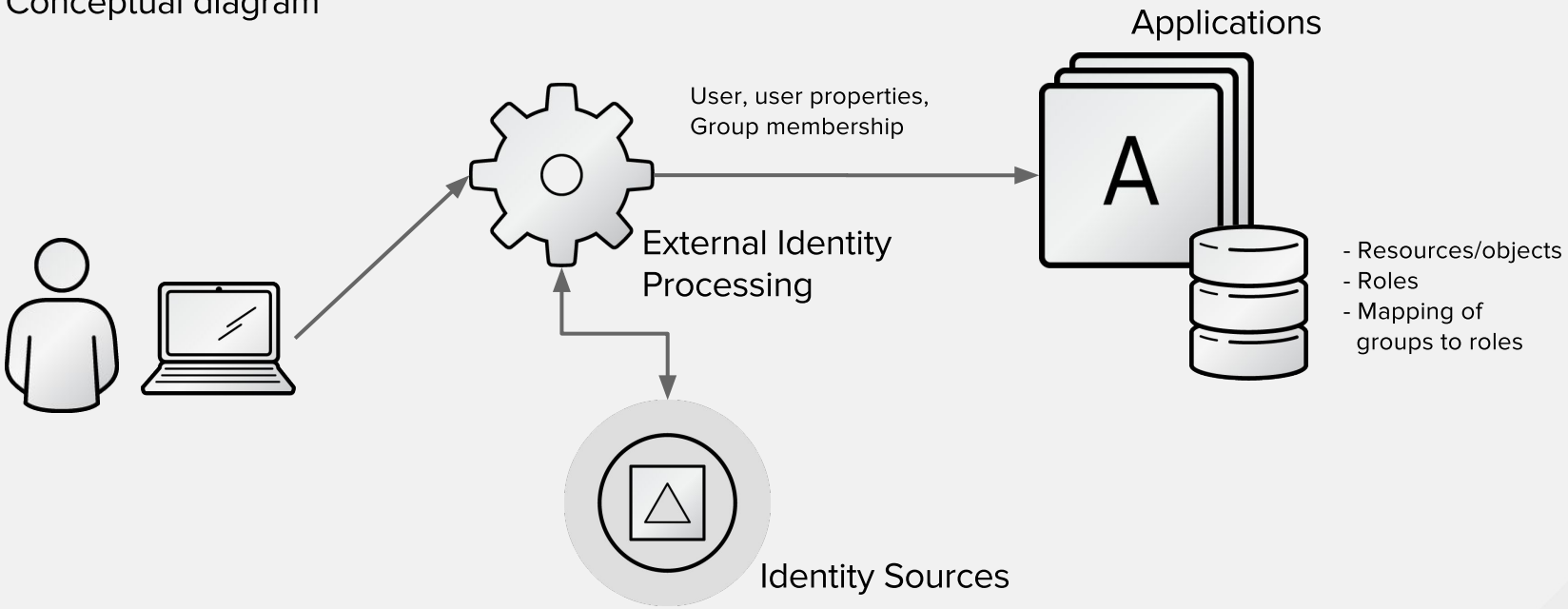
Conceptual diagram



redhat.

Architecture

Conceptual diagram



redhat.

What can be externalized?

Authentication

- AD, FreeIPA (IdM), AD/IdM Trusts, LDAP
- Kerberos GSSAPI
- OTP 2FA
- Certificates
- SAML, OpenID, OpenID connect, Persona

redhat.

What can be externalized?

Authorization

- You need to have authorization added in case of:
  - Kerberos GSSAPI case
  - Certificate based authentication

redhat.

What can be externalized?

Identity lookup

- Traditional:
  - AD
  - FreeIPA/IdM
  - AD/IdM Trusts
  - LDAP
- Sometimes no identity lookups possible
  - Assertion has all the data

redhat.

Apache

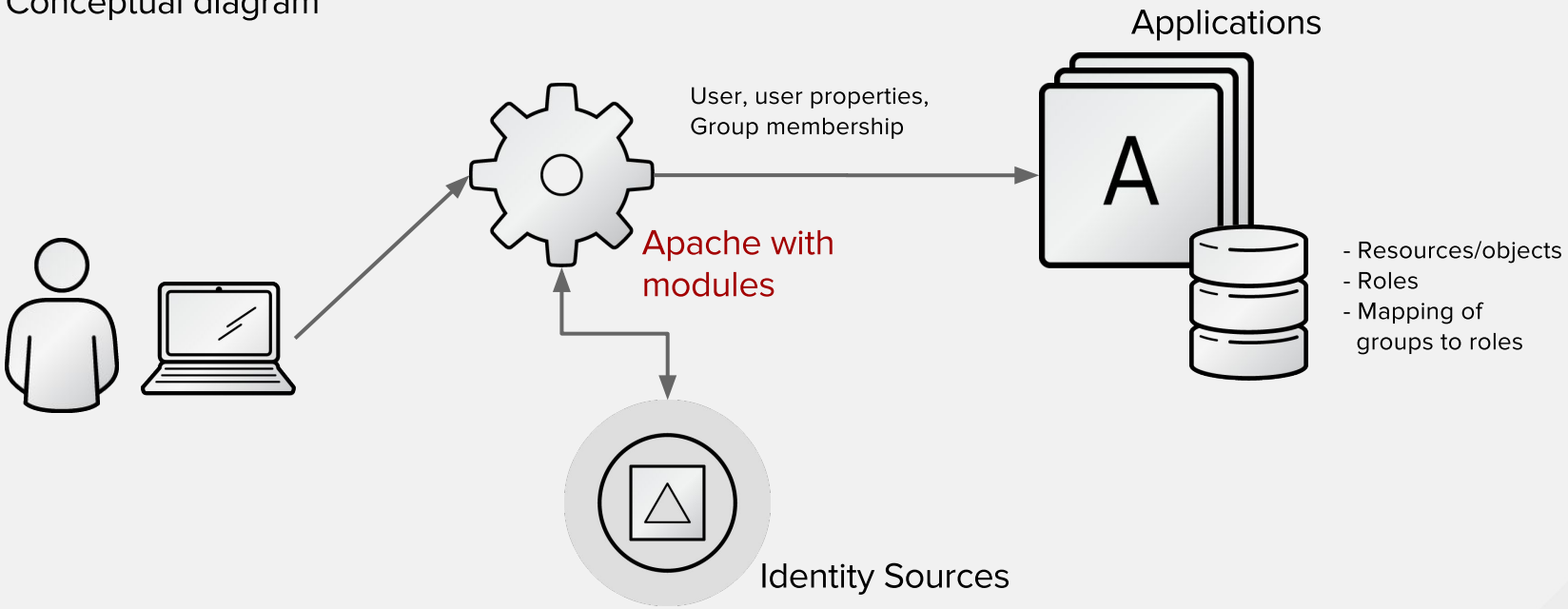
Why?

- Apache is a well established web server
- Available on many platforms
- Has a very good pluggable architecture

redhat.

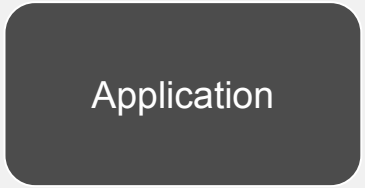
Architecture

Conceptual diagram



redhat.

Architecture

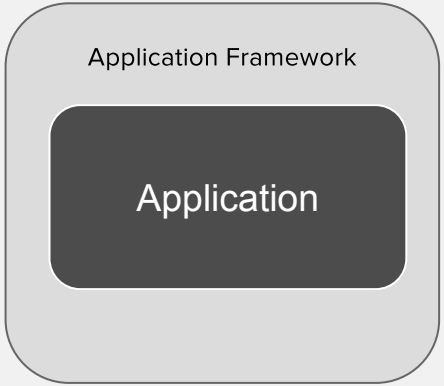


redhat.

Architecture

redhat.

Architecture

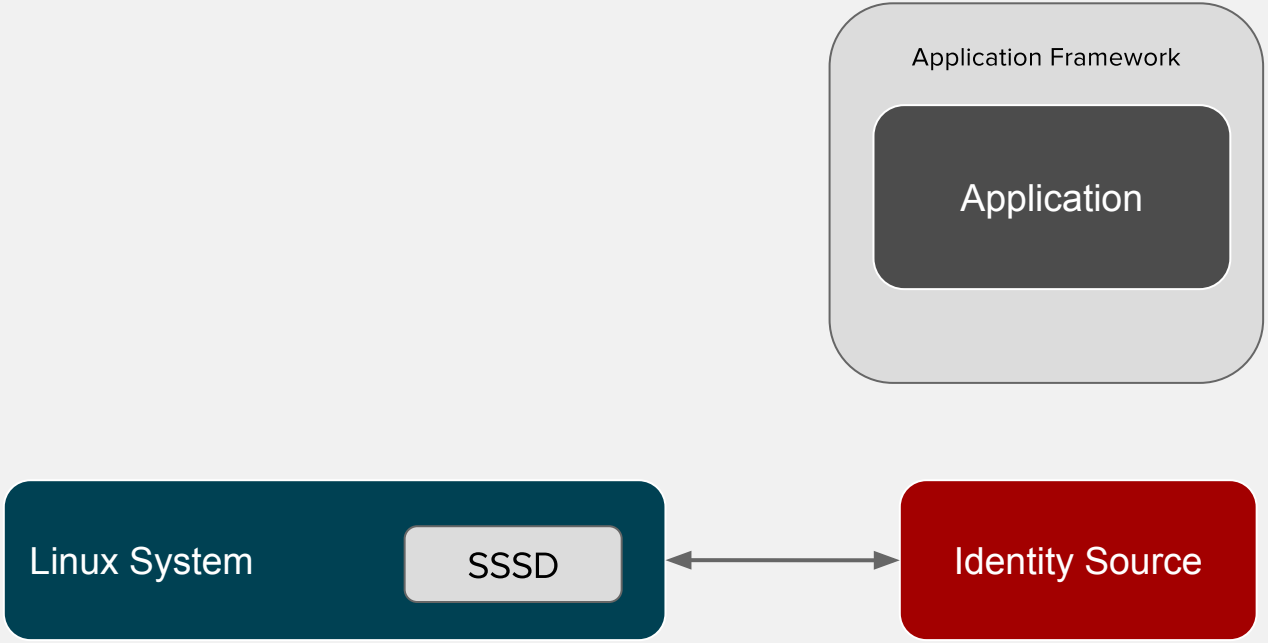


redhat.

Architecture

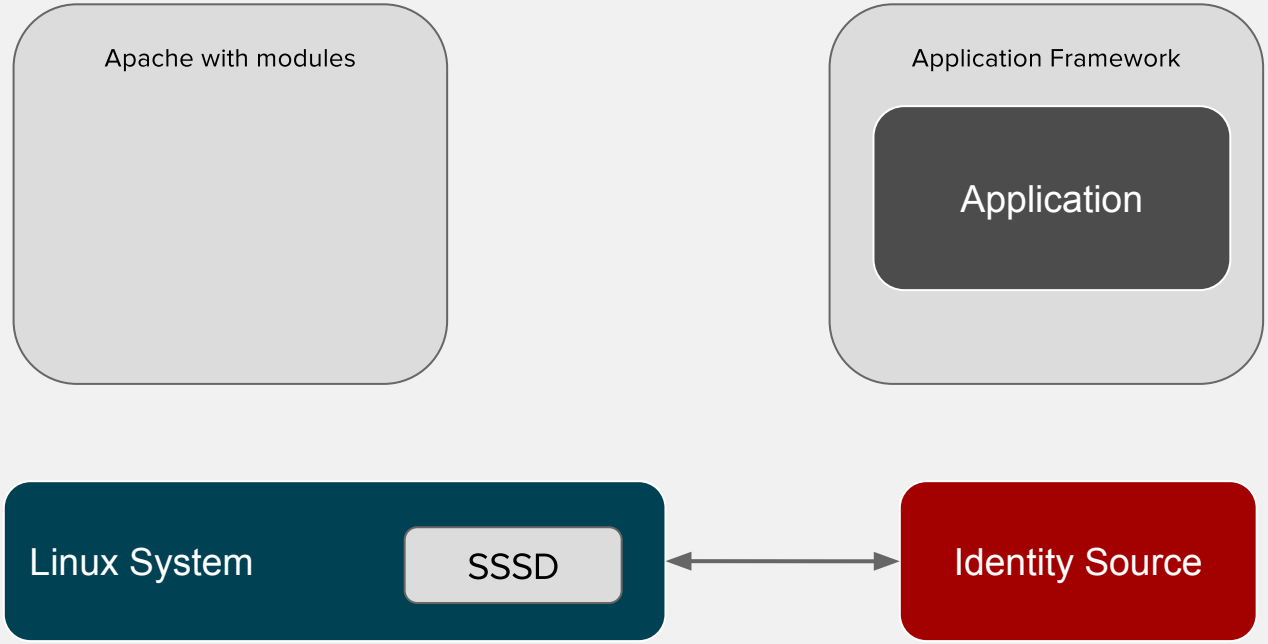
redhat.

Architecture



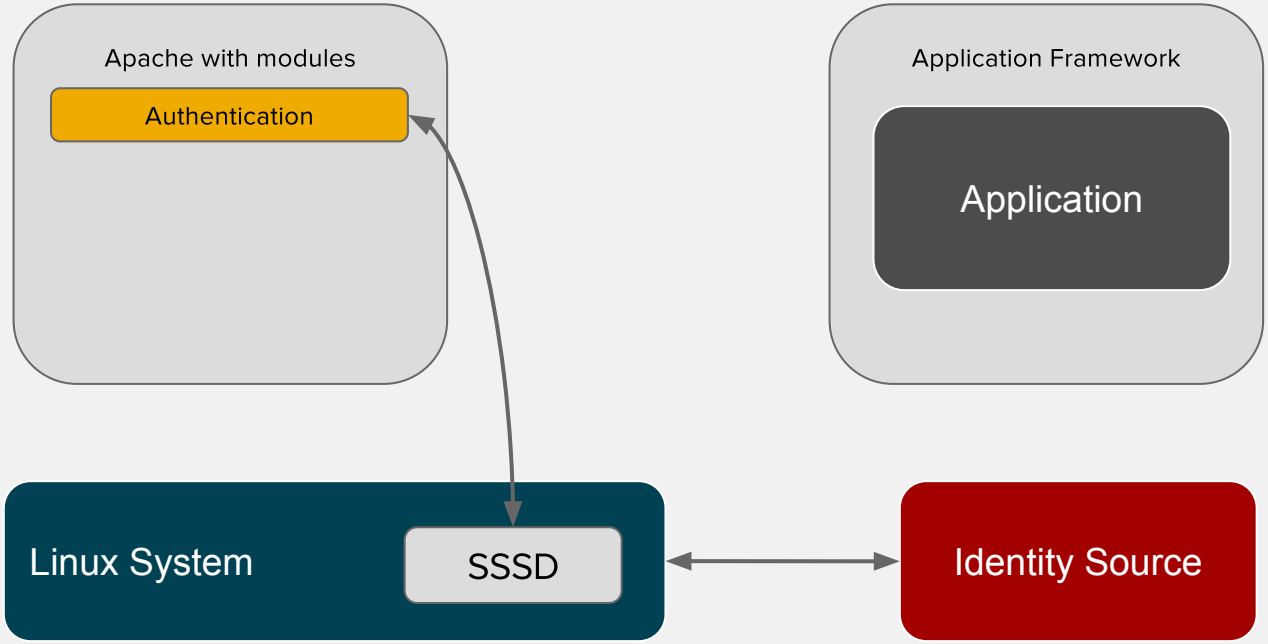
redhat.

Architecture



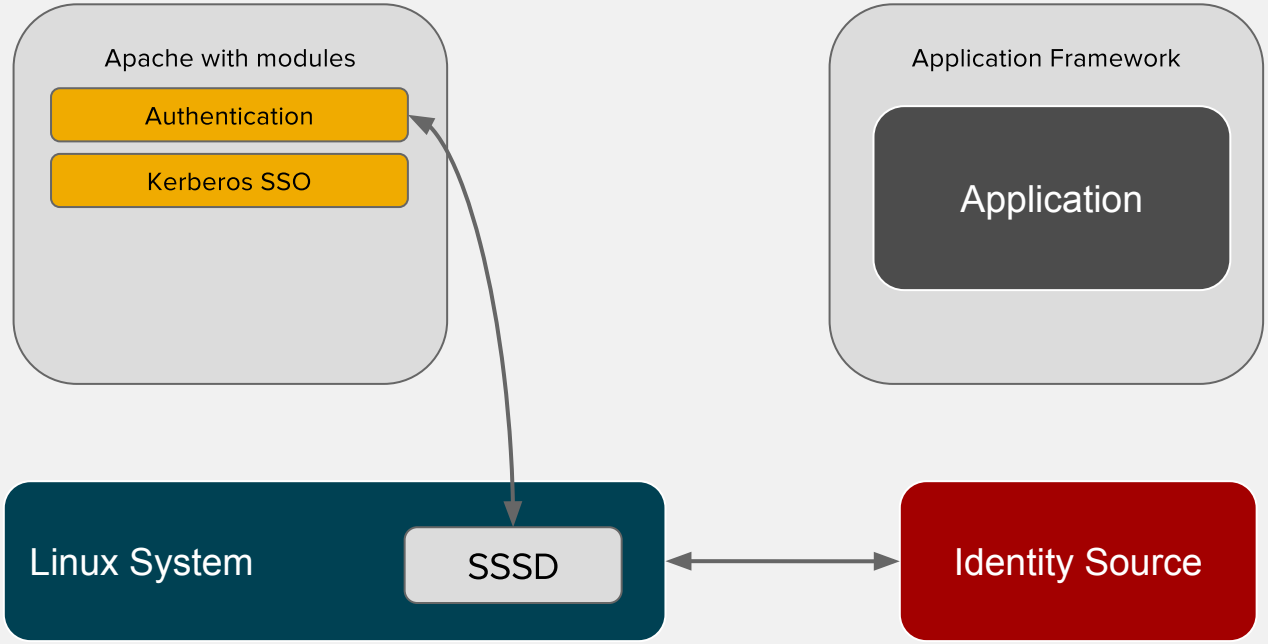
redhat.

Architecture



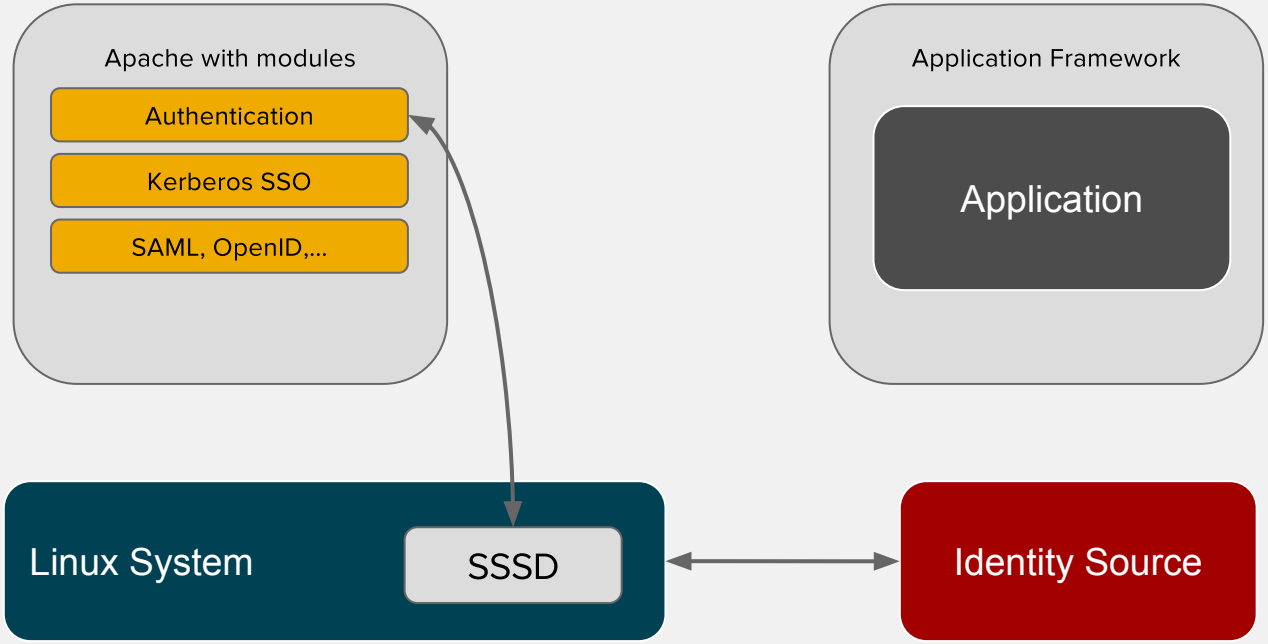
redhat.

Architecture



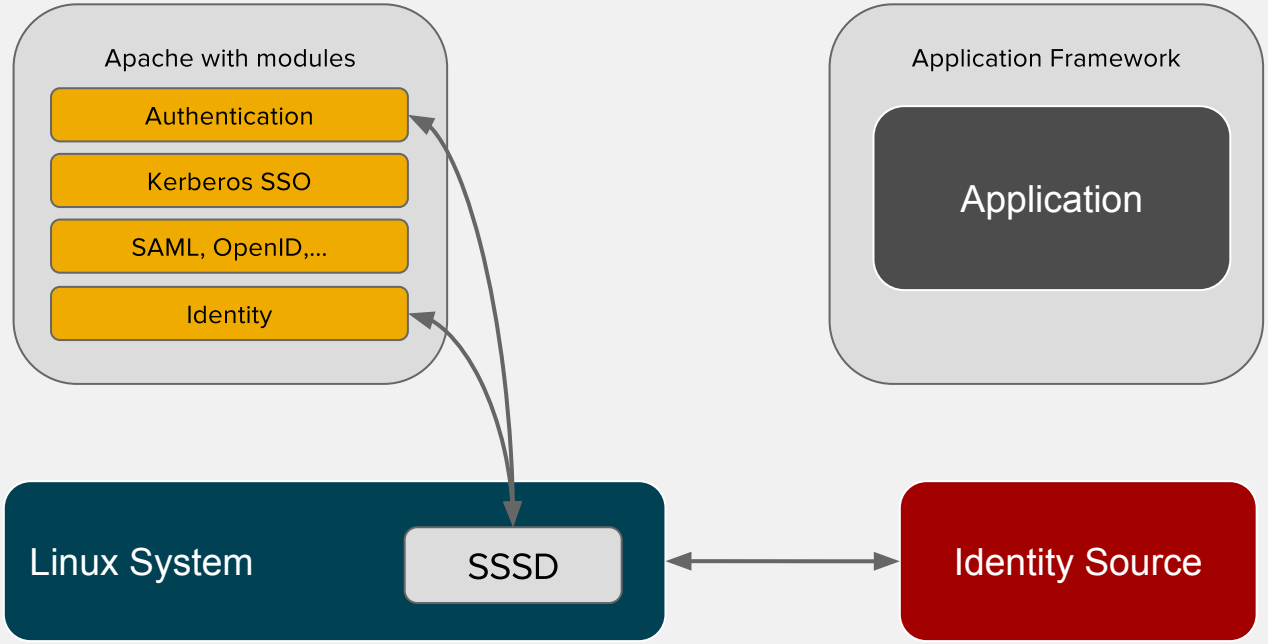
redhat.

Architecture



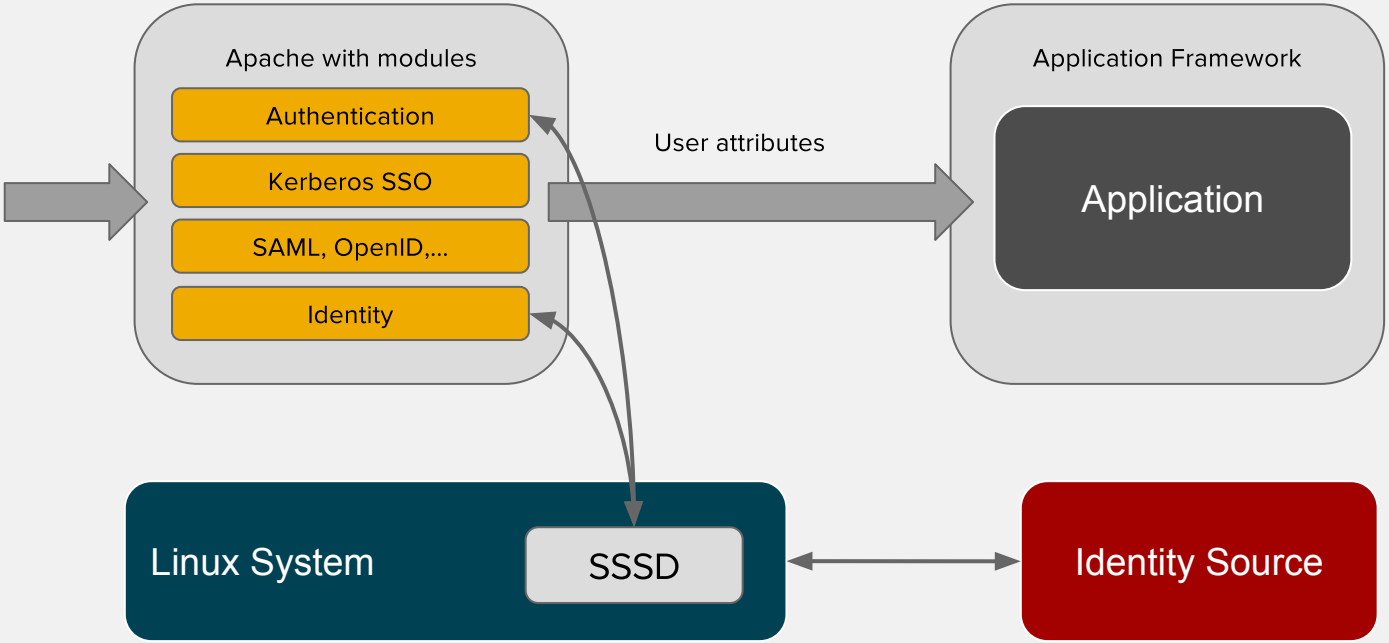
redhat.

Architecture



redhat.

Architecture



redhat.

Flow

Overview

- Request hits a URL
- Modules intercept the request
- Authenticate
- Do access control check if needed
- Fetch related information
- Pass it to the application via environment variables
- Application reads the variables and uses them

redhat.

Modules

Overview

<table>
<tr><th>Authentication Method</th><th>Authentication</th><th>Access check</th><th>Extra user info</th></tr>
<tr><td>Kerberos</td><td>mod_auth_gssapi
(mod_auth_kerb)</td><td rowspan="3">mod_authnz_pam</td><td rowspan="4">mod_lookup_idenity</td></tr>
<tr><td rowspan="2">Certificate</td><td>mod_nss</td></tr>
<tr><td>mos_ssl</td></tr>
<tr><td>Formas based</td><td colspan="2">mod_intercept_form_submit</td></tr>
<tr><td>SAML</td><td colspan="3">mod_auth_mellon</td></tr>
<tr><td>OpenID Connect</td><td colspan="3">mod_auth_openidc*</td></tr>
</table>

redhat.

Open Questions

- What about NGNIX?
  - We are working with them
- What about administrative use case?
  - SSSD has built a D-BUS interface to list users
  - Sometimes listing users is not possible so application should be ready

redhat.

Benefits

Summary

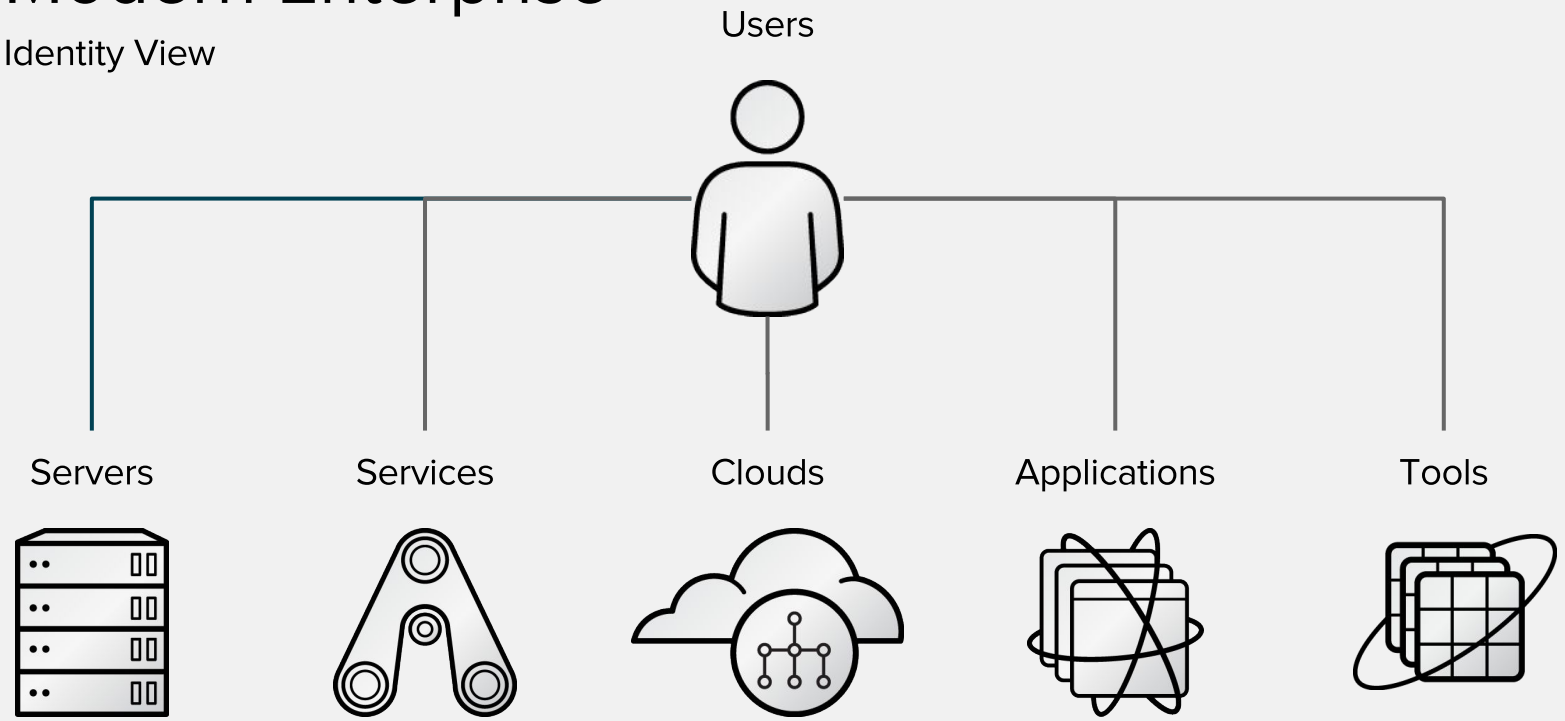
- All complexity is removed from the application
- Easy to use application in all required modes
- Enables use of the application in multiple deployment scenarios without modification
- Faster development and delivery
- Optional and flexible
- **NO MORE DIRECT LDAP CONNECTION**

redhat.

Wrap-up

Modern Enterprise

Identity View



redhat.

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

redhat.

Questions?

Finally

redhat.

redhat.

THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos