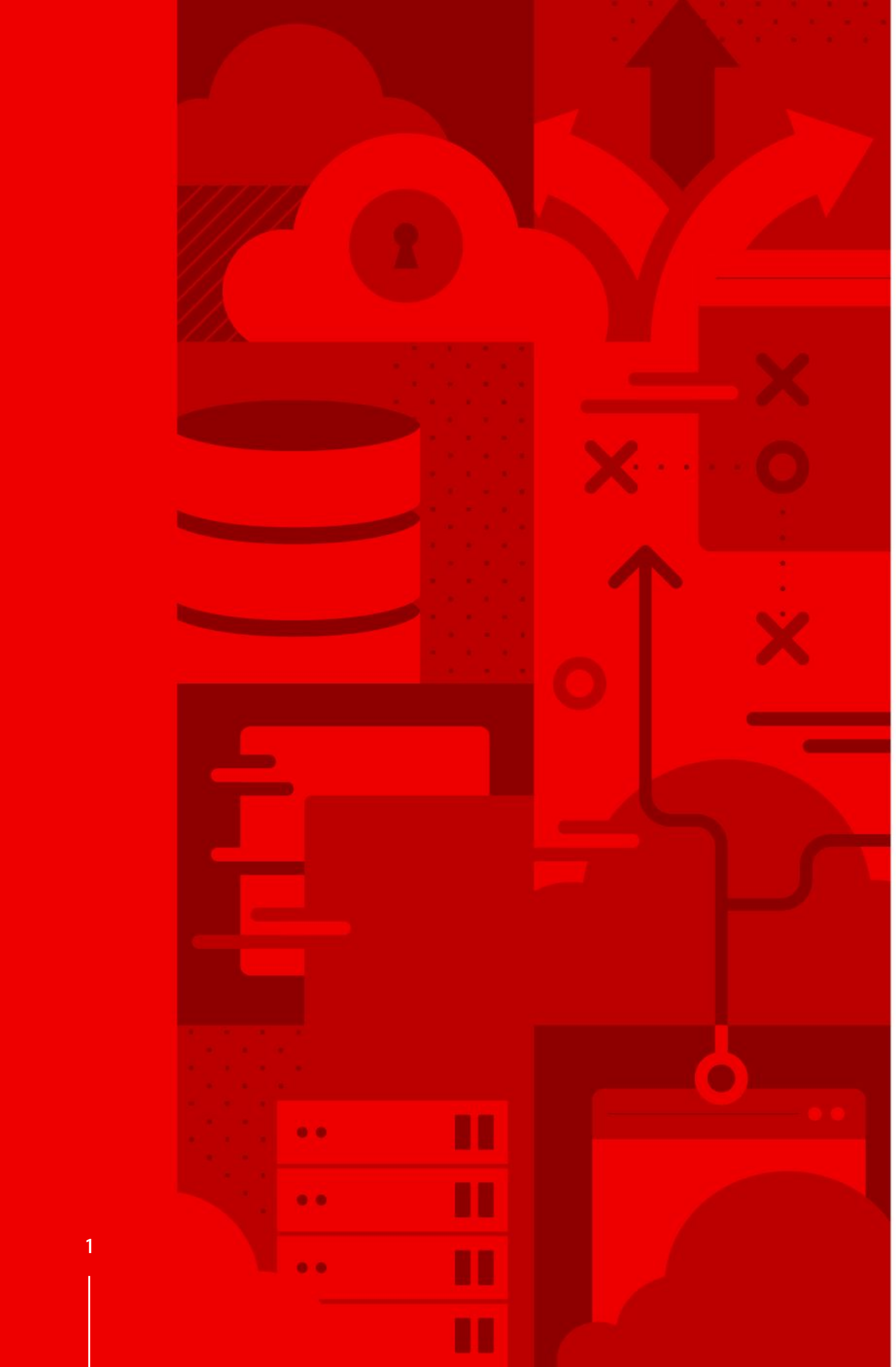


Identity Management in Red Hat Enterprise Linux

Introduction to Authentication and Authorization, IdM and Active Directory Integration

Alfred Bach
Principal Solution Architect
Cloud and Infrastructure

Red Hat

High-Level View

Modern Enterprise

Infrastructure View

Servers / Infrastructure

Windows

Linux

UNIX

Services

Internal and External

Clouds

Private and Public

Apps & Tools

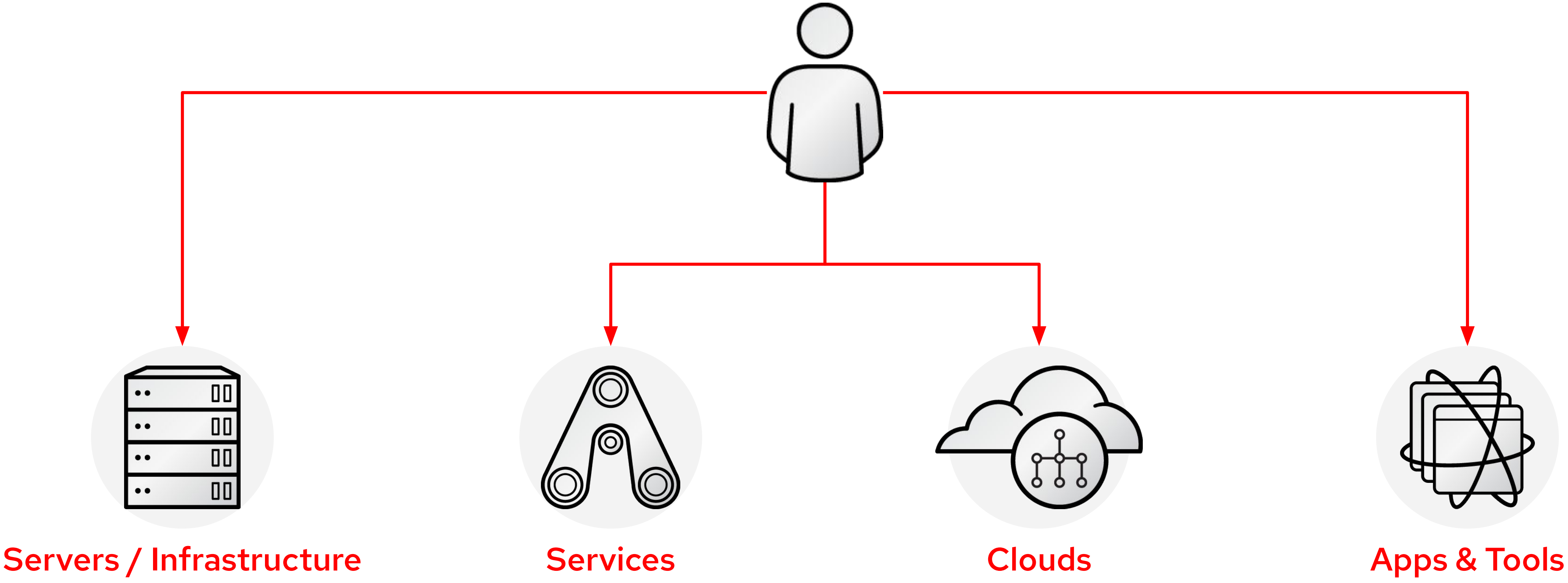
Bare Metal/VM/Container

Developer/QE/DevOps/IT

Source: Identity Management Crash Course LISA 2017

Modern Enterprise

Identity View



Modern Enterprise

Identity View

Employees

Contract-based life cycle

Coordination with company's HRM, ERP system (*Workday, NetSuite, etc.*)

Contractors

Slightly more flexible user life-cycle

Same or different user database as Employees

Customers

Driven by company ERP or standalone CRM software *(Salesforce, SAP, Oracle, Microsoft, etc.)*

Partners

Driven by company ERP or standalone CRM software *(Salesforce, SAP, Oracle, Microsoft, etc.)*

Modern Enterprise

Identity View

Internal Namespace

Employees

Contract-based life cycle

Coordination with company’s HRM, ERP system (*Workday, NetSuite, etc.*)

Contractors

Slightly more flexible user life-cycle

Same or different user database as Employees

Main Focus of RH IdM / this presentation

External Namespace

Customers

Driven by company ERP or standalone CRM software *(Salesforce, SAP, Oracle, Microsoft, etc.)*

Partners

Driven by company ERP or standalone CRM software *(Salesforce, SAP, Oracle, Microsoft, etc.)*

High-Level Mention Only

Administrator’s Challenge

Every networked machine needs **accounts and authentication services**.

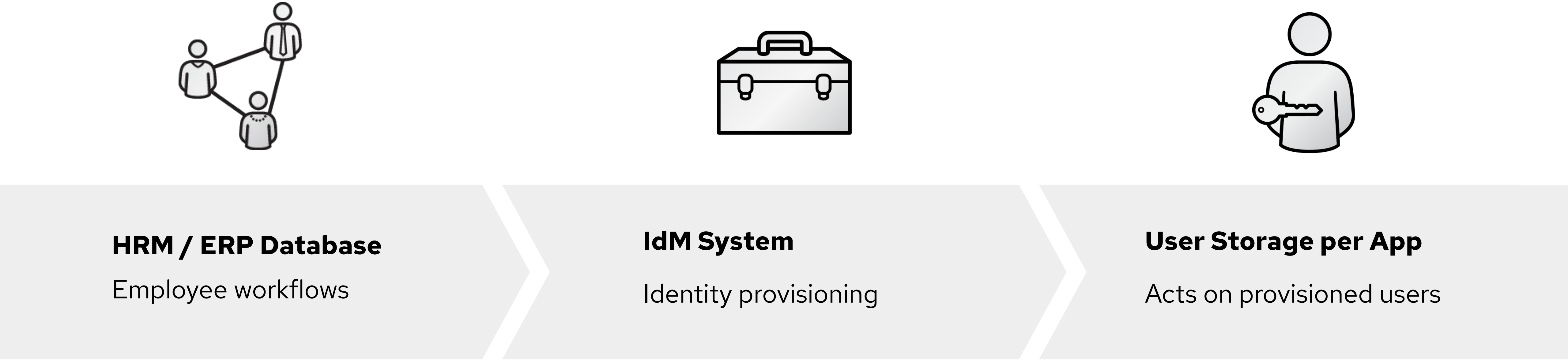
From small startups to big enterprises, from cloud deployments to on-premise, every system admin or devop environment faces the problem of managing users, admins, systems, their credentials and keys, and control and coordinate access.

Purpose built Identity Management systems **reduce errors, and improve productivity of both admins and users by simplifying management**.

Source: Identity and Directories with FreeIPA

Internal Namespace

Traditional Model



HRM / ERP Database

Employee workflows

IdM System

Identity provisioning

User Storage per App

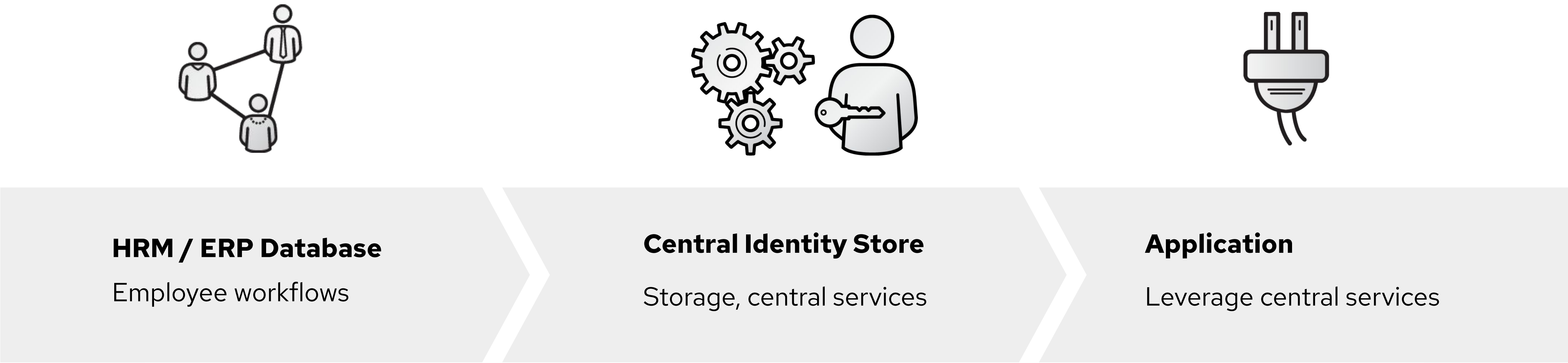
Acts on provisioned users

Cons

Complex, costly. Applications are isolated. Hard to manage and make sure that all systems are aligned. Hard to be compliant with different regulations

Internal Namespace

Modern Model



HRM / ERP Database

Employee workflows

Central Identity Store

Storage, central services

Application

Leverage central services

Pros

Less complex (but not trivial either), thus less costly

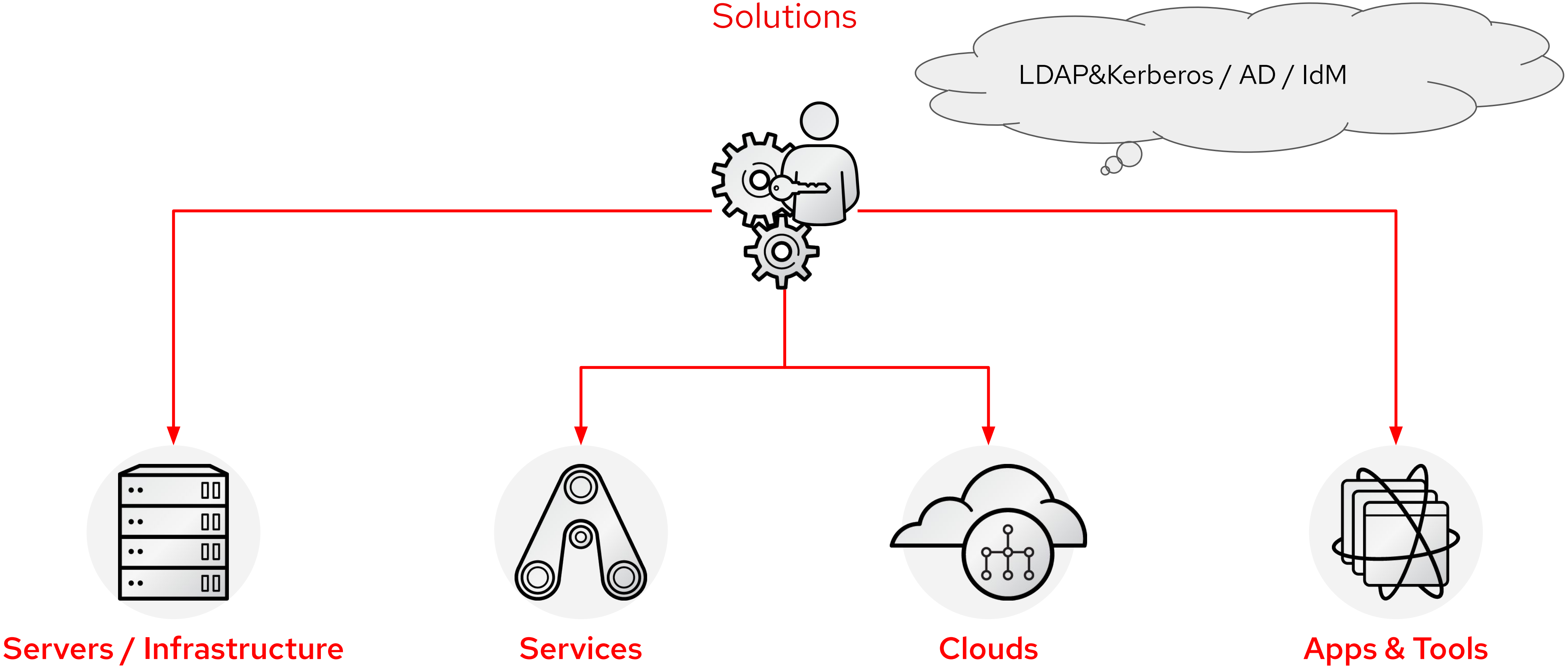
Easier to achieve compliance

Cons

Applications plugs in, but still need additional data –

adds complexity to the app

Modern Enterprise



Solutions

LDAP&Kerberos / AD / IdM

Servers / Infrastructure

Services

Clouds

Apps & Tools

Internal Namespace

Modern Model

Home-grown LDAP/Kerberos

A lot of craft and magic

Hard to support and modernize –> costly

Windows client systems still require AD

SSO?

Active Directory

Difficult to deal with Linux specifics (policies and access control, POSIX, other additional data) and mapping AD specifics to Linux (domains and forests), lack of control over AD. SSO?

RHEL IdM

Built on Linux, for Linux

Can establish Forest Trust with AD

SSO

Windows clients still require AD

IdM Server and Client Interfaces

IdM Server - responsibilities

What is expected from the service?

Identity Store

- Users, Hosts, Services
- Groups

Authentication

- Passwords, 2FA (Smart Cards, OTP soft/hard tokens)
- SSO
- Client/Server certificates (PKI)

Authorization

- Access rules per host
- Privileged operations
- IdM itself - RBAC - user roles and admin delegations

Security-related service management

- Secrets (passwords)
- Linux - SUDO, SELinux, etc.

Auditing and reporting

IdM Server - standard interfaces

How Identity Servers interact with the outer world

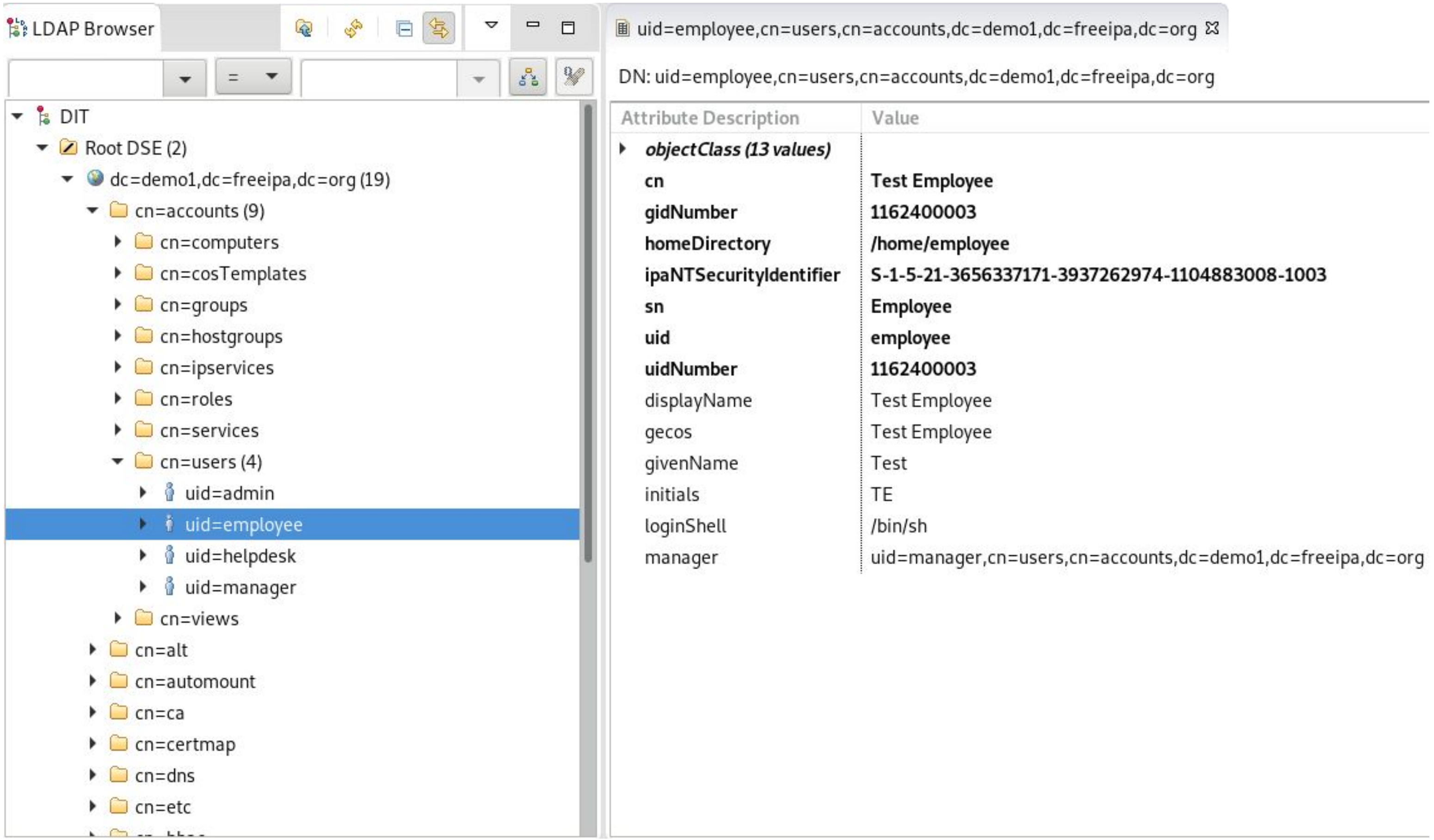
Infrastructure

- **LDAP**: old & proven protocol for sharing data, sometimes authentication too (v3 from *1997)
- **Kerberos**: old & proven protocol for authentication (*1993, revised 2005)
- **Deprecated**: NIS, NTLM

Applications

- **LDAP**: user details, often authentication too
- **Kerberos**: authentication (SSO), mostly for **internal** applications
- **SAML**: old, robust, proven (but may go away too)
- **OAuth 1.0**: old, has weaknesses, should not be used
- **OAuth 2.0 / OpenID Connect** (OIDC): modern, proven, recommended for new applications

IdM Server Interfaces - LDAP



```
DIT
  Root DSE (2)
    dc=demo1,dc=freeipa,dc=org (19)
      cn=accounts (9)
        cn=computers
        cn=cosTemplates
        cn=groups
        cn=hostgroups
        cn=ipservices
        cn=roles
        cn=services
        cn=users (4)
          uid=admin
          uid=employee
          uid=helpdesk
          uid=manager
        cn=views
      cn=alt
      cn=automount
      cn=ca
      cn=certmap
      cn=dns
      cn=etc
```

uid=employee,cn=users,cn=accounts,dc=demo1,dc=freeipa,dc=org

DN: uid=employee,cn=users,cn=accounts,dc=demo1,dc=freeipa,dc=org

Attribute Description	Value
objectClass (13 values)	
cn	**Test Employee**
gidNumber	**1162400003**
homeDirectory	**/home/employee**
ipaNTSecurityIdentifier	**S-1-5-21-3656337171-3937262974-1104883008-1003**
sn	**Employee**
uid	**employee**
uidNumber	**1162400003**
displayName	Test Employee
gecos	Test Employee
givenName	Test
initials	TE
loginShell	/bin/sh
manager	uid=manager,cn=users,cn=accounts,dc=demo1,dc=freeipa,dc=org

Basic features

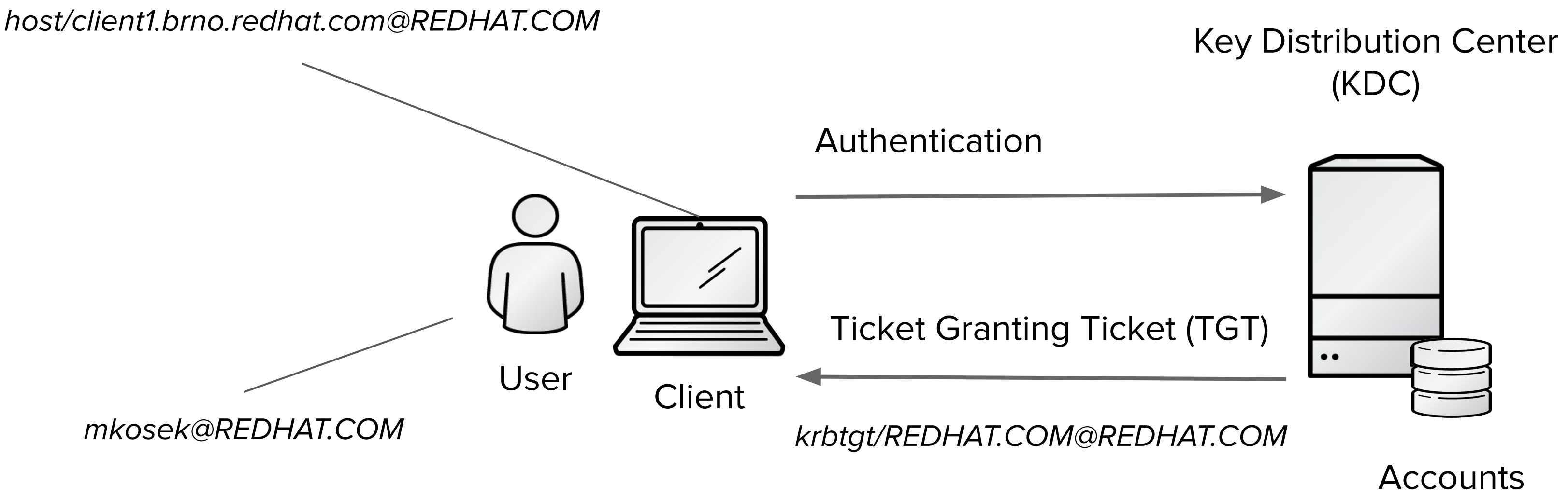
- Tree based directory
- Fast read, slow write
- Multi-master and read-only replication

Why not a custom database? SQL?

- Custom database = custom clients
- Multi-master and read-only replication
- Fine grained Access Control
- Integration, Interoperability

IdM Server Interfaces - Kerberos

User Authentication

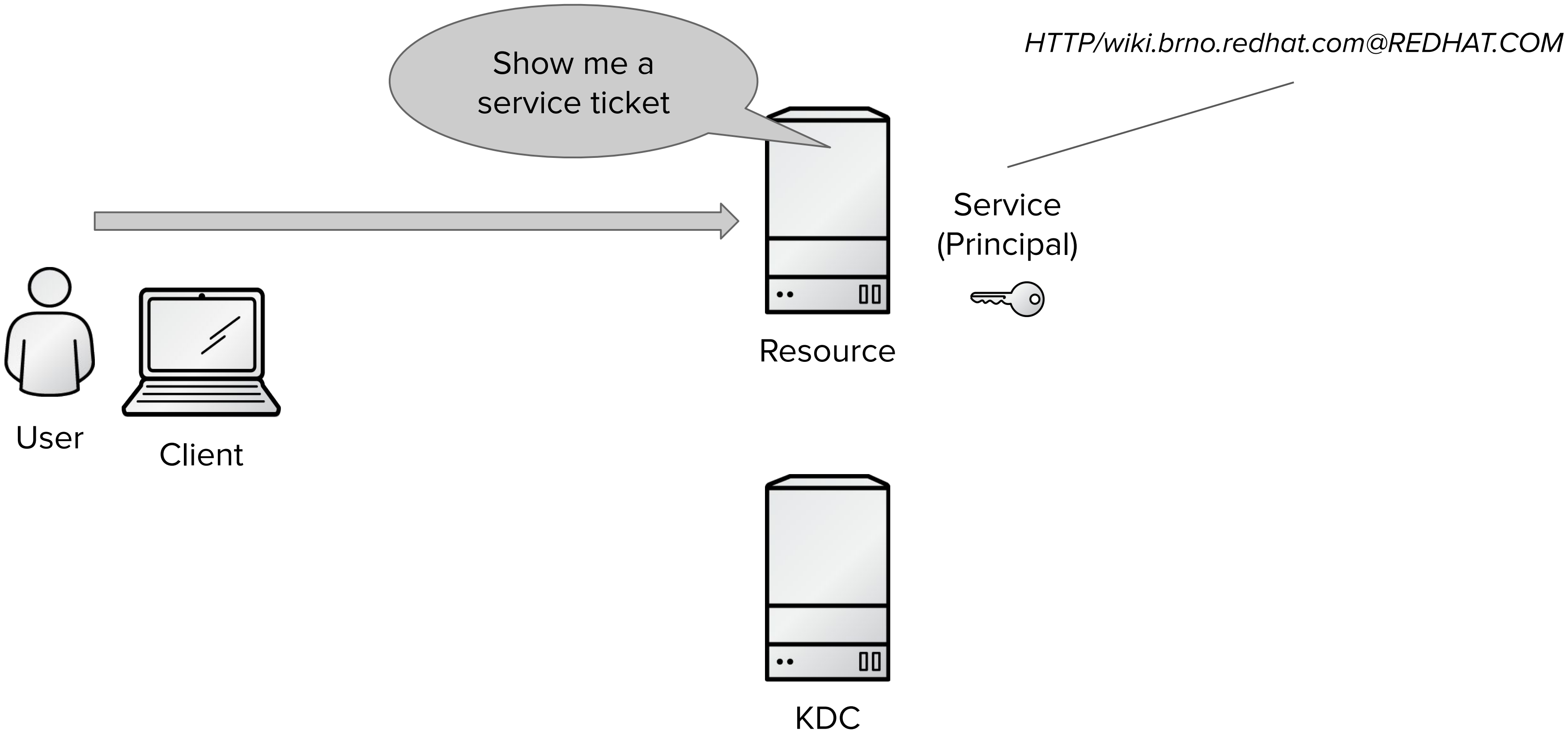

host/client1.brno.redhat.com@REDHAT.COM
Key Distribution Center
(KDC)
Authentication
User
Client
Ticket Granting Ticket (TGT)
mkosek@REDHAT.COM
krbtgt/REDHAT.COM@REDHAT.COM
Accounts

- Password does not leave the system
- Based on a symmetric cryptography, can also use asymmetric for initial authentication

- Different methods: password, 2FA, Smart Card (PKINIT), file keytab

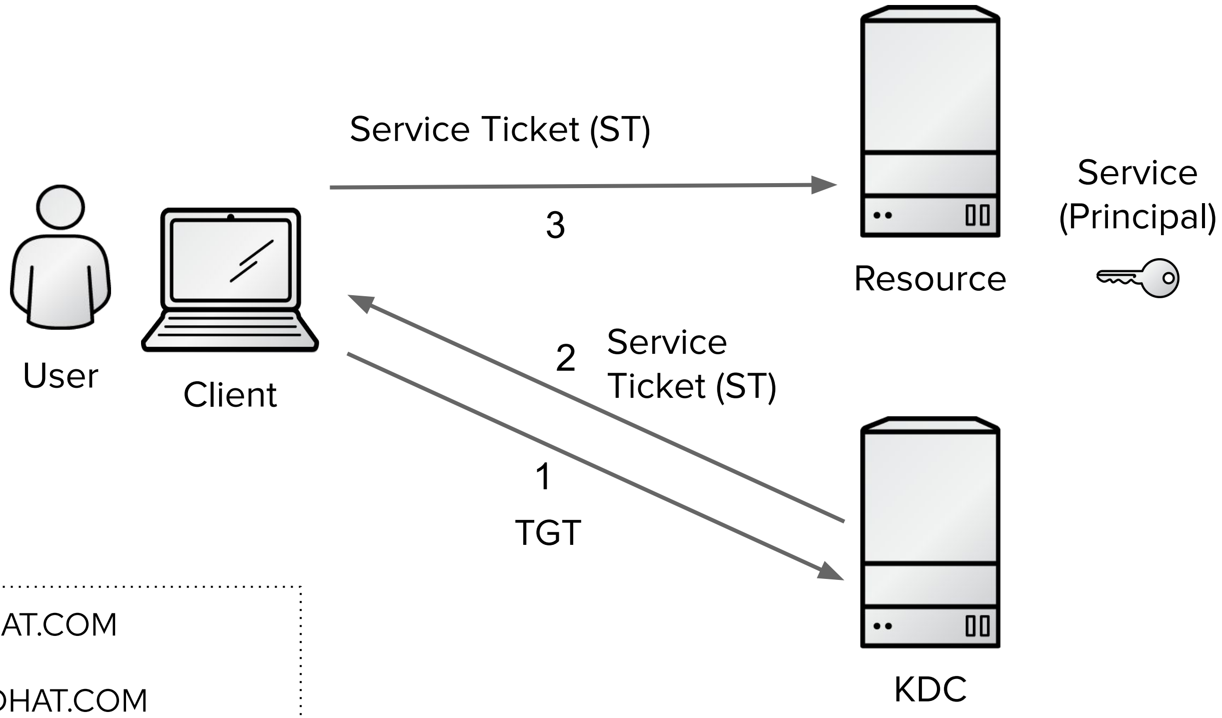
IdM Server Interfaces - Kerberos

Accessing a Resource



IdM Server Interfaces - Kerberos

Accessing a Resource



Kerberos Credential Cache

Default principal: mkosek@REDHAT.COM

HTTP/wiki.brno.redhat.com@REDHAT.COM
krbtgt/REDHAT.COM@REDHAT.COM

IdM Client - Responsibilities

What client (operating system) expects from IdM?

Retrieving Identity information

- Users, Groups, netgroups, host groups, roles
- Certificates, keytabs

Authentication

- Passwords, tickets

Authorization

- HBAC, sudo rules, SSH keys

Misc

- SELinux users
- Automount maps, other configuration
- DNS discovery, DNS Updates, time synchronization

IdM Client - interfaces

Where do IdM services plug in

NSS - Name Service Switch

- Old protocol for Unix-like OS for common configuration databases and name resolution mechanisms (* ~1993)
- Configured in /etc/nsswitch.conf
- Example calls: getpwent(), gethostbyname(), …

PAM - Pluggable authentication module

- Traditional (* ~1995), evolved from Unix PAM
- Mechanism to integrate multiple low-level authentication schemes into a high-level application programming interface (API).
- Authentication stages/groups: account, authentication, password, session
- Example modules: login, sudo, gdm, vsftpd, …

Source: Wiki - NSS, Wiki - PAM, abokovoy-fedora-auth-n-identity-state

IdM Client Interfaces - NSS

/etc/nsswitch.conf

```
passwd:     sss files systemd
group:      sss files systemd
netgroup:   sss files
automount:  sss files
services:   sss files
sudoers:    files sss
shadow:     files sss
hosts:      files dns myhostname
...
```

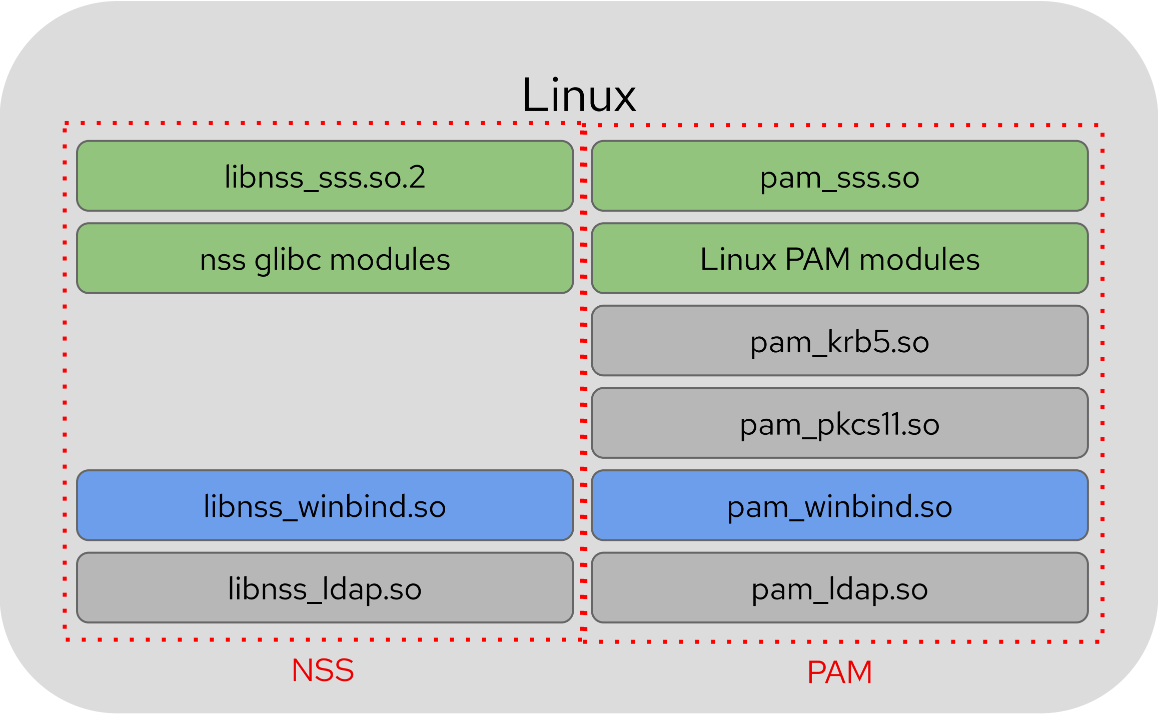
IdM Client Interfaces - PAM

/etc/pam.d/system-auth (one of many in /etc/pam.d/)

```
auth        required                                  pam_env.so
...
auth        requisite                                 pam_succeed_if.so uid >= 1000 quiet_success
auth        sufficient                                pam_sss.so forward_pass
auth        required                                  pam_deny.so

account     required                                  pam_unix.so
...
account     [default=bad success=ok user_unknown=ignore] pam_sss.so
account     required                                  pam_permit.so
...
```

IdM Client Interfaces - Examples

IdM Focused NSS/PAM Modules in a Typical Linux Distribution



🟩 Actively developed

🟦 Actively developed, not preferred/default in RHEL

⬜ Deprecated

Introducing RHEL IdM

IdM Server in RHEL

Centralized Identity Management Server

Introduction

- IdM – Identity Management in Red Hat Enterprise Linux
- Integrates several projects, FreeIPA is the umbrella

Main Interfaces

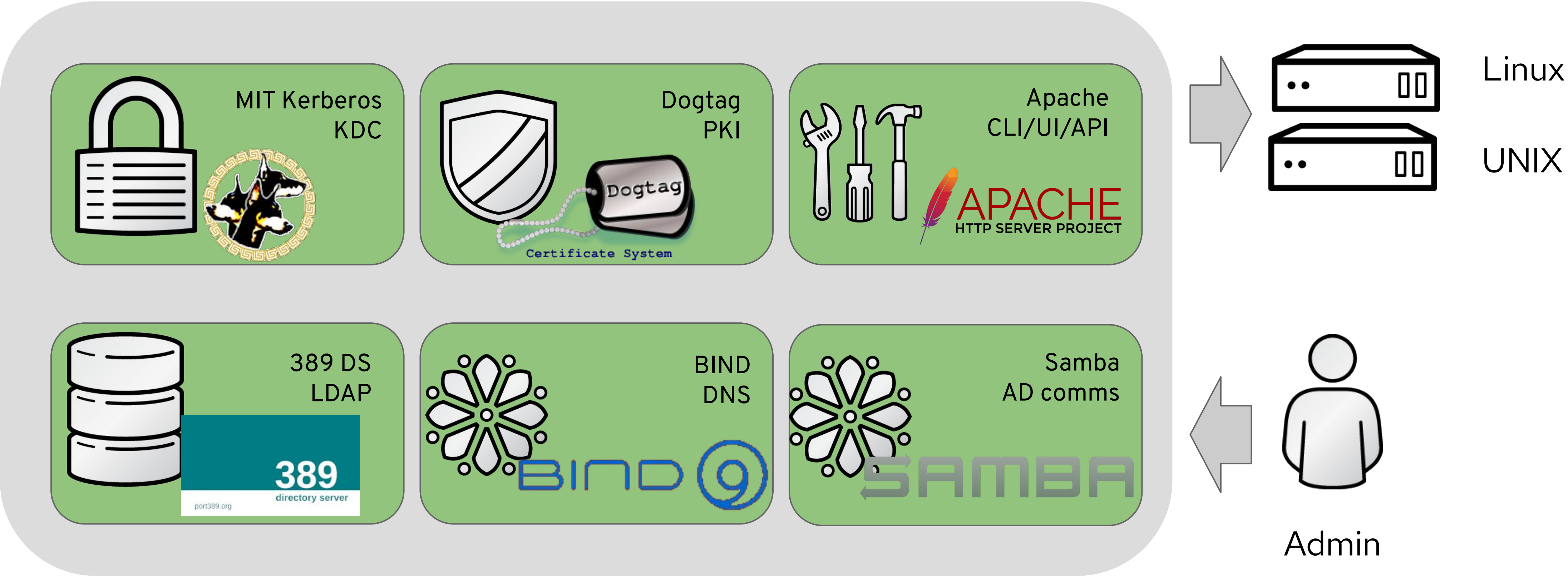
- LDAP, Kerberos
- JSON-RPC API
- AD-specific interfaces

Problems it solves

- Central management of authentication and identities for Linux clients - better than standalone LDAP/Kerberos
- Gateway between the Linux infrastructure and AD

IdM Server

Main Components



Admin

IdM Client - SSSD

Connecting operating system to the Identity Servers

Introduction

- System Security Services Daemon
- Connects Linux system to central identity stores (IdM, AD, LDAP)

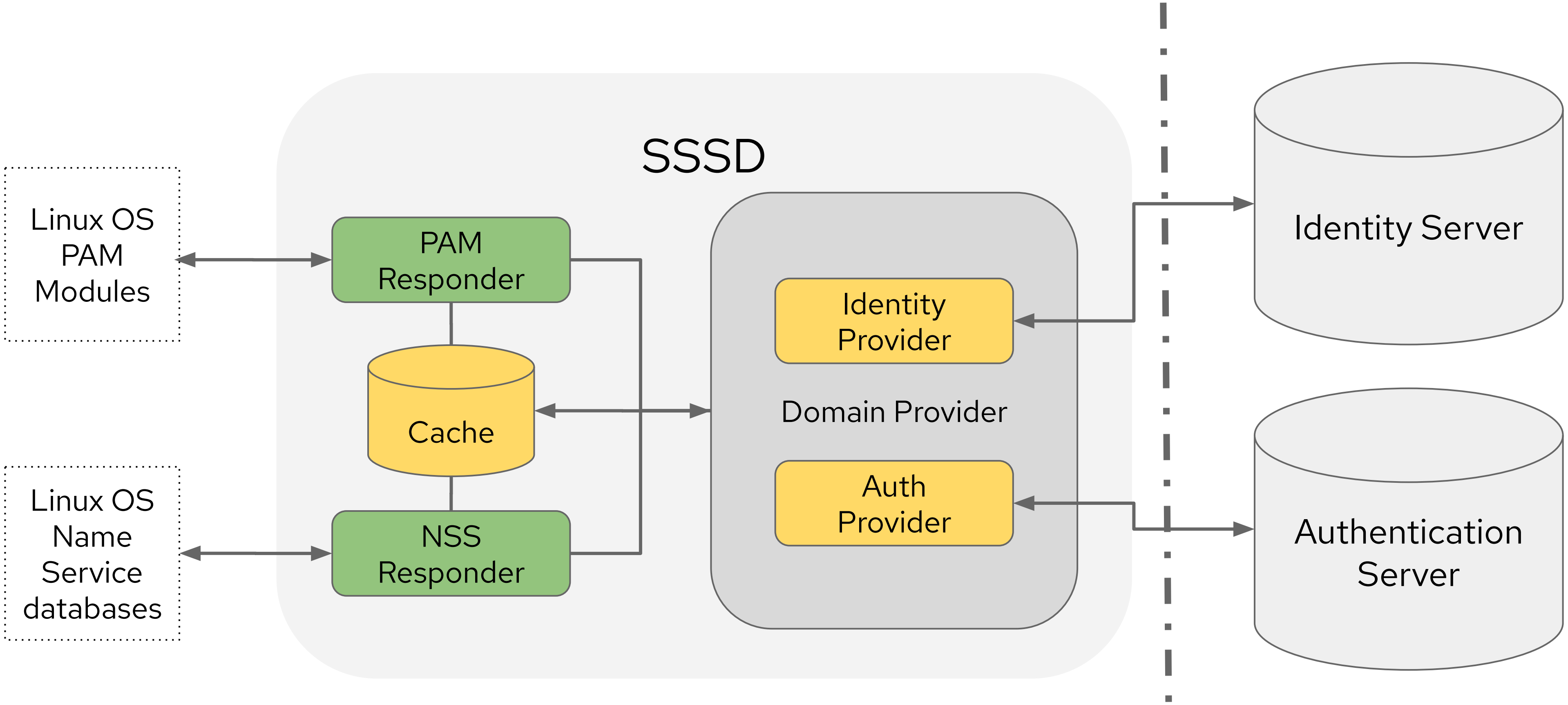
Supported Environments

- Servers: IdM Server, AD, LDAP/Kerberos
- OSes: all major Linuxes; some support in FreeBSD

Main Features

- Caching of information, for offline use case
- Advanced integration with IdM and AD
- Supports Linux features - SUDO, SELinux, 2FA

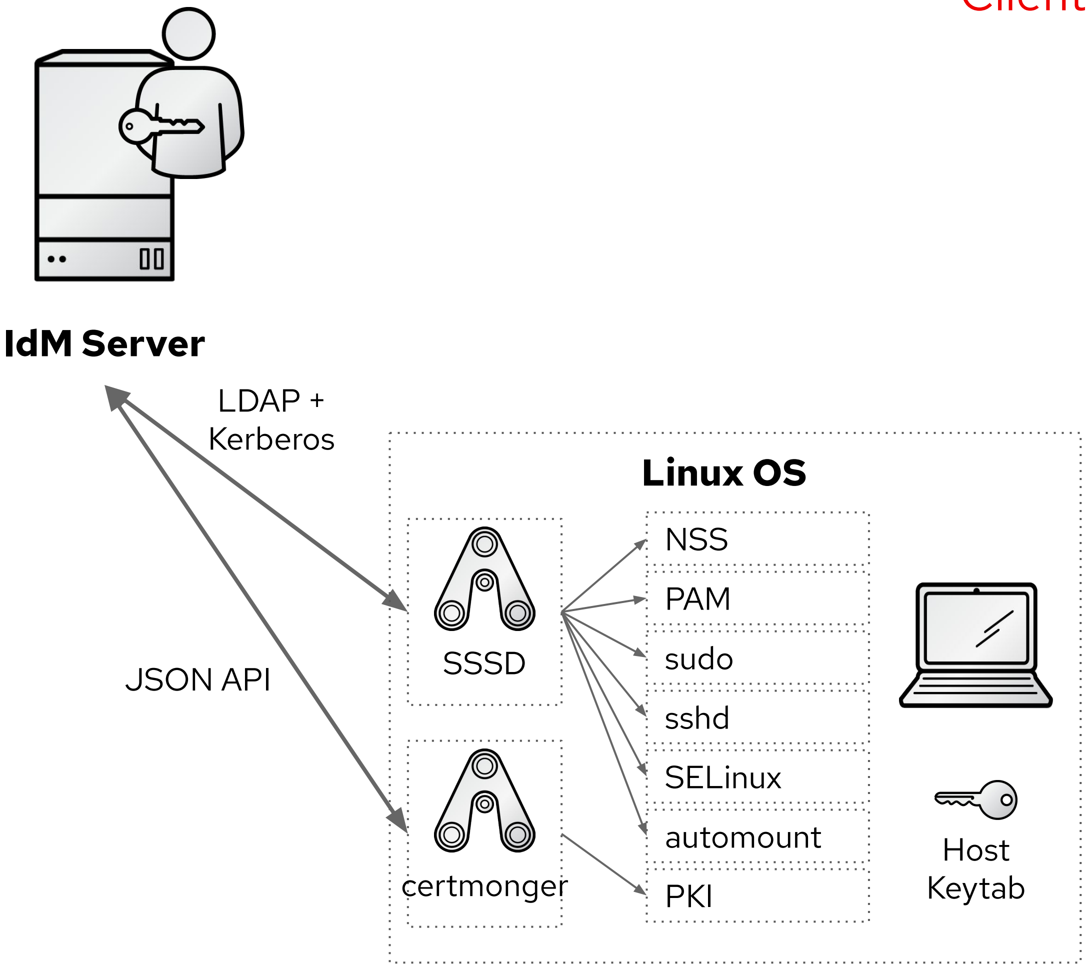
IdM Client - SSSD



id_provider: proxy, files, ldap, ipa, ad
auth_provider: ldap, krb5, ipa, ad, proxy, none

IdM Server

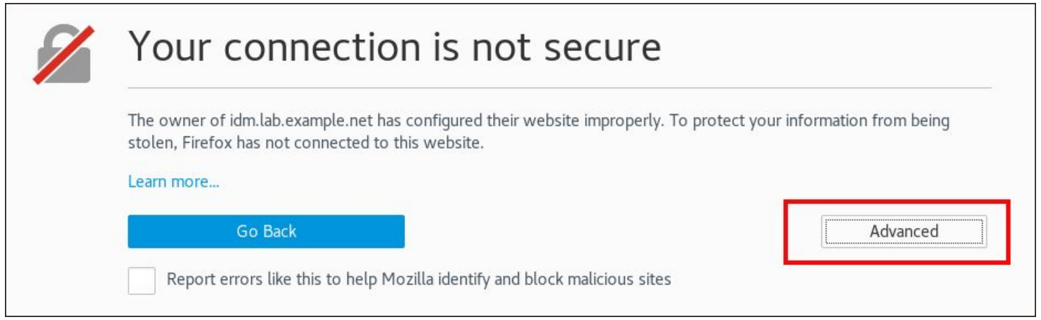
Client OS Integration



- **SSSD**: handles most of the heavy-lifting on the client
  - Identity
  - Authentication + authorization (HBAC)
  - Linux specific integration - SELinux, automount
  - *ipa* and *ad* provider require Host Keytab - used for Kerberos auth or tunneling connections (used for 2FA)
- **certmonger**: optional certificate renewal tool
  - Useful to avoid expired service certificates
  - Can work with both PEM and NSS DB formats

IdM Server Infrastructure

PKI - Capabilities



Your connection is not secure

The owner of idm.lab.example.net has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

Learn more…
Go Back
Advanced
□ Report errors like this to help Mozilla identify and block malicious sites

- Deployment types
  - Self-signed
  - Chained to other CA (typically AD)
  - CA-less
- Capabilities
  - Certificate provisioning for users, hosts and services
  - Multiple certificate profiles
  - Lightweight Sub-CAs (and ACLs who can use them)
- Smart Card authentication
  - PKINIT authentication (Smart Card → TGT)
- Secret store (Vault)

IdM Server Infrastructure

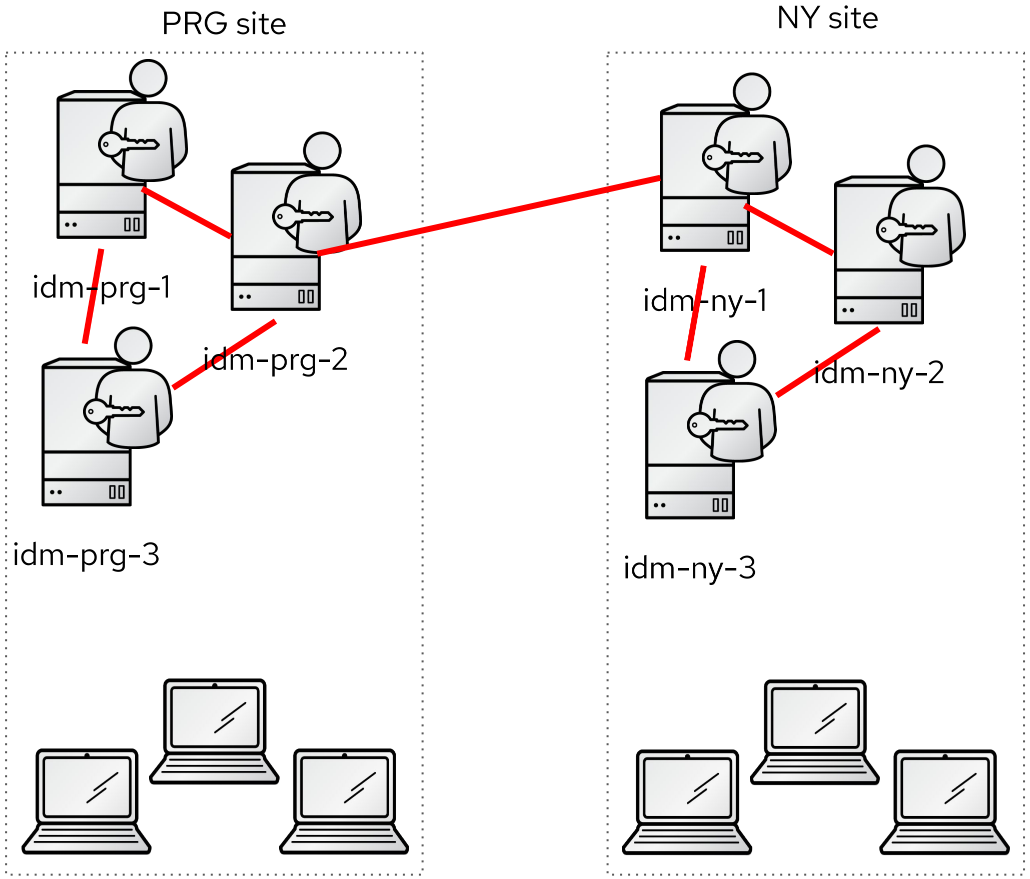
PKI - Tools

```
$ ipa-getcert request -r -f
/etc/httpd/conf/ssl.crt/server.crt -k
/etc/httpd/conf/ssl.key/server.key -N
CN=`hostname --fqdn` -D `hostname` -U
id-kp-serverAuth


$ ipa-getcert request -d /etc/httpd/alias -n
Server-Cert -K HTTP/client1.example.com -N
'CN=client1.example.com,O=EXAMPLE.COM'
```

- Available tools on IdM Server:
  - Tool to **install CA or KRA** (Vault - secret management)
  - Tool to **change deployment type** and rotate CA keys
  - Tool to **change CRL master**
  - Tool to **enable PKINIT** authentication
- IdM Client tools
  - Certmonger - can request and **renew certificates**
    - See example on the left
    - Supports NSS and PEM format
  - Tool to **update CA certificates**

IdM Server Infrastructure

Supports Multi-Master Replication



- Supports multi-server deployment based on the **multi-master replication** (up to 60 replicas)
- Recommended deployment 2K-3K clients per replica
  - Depends on the load - lazy vs. busy clients
- Details depend on the number of data centers and their geo-location

IdM Server Infrastructure

Configuration Tools - Server

RHEL 8.0+

```
# yum module enable idm:DL1

# yum module install idm:DL1/server

# yum module install idm:DL1/adtrust

# ipa-server-install
```

RHEL 6.x and 7.x

```
# yum install ipa-server ipa-server-trust-ad

# ipa-server-install
```

- Server / Replica installer (available since RHEL 6)
  - **Interactive installer** (can run --unattended)
- Preparation before installation
  - DNS is set up
  - PKI chaining is decided
  - Firewall ports are open
- Other tools important for deployment
  - ipa-backup, ipa-restore (but multi-master replication lowers risk already)
  - ipa-healthcheck (from RHEL 8.1)

IdM Server Infrastructure

Configuration Tools - Client

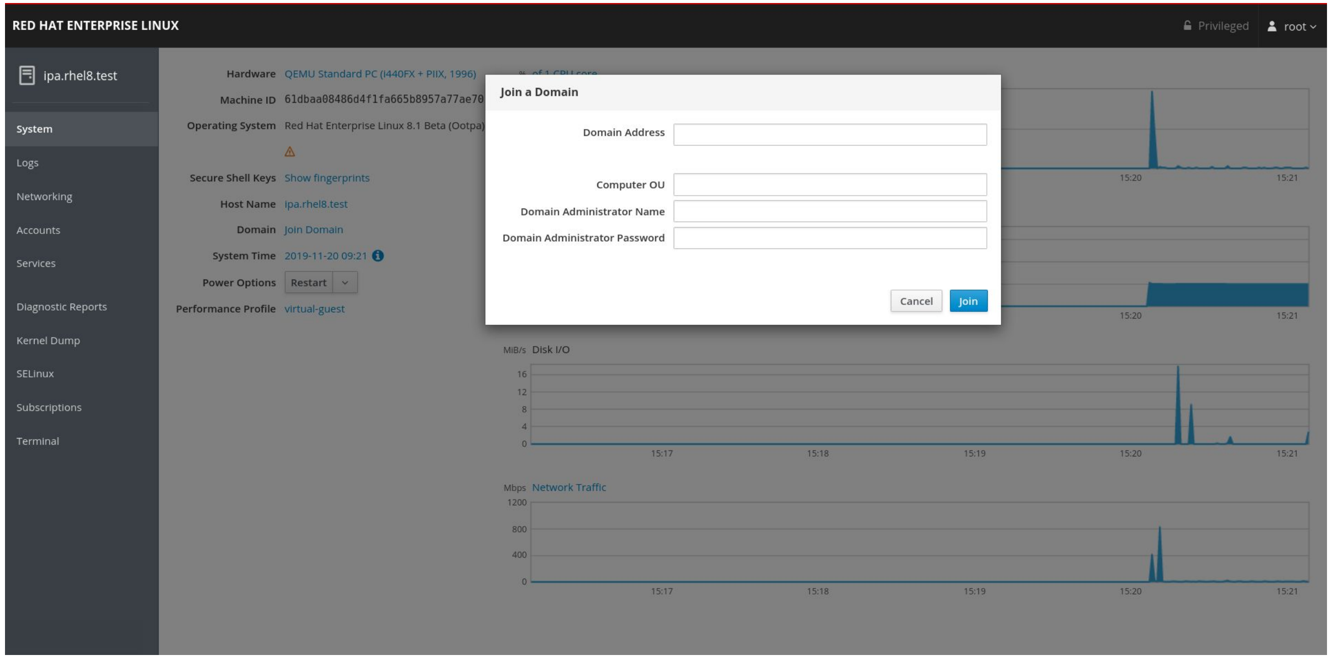
```
# yum install ipa-client     (RHEL 6-7)
# yum module install idm     (RHEL 8.0+)


# ipa-client-install
Client hostname: client.example.com
Realm: EXAMPLE.COM
DNS Domain: example.com
IPA Server: server.example.com
BaseDN: dc=example,dc=com
...
```

- Native client installer (available since RHEL 6)
  - Can autodect server based on hostname
- Can be also installed with:
  - **realmd** - configuration script supporting IdM client, Winbind with different servers (IdM, AD)
  - **GNOME** - in account configuration
  - **Cockpit** - Web Console (SSO doc)
- Lower level tools
  - Authconfig (RHEL 7.x and earlier) / authselect (RHEL 8.0+) - used for NSS/PAM configuration

IdM Server Infrastructure

Configuration Tools - realmd / Web Console

```
# yum install realmd

# realm join ad.example.com

Password for Administrator:



# id user@ad.example.com

uid=1348601103(user@ad.example.com)
gid=1348600513(domain group@ad.example.com)
groups=1348600513(domain group@ad.example.com)
```



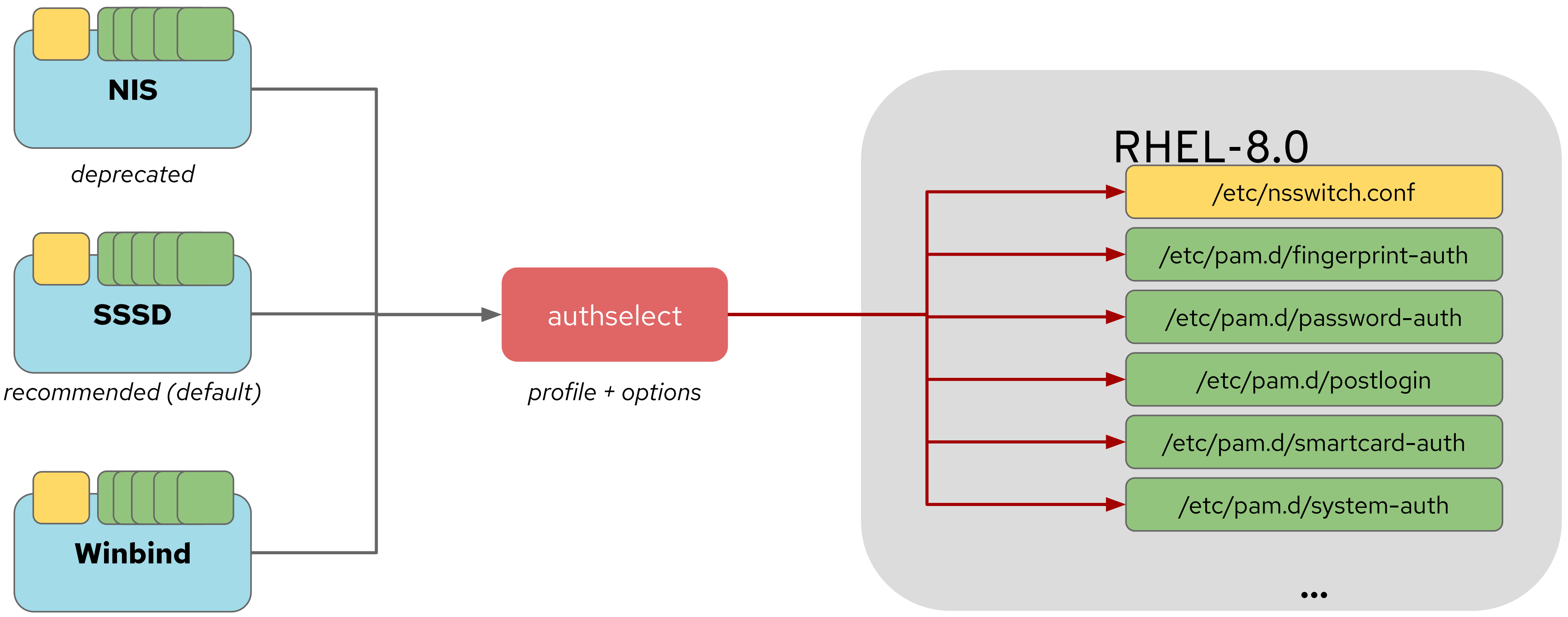
Hardware QEMU Standard PC (I440FX + PIIX, 1996)
Machine ID 61dbaa08486d4f1fa665b8957a77ae70
Operating System Red Hat Enterprise Linux 8.1 Beta (Ootpa)
Secure Shell Keys Show fingerprints
Host Name ipa.rhel8.test
Domain Join Domain
System Time 2019-11-20 09:21
Power Options Restart
Performance Profile virtual-guest

Join a Domain

Domain Address
Computer OU
Domain Administrator Name
Domain Administrator Password

IdM Server Infrastructure

Configuration Tools - Authselect (NSS & PAM)



IdM Server Infrastructure

Configuration Tools - Ansible

```
---
- name: Install IPA servers
  hosts: ipaservers
  become: true

  roles:
  - role: ipaserver
    state: present
```

- Supported Ansible roles and modules
  - Ansible Galaxy, Fedora or RHEL packages (8.1+)
- Roles - Server, Replica, Client
- Modules - topology, user, group, host, etc.
  - Actively developed!

IdM Server Infrastructure

Configuration Tools - Ansible

```
---
- name: Playbook to handle users
  hosts: ipaserver
  become: true

  Tasks:
  - ipauser:
      ipaadmin_password: Secret123
      name: sysop
      first: Sys
      last: Op
      password: "Secret123"
      update_password: on_create
```

```
---
- name: Playbook to handle groups
  hosts: ipaserver
  become: true

  tasks:
   - ipagroup:
       ipaadmin_password: Secret123
       name: sysops
       action: member
       User:
       - sysop
```

ipaadmin_password is not needed if Ansible vault is used for passwords

IdM Server Infrastructure

Configuration Tools - API

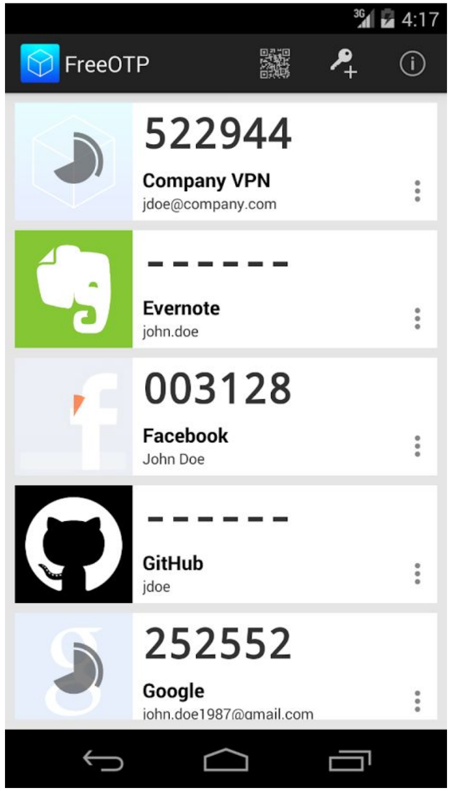
```
curl -v  \
     -H referer:https://$IPAHOSTNAME/ipa  \
        -H "Content-Type:application/json" \
        -H "Accept:application/json"\
        -c $COOKIEJAR -b $COOKIEJAR \
        --cacert /etc/ipa/ca.crt  \
        -d
'{"method":"user_find","params":[[""],{}],"id":0}' \
        -X POST \
        https://$IPAHOSTNAME/ipa/session/json
```

- XMLRPC API (deprecated)
- JSONRPC API
  - Used internally by Web UI, certmonger or other tools
  - API Browser (public demo example)
- Python API libraries

Source: Talking to FreeIPA API with sessions and JSON-RPC

IdM Server Features

User 2-Factor Authentication



4:17

FreeOTP

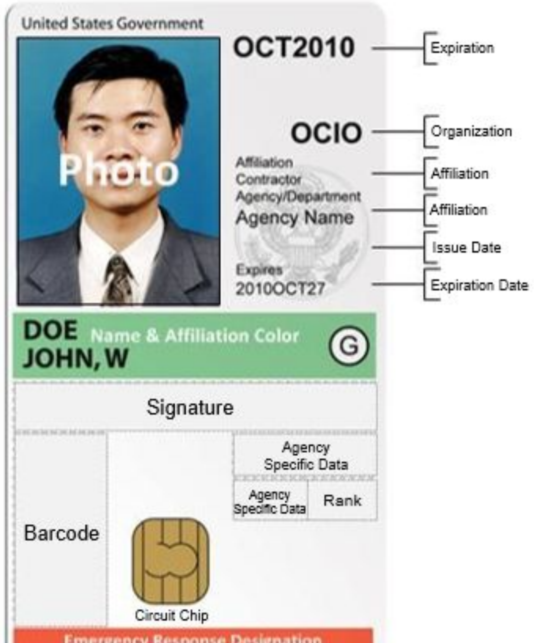
522944
Company VPN
jdoe@company.com

Evernote
john.doe

003128
Facebook
John Doe

GitHub
jdoe

252552
Google
john.doe1987@gmail.com

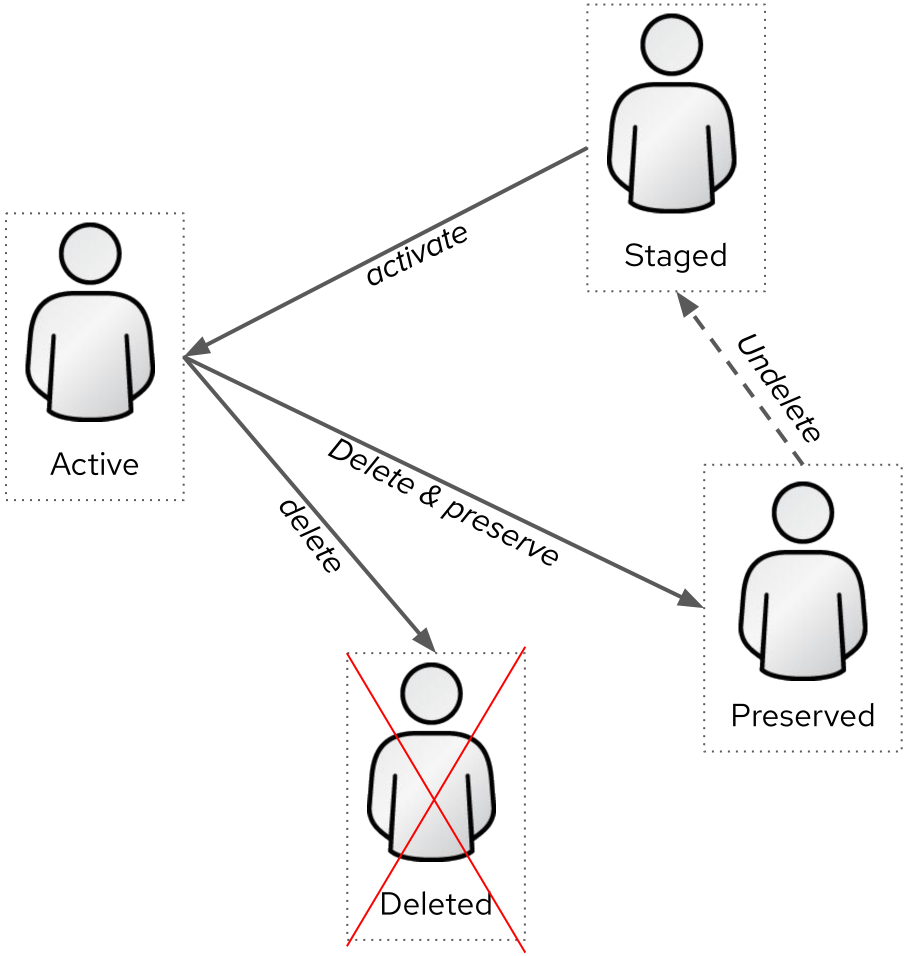


- **OTP**: One-Time Password authentication
  - HOTP and TOTP standards supported
  - Different OTP clients supported - softtoken (FreeOTP, Google Authenticator, …), hardware tokens (YubiKey, RSA SecurID, etc.)
  - Can **proxy 2FA authentication** via RADIUS
- **Smart Card**: physical card, custom device
  - Typical Smart Card requires a special reader attached to a client system; some devices require only USB
  - Typical for high-security environments - governments, finance, healthcare
  - IdM Server and SSSD can contain **rules for mapping SC** to a user

Image source: FreeOTP, wikipedia - FIPS 201

IdM Server Features

Advanced User Life-Cycle



- **User group**
  - Basic user management
  - Can be used in most policy features - HBAC, SELinux, …
  - Available in client Linux OS (POSIX groups only)
- **Automembership**
  - Server can place users in defined groups according to rules based on user attributes
- **Advanced User Life-Cycle**
  - Enables integration with enterprise HR system

IdM Server Features

Policy - HBAC

```
$ ipa hbacrule-show managers_can_ssh_to_ipa
  Rule name: managers_can_ssh_to_ipa
  Enabled: TRUE
  User Groups: managers
  Host Groups: ipaservers
  Services: sshd
```

- **Host Based Access Control**
  - Basic authorization control
  - Based on a tripple - who/where/what
    - **Who**: user or user group
    - **Where**: host or host group
    - **What**: PAM service
- SSSD can print **access control list** for given host
  - *sssctl access-report*
  - Useful for audit purposes

IdM Server Features

Policy - SUDO

```
$ ipa sudorule-show managers_can_reboot
  Rule name: managers_can_reboot
  Enabled: TRUE
  User Groups: managers
  Host Groups: ipaservers
  Sudo Allow Commands: /usr/sbin/reboot
  Sudo Option: type=unconfined_t,
role=unconfined_r
```

- Allows **central management of SUDO** rules
  - When SSSD is used, also caching of them
- Defines SUDO rules allowed for user/host tuple
- Very popular IdM service

IdM Server Features

Policy - SELinux

```
$ ipa selinuxusermap-show managers_are_staff_u
  Rule name: managers_are_staff_u
  SELinux User: staff_u:s0-s0:c0.c1023
  Host category: all
  Enabled: TRUE
  User Groups: managers
```

- Mapping of host & user tuples to a SELinux user role (like *staff_u:s0-s0:c0.c1023*)
- Used in environments with highly restrictive security policies (e.g. military) that require **SELinux MLS policies**

IdM Server Features

Non-Linux & Legacy System Support

- Some OSes do not have native SSSD support and cannot even use IdM user LDAP scheme (RFC2307bis)
- IdM provides **software-generated virtual LDAP tree** (scheme RFC2307)
  - Allows basic LDAP user identity and authentication service
  - Supports AD users when IdM Trust is established
- Caveat: **no other policies** available in the tree
  - **FreeIPA authorization** may be provided with community-supported pam_hbac project

Image source: Wikipedia

Application Integration

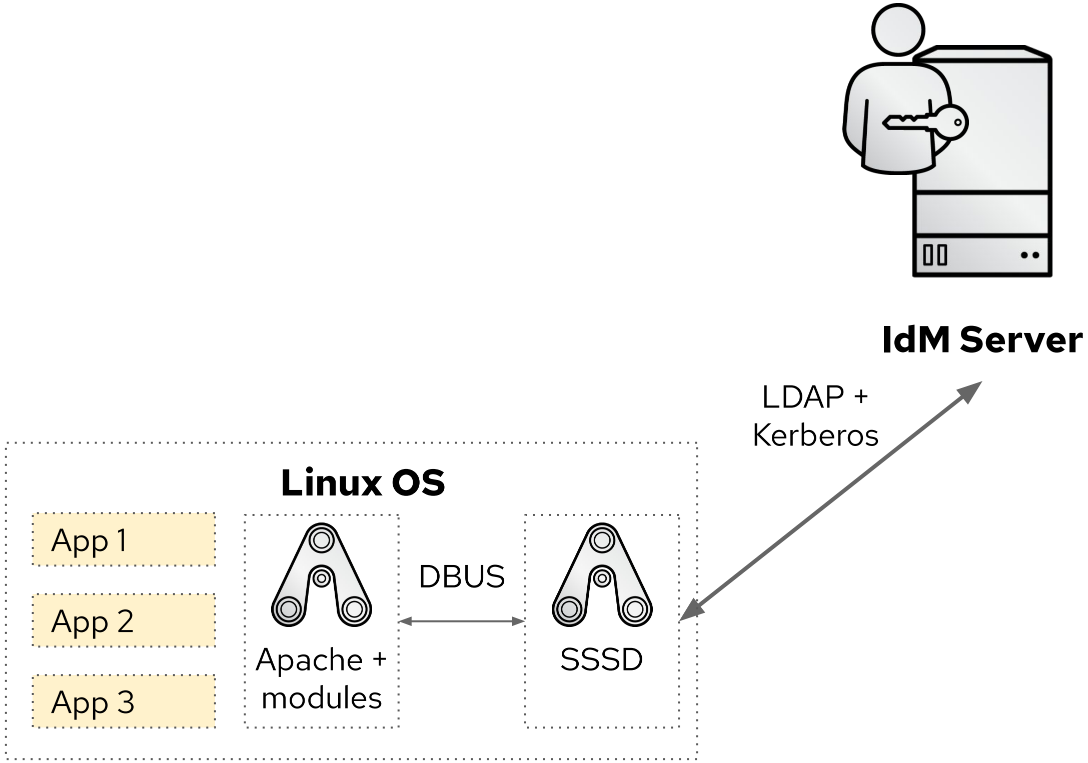
Developing an Application

- Proper user management in application is too often just **afterthought** for development (not cool-enough for MVP)
- Where does the **complexity** come from?
  - Ordinary users, admins, different levels, …
  - Starts with local user database – easy! Then – requests to integrate with company LDAP/AD or external directory, support for different modes of function (Dev, Demo, POC, Production), different user powers, groups, etc.
- What can be **externalized**?
  - Identity Lookup: AD, FreeIPA (IdM), LDAP, SAML, OIDC
  - Authentication and Authorization: LDAP, Kerberos/GSSAPI, Certificate, SAML, OIDC
- **How?**

Source: External Auth and Access Control for Web

Externalizing Authentication & Authorization

Method 1: Leveraging Platform and Web Server (Apache)



- Apache has lot of **basic modules available**
  - mod_ssl, mod_auth_gssapi, mod_auth_oidc, …
- Apache can even **leverage SSSD** from the OS - mod_intercept_form_submit, mod_authnz_pam (on the left)
  - Apache modules leveraging SSSD running on the platform
  - Will offload the complexity to SSSD (IdM, LDAP, trust with AD, etc.)
  - Requires full control over the server
- **Requires control** - ability to configure Apache

- More details: FreeIPA.org - Web App Authentication

Source:

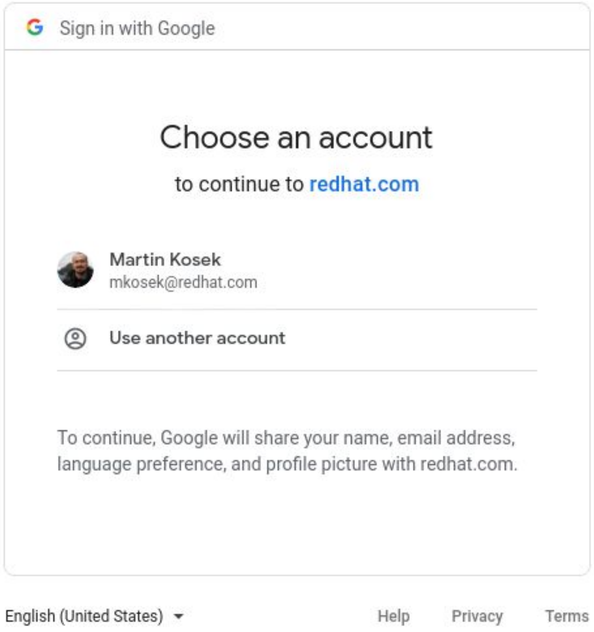
Externalizing Authentication & Authorization

Method 1: Leveraging Platform and Web Server (Apache)

	Authentication	**Access check**	**Extra user info**
Kerberos	mod_auth_gssapi	mod_authnz_pam	mod_lookup_identity
Certificate	mod_ssl		
Forms based	mod_intercept_form_submit		
SAML	mod_auth_mellon		
OpenID Connect	mod_auth_openidc		

Externalizing Authentication & Authorization

Method 2: Leveraging App Libraries and Federation



Sign in with Google

Choose an account

to continue to redhat.com

Martin Kosek
mkosek@redhat.com

Use another account

To continue, Google will share your name, email address, language preference, and profile picture with redhat.com.

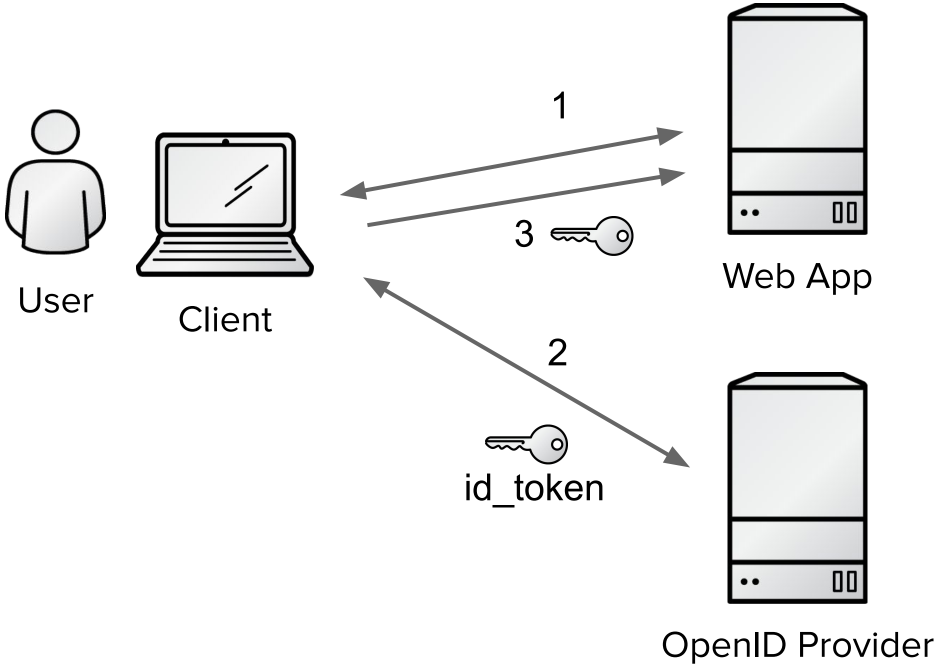
English (United States) Help Privacy Terms

- App libraries – available for traditional LDAP, but especially **for federation protocols**
  - flask-OIDC
  - django-saml2-auth
  - jumbojett/OpenID-Connect-PHP
- Suitable for external applications, without control over the web server or the platform
- **Active development ecosystem** around the libraries
- Lot of free Identity Providers and Authorization Servers (Google, Microsoft, Facebook, GitHub, etc.)
- Open Source infrastructure available as well – Keycloak / Red Hat Single Sign On

Source:

Externalizing Authentication & Authorization

Method 2: How It Works - Basic OIDC Workflow Example ("Implicit Flow")

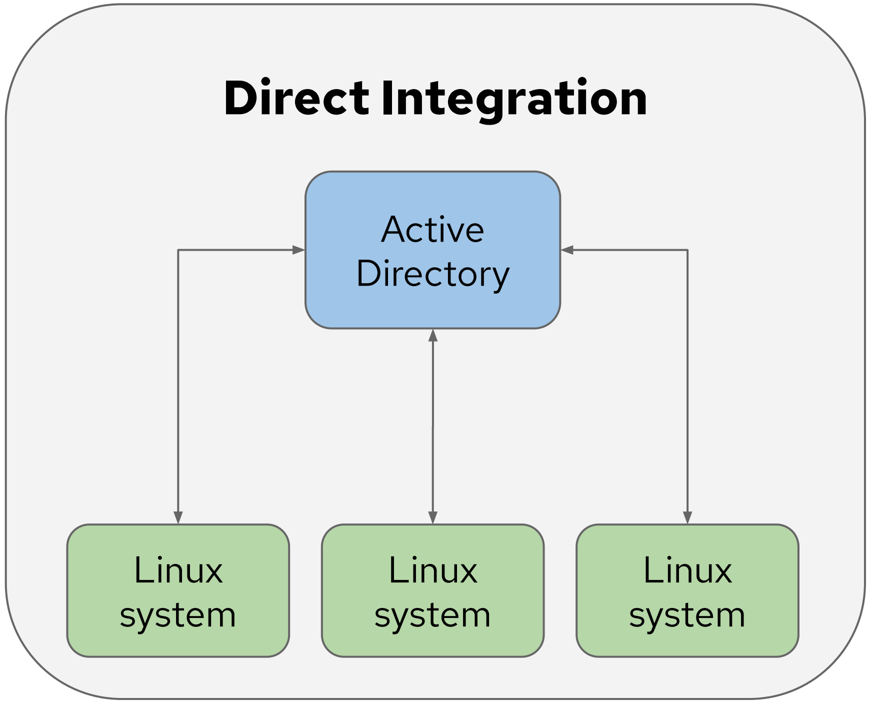


1. User **accesses a Web App**, requiring OIDC Sign In first
2. User is **redirected** to OpenID Provider (with *client_id* of the Web App)
   a. OpenID Provider authenticates and authorizes the user (from own DB, or account from other OpenID Provider)
   b. User details are encoded into an id_token (JWT) that contains user information and signature
   c. Redirects the session back to configured Redirect URI
3. Web App **confirms** *id_token* and confirms signature

Active Directory Integration

Integration Options

Direct Linux-AD Integration



- ● Different ways: SSSD, Winbind, 3rd party
- + Easy to set up (mostly), lower maintenance cost
- − Cannot control Linux native user attributes (POSIX) or policies (SUDO, HBAC, SELinux)
  - Some available via AD schema extensions (got more difficult after Windows Server 2016)
  - Authorization available with AD GPOs
- − Can get expensive (AD device CALs, 3rd party license), Linux is 2nd class citizen

Voice of the Customer

Direct Linux-AD Integration - Customer

I **do not know how to manage** all these Linux systems. Linux admins should do it, but I do not want to **give Linux admins all the control** and access to my AD.

I do not want to **pay premium** for all those third party solutions that cost a fortune. If I buy Linux it should **come with the solution** to integrate it with AD.

It is really hard to do my job if everything is controlled by AD. I have to **ask AD guys** for every small change I need to make. I wish I could have some **control and independence**.

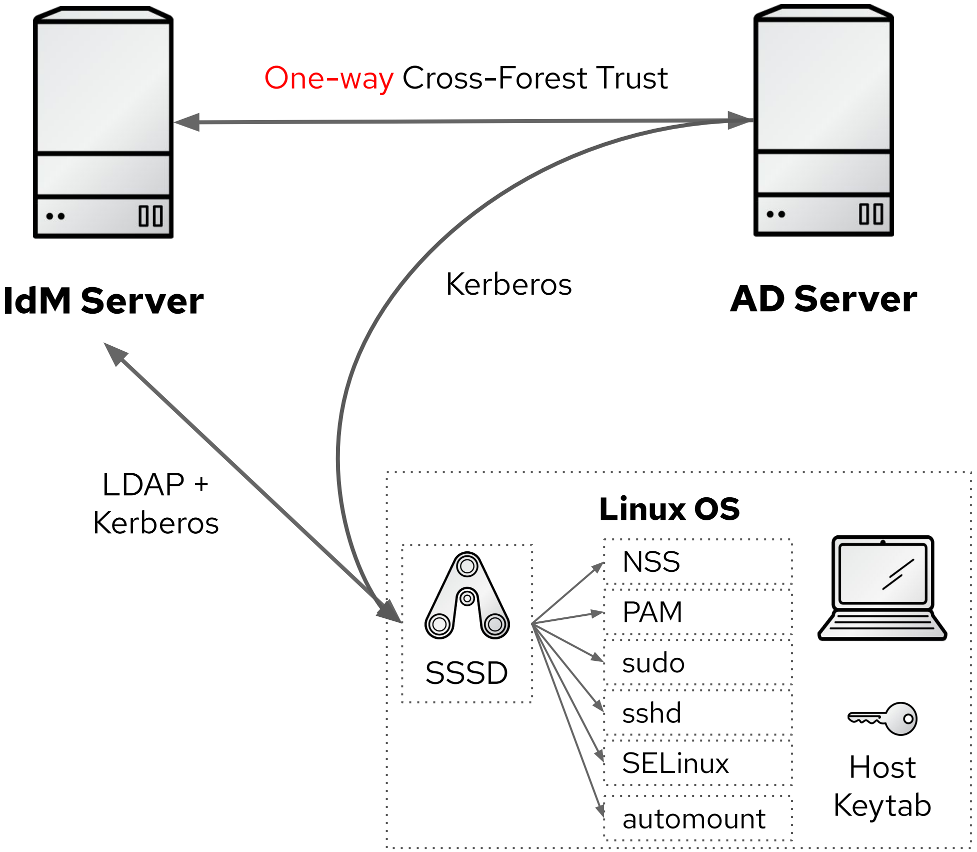
AD Admin

Decision Maker

Linux Admin

Integration Options

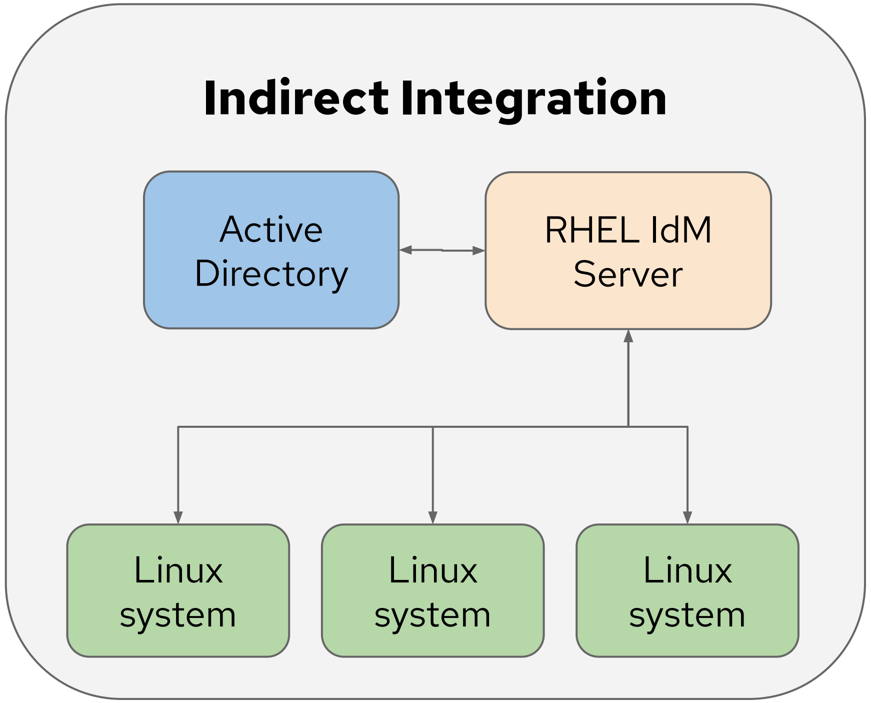
Indirect Linux-AD Integration



- IdM Server **behaves** as another AD Forest Root
  - Provides expected interfaces - LDAP, Kerberos
  - Leverages Samba for AD-native protocols
- IdM and AD **trusts** each other for identity and authentication
  - Actual authentication happens against AD (with cross-realm TGT)
  - One-Way trust (IdM trusts AD) - will change in future
- PKI or DNS can be easily chained to AD too
- IdM can *override* some of the AD user settings
  - Look for "IdM ID Override" in the documentation

Integration Options

Indirect Linux-AD Integration - Benefits



- ＋ Separation of Administrator duties
- ＋ Higher control about forest security (SID filtering)
- ＋ Enables independent growth of the Linux environment
- ＋ Reduces licensing cost (no CALs or 3rd party)
- ＋ Centralized flexible certificate mapping and rulesets for smart-card authentication (both IdM and AD users)
- — Maintenance overhead
- — Requires proper setup (DNS, relationships) and minimal architectural knowledge

Voice of the Customer

Indirect Linux-AD Integration - Customer

I can **delegate managing Linux** to Linux admins. I just set up trust with IdM and they manage Linux systems while **I manage users**.

AD Admin

I **do not have to pay** for the costly third party solutions. **I can now afford more** (Red Hat) infrastructure to support the needs of my business applications!

Decision Maker

I have all means and tools to provide centralized management of my Linux and UNIX environment and **I can do my job** well without asking AD guys.

Linux Admin

Main Selling Points

Smoother Linux Adoption

AD Integration

IdM solves the problem of integrating of the Linux systems and infrastructure into a data center dominated by Active Directory for AD-centric customers (around 90%!)

Separation of Duties

IdM and AD Administrators rule their realms - permission from AD Administrator not needed to install a new Linux system

Cost Reduction

IdM included in base RHEL subscription

No extra cost for client licenses on AD side

No extra cost for 3rd party integration solution

IT Optimization

Simplification

Make user management workflows easier. **No cloning of users** to different IdM systems. No one-purpose Identity servers. The IdM vision is that there is just IdM (and AD) in the infrastructure.

Remove Custom Infrastructure

Get rid of old **infrastructure "cruft"** - custom LDAP+Kerberos servers, NIS servers, etc.

IdM can work with both modern (Linux) systems and also the UNIXes (Solaris, HP-UX, AIX, etc.)

Enable SSO for Entire Infrastructure

Get **SSO authentication for infrastructure and web applications** using Kerberos.

External web authentication can be solved via federation protocols (SAML, OpenID Connect) – IdM can be integrated with Red Hat SSO.

Secure Authentication from kickstart

Automated deployment with **preconfigured Identity, Authentication, Authorization** – for bare metal, VMs, containers

Technologies: kickstart+realmd, Ansible, IdM API

Regulated Environments

Expectations

- Regulated environments have **higher security and compliance expectations**
  - Governments, Finance, Healthcare, …
  - PCI-DSS, FIPS 140-2, FedRAMP, DISA STIG, etc.
- Frequently, requirement to use **2FA / Smart Cards**
  - IdM Server and SSSD supports both server-based Smart Card management and local-only for air-gapped systems

Regulated Environments

PCI-DSS Compliance Study

PCI-DSS Requirement	What is Required	IdM Technologies/Features
Requirement 2: Do Not Use Vendor-Supplied Defaults for System Passwords and Other Security Parameters	Default users, passwords	IdM Server: centralized accounts, SSH settings
	Security parameters	IdM Server certificate tools (certmonger) IdM Server HBAC IdM Server SUDO
Requirement 6 – Develop and Maintain Secure Systems and Applications	Secure application development and testing	IdM Application Apache modules Red Hat SSO (Keycloak) with SSSD backend (SAML, OIDC)
Requirement 7 – Restrict Access to Cardholder Data by Business Need to Know	Access control and limiting the privileges of administrative accounts	IdM Server HBAC IdM Server SUDO IdM Server SELinux User Mapping

Source: https://www.redhat.com/en/blog/identity-management-and-related-technologies-and-their-applicability-pci-dss

Regulated Environments

PCI-DSS Compliance Study

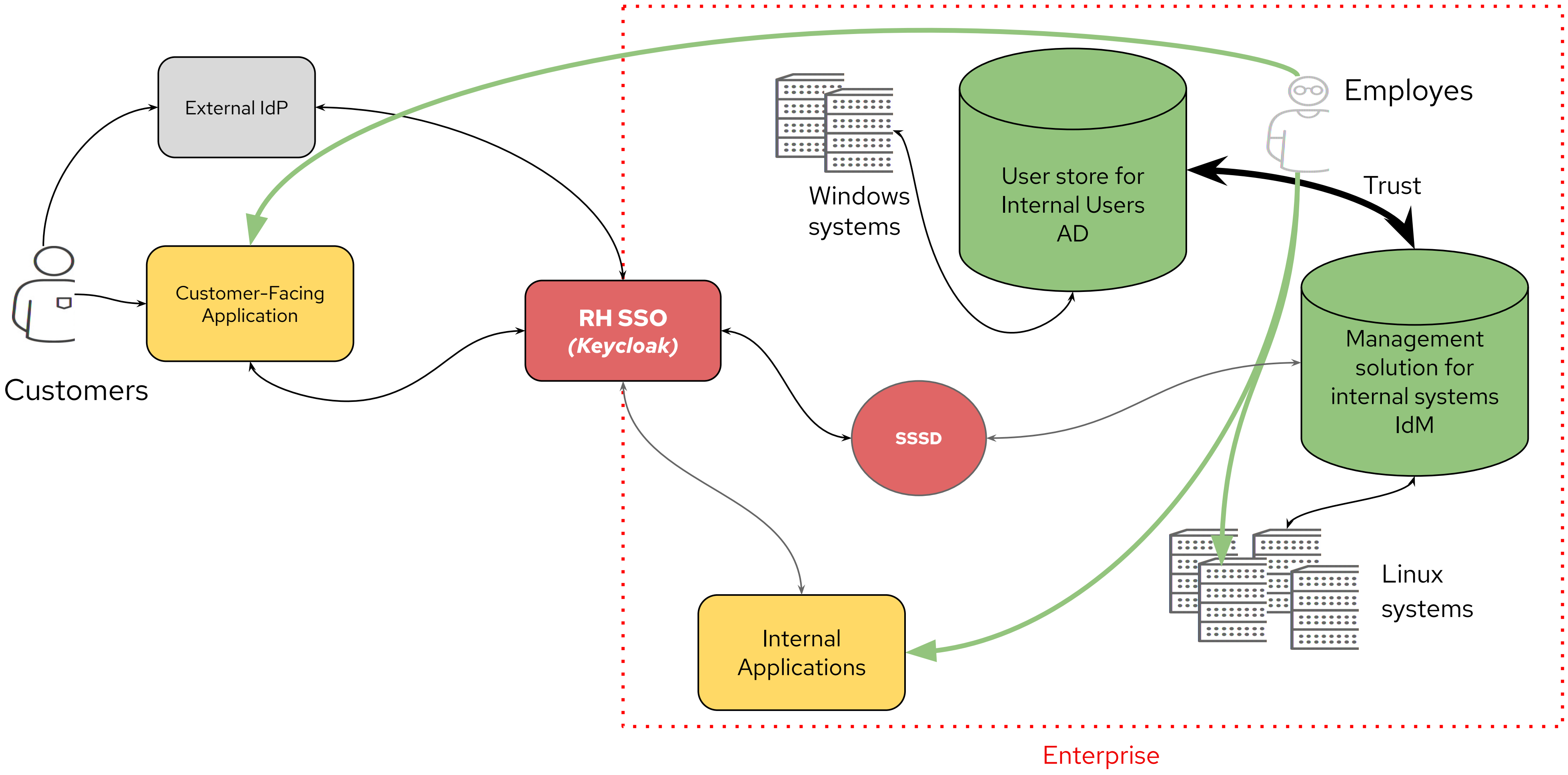
PCI-DSS Requirement	What is Required	IdM Technologies/Features
Requirement 8 – Identify and Authenticate Access to System Components	Identify and authenticate access to system components	IdM Server + SSSD in general
	Multi-factor authentication	IdM Server 2FA (Smart Cards, Yubikey, FreeOTP)
Requirement 10 – Track and Monitor All Access to Network Resources and Cardholder Data	Audit and Monitoring	RHEL audit trail (audit subsystem, logs, rsyslog) Session Recording SSSD Attestation Report

Source: https://www.redhat.com/en/blog/identity-management-and-related-technologies-and-their-applicability-pci-dss

Bringing It All Together

How It All Fits Together?

Example Deployment



Source: Identity Management in Red Hat Enterprise Linux - Dmitri Pal, Summit 2019

Red Hat

Demo

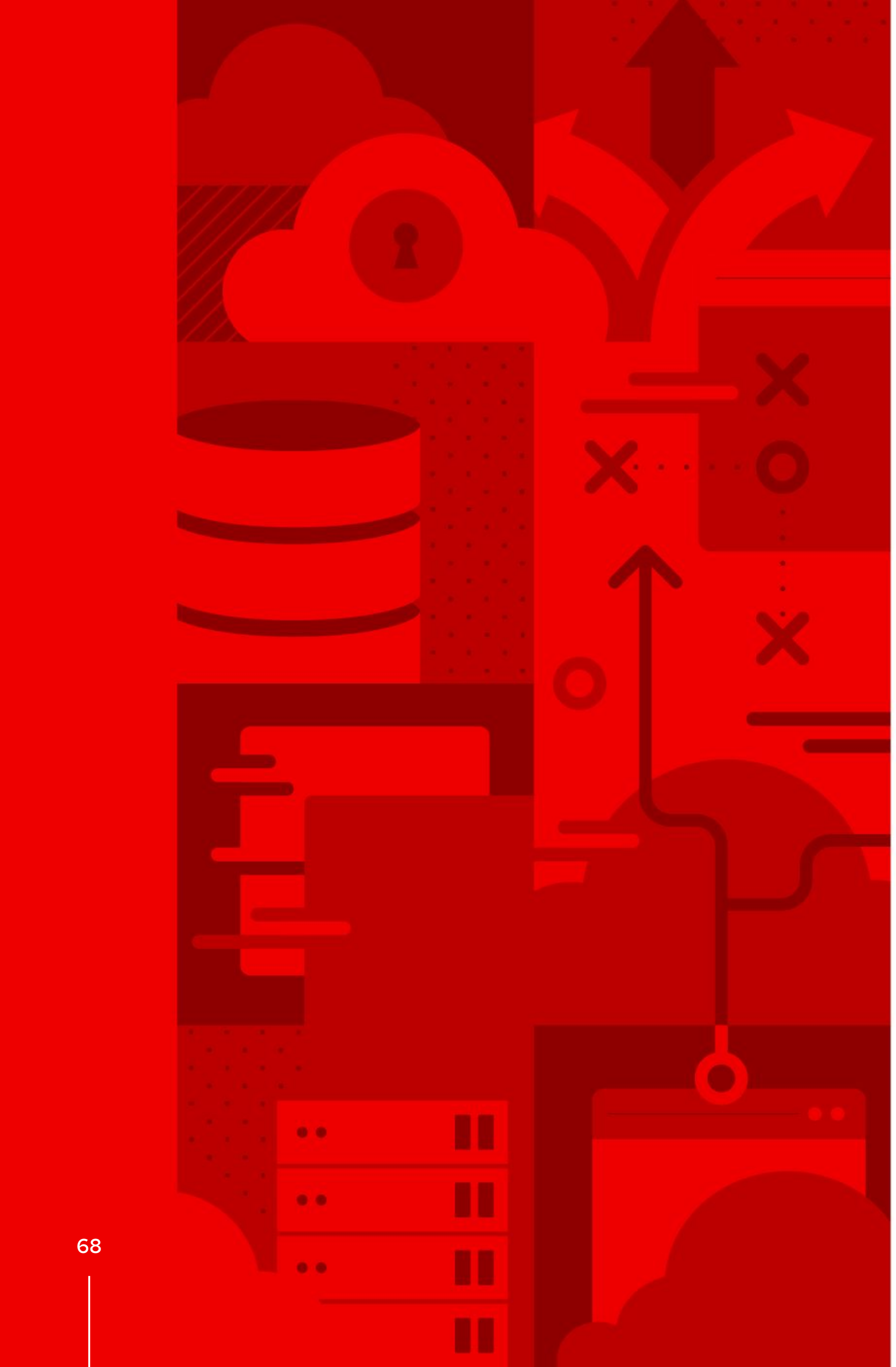
https://ipa.demo1.freeipa.org/ipa/ui/

More Information

Contacts, feedback

Community

Project pages: FreeIPA | SSSD | Directory Server | Certificate Server (active *-users lists!)



Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

in linkedin.com/company/red-hat	f facebook.com/redhatinc
▶ youtube.com/user/RedHatVideos	twitter.com/RedHat

Red Hat