Trusting Active Directory with FreeIPA: a story beyond Samba

Alexander Bokovoy <ab@samba.org>

Red Hat

May 15th, 2014

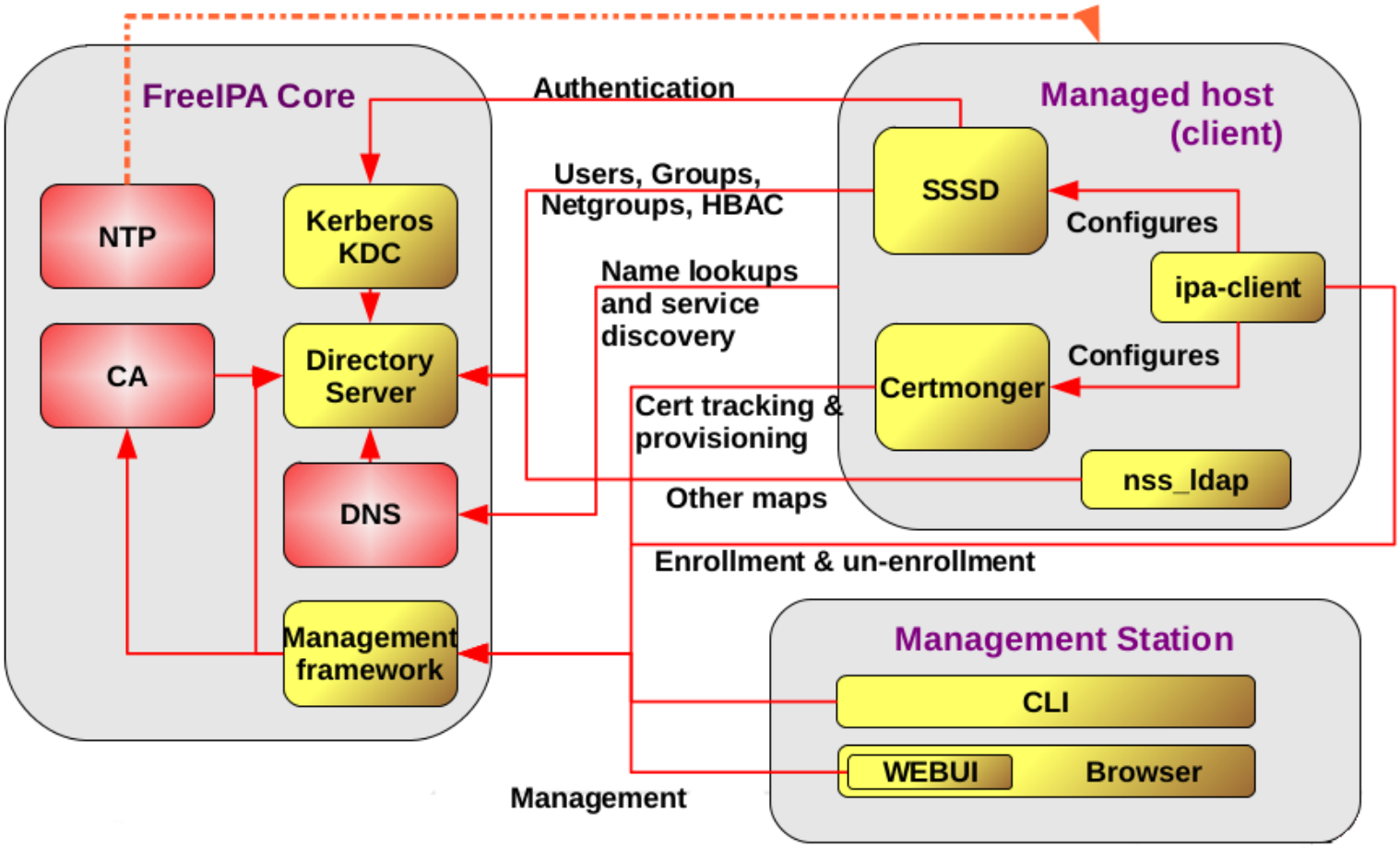
Trusting Active Directory with FreeIPA

Trusting Active Directory with FreeIPA
A story beyond Samba

FreeIPA: http://www.freeipa.org

- **I: Identity**
  - LDAP-based store for common objects (users, groups, hosts, services, ...)
  - 389-ds as an LDAP server with FreeIPA server-side plugins
  - MIT Kerberos KDC with FreeIPA driver
  - Integrated certificate management with Dogtag Certificate Authority
  - Python-based command line and Web management tools
- **P: Policy**
  - Delegation and separation of access
    - Flexible delegation of editing controls
  - Host-based access controls to services:
    - Everything is denied by default, define rules to allow
    - <user or group[, source host]>→<host, service>
  - Rules enforced at client side with **SSSD** project
- **A: Audit** Coming...

FreeIPA: http://www.freeipa.org



Trusting Active Directory with FreeIPA
A story beyond Samba

Kerberos cross-forest trusts

FreeIPA deployment is a fully managed Kerberos realm

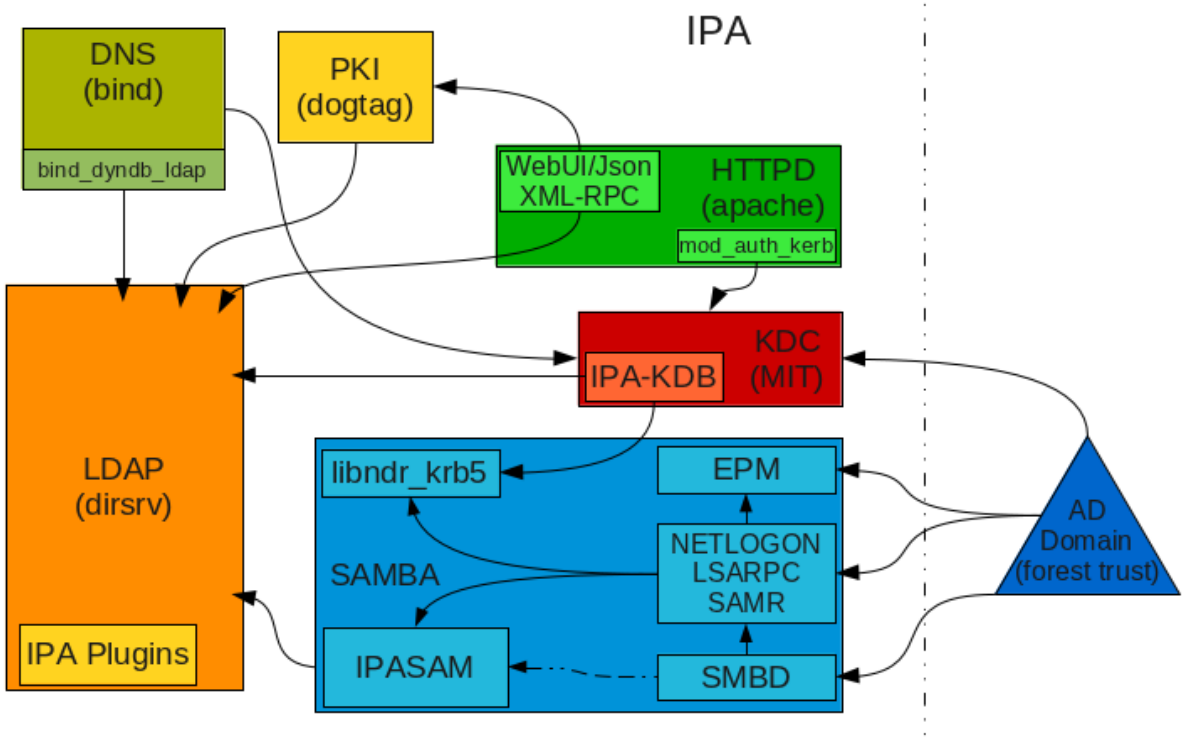
- Can be integrated with Windows as RFC4120-compliant Kerberos realm
- Traditional Kerberos trust management applies:
  - on GNU/Linux side `~/.k5login` should be defined to impersonate users with identities
- Does not scale well for thousands of users and hosts:
  - a foreign realm principal impersonates our realm's user
  - requires additional management of special users to impersonate doubling the management effort
  - mapping has to happen on every single machine. Manually?

Active Directory native cross forest trusts

- Require two Active Directory domains
- AD domain establishes trust with another AD domain via LSA RPC
- AD uses LSA RPC to map incoming principals to SIDs
  - technically: KDC + CLDAP + LSA RPC
  - FreeIPA provides KDC and LDAP, Samba provides LSA RPC

FreeIPA v3 architecture

Full overview is available at
`http://freeipa.org/page/IPAv3_Architecture`



FreeIPA passdb backend

- Expansion of traditional LDAP passdb backend
- New schema objects and attributes to support trusted domain information
- Support for uid/gid ranges for multi-master replicas
- Kerberos principal creation for foreign domain account

FreeIPA KDC backend

- Generates MS PAC information out of LDAP info and add to the ticket
- Allows to accept principals and tickets from a trusted cross forest realm
- Filters out SIDs for both incoming and outgoing trust paths
- Verifies and signs MS PAC coming from a trusted cross forest realm
- Checks authentication paths for non-hierarchical cross-realm

FreeIPA configuration tools

- FreeIPA has command line (CLI) and Web user interfaces
- `ipa trust-add` creates new cross-forest trust
  - CLI operates with Kerberos authentication
  - Request is sent to FreeIPA server via XML-RPC over HTTPS with Kerberos auth
  - FreeIPA uses S4U2Proxy Kerberos feature to allow constrained delegation
  - Samba 4 Python bindings are used to establish trust
    - Code runs under non-privileged account (apache)
    - Uses Kerberos ticket obtained via XML-RPC with the help of mod_kerb_auth
    - Issues Kerberos-authenticated LSA RPC requests to a local smbd
    - Uses AD credentials or shared secret passed via XML-RPC request to talk to AD DC

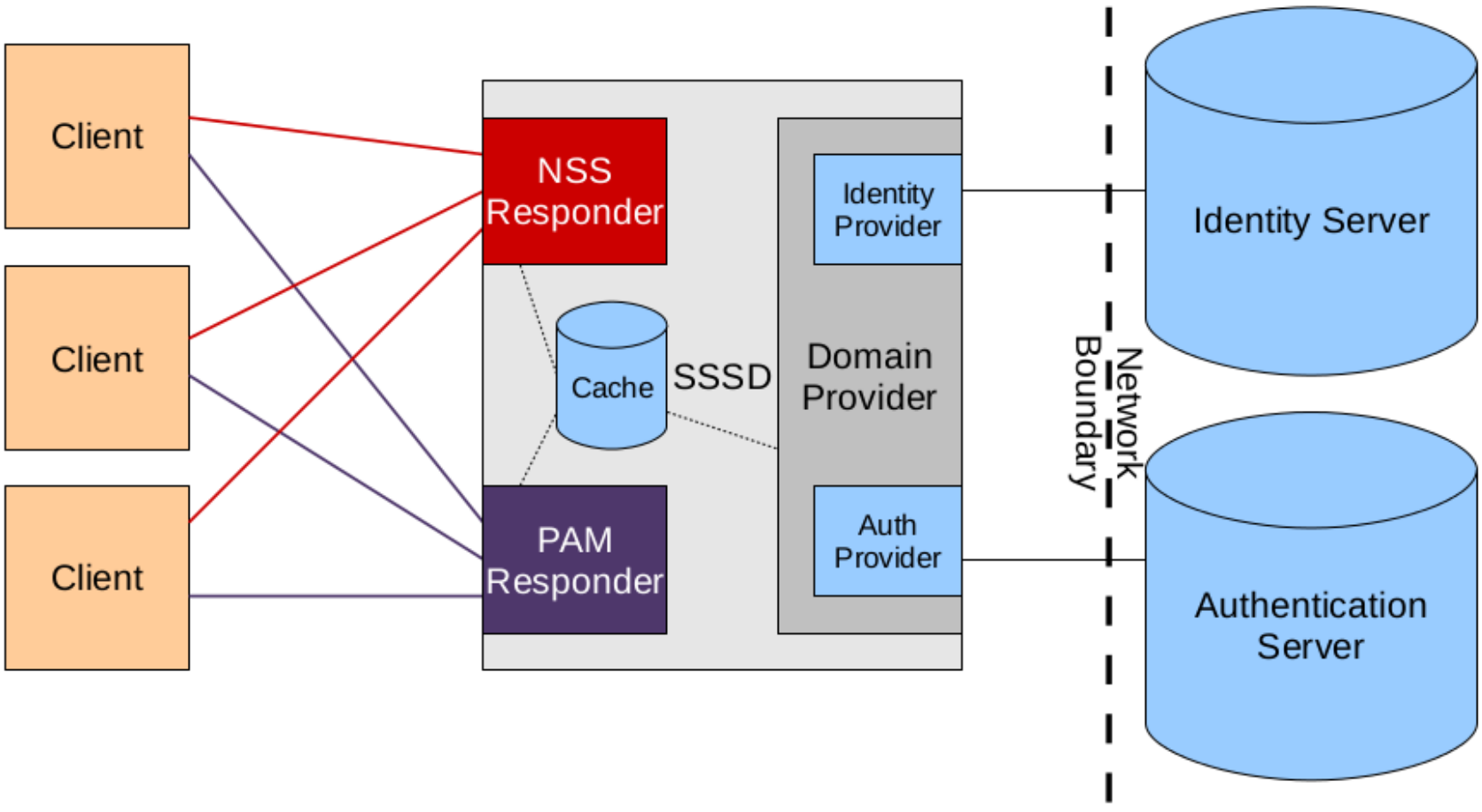
Trusting Active Directory with FreeIPA
A story beyond Samba

Using FreeIPA services with AD credentials

Use of FreeIPA client system with AD cross forest credentials:

- Client system is provisioned with `ipa-client-install`
- SSSD is configured during provisioning to talk to FreeIPA LDAP
- `krb5.conf` is configured to perform mapping of cross forest trusted realm principal to user name 1:1 without removing the realm, e.g. Administrator@AD.SAMBAXP becomes user 'Administrator@ad.sambaxp'
- Manual auth_to_local rules in `krb5.conf` will not be needed with MIT Kerberos 1.12 once SSSD will provide a plugin to fetch these rules from the FreeIPA server

Using FreeIPA services with AD credentials



Using FreeIPA services with AD credentials

On the IPA client, an SSH log-in causes following activity:

- SSH checks if user exists on the system
- SSSD handles the request and sees the user is not local. From IPA client SSSD sends LDAP extended operation to FreeIPA LDAP server.
- Extended operation on LDAP server results in calls to SSSD on the server to perform external domain user/group mapping
  - in FreeIPA 3.0 SSSD uses Winbind on IPA server to resolve the mapping
  - in FreeIPA 3.3 SSSD on IPA server performs the mapping natively against AD DCs
- UID/GID are returned to SSH, GSSAPI is used to log-in that 'local' user
- SSSD uses MS PAC from the Kerberos ticket to fill up groups information, converting SIDs to groups using FreeIPA LDAP extended operation

Trusting Active Directory with FreeIPA
A story beyond Samba

Using FreeIPA services with AD credentials

- SSSD plays crucial role: it forwards ID resolution of AD users to IPA server and performs MS-PAC analysis
- Support for FreeIPA LDAP extended operation was added in SSSD 1.9
- Older versions or machines without SSSD can't use the feature, are we lost?

Supporting ‘legacy clients’

- FreeIPA provides a compatibility LDAP tree, `cn=compat,cn=ipa,cn=sambaxp`
- Compatibility tree re-exports IPA users for systems that cannot be configured against the primary tree
- When cross-forest trust is enabled, the compatibility tree shows AD users
- `ipa-advise` tool can be used to get configuration recipes for legacy clients

Supporting ‘legacy clients’

A request comes from `nss_ldap`, `nslcd`, or old version of SSSD over compatibility tree:

- Internally, search is performed over the primary tree
- If search does not succeed, FreeIPA decodes the request
  - `(uid=`*`username`*`)` is recognized as a user search by name
  - `(uidNumber=`*`number`*`)` is recognized as a user search by ID
  - `(&(cn=`*`groupname`*`)(objectClass=posixGroup))` is recognized as a group search by name
  - `(gidNumber=`*`number`*`)` is recognized as a group search by ID
  - `(memberuid=`*`username`*`)` is recognized as a group search by a member name
- Request is sent to SSSD for lookup, result is inserted into the compatibility tree
- LDAP bind operation is translated into PAM authentication request

Supporting ‘legacy clients’

- Compatibility tree works fine with any Linux or UNIX-like systems that supports PAM and NSS API
- However, there are caveats:
  - AD user name is fully qualified: Administrator@AD.SAMBAXP = 24 characters
  - FreeBSD systems have by default limit for user names of 16 letters even in FreeBSD 9+
  - The limit is kernel compile-time, http://www.freebsd.org/cgi/query-pr.cgi?pr=kern/133926
  - Other *BSD systems have similar issues
  - Kerberos authentication might not be possible unless `krb5.conf` is explicitly configured on the client to recognize AD realms
  - All authentication for legacy logons with AD credentials happens on IPA server

Trusting Active Directory with FreeIPA
A story beyond Samba

Compatibility with Active Directory

FreeIPA 3.3 was tested by a Microsoft team in summer 2013 against Windows Server 2012 build 9200.

- FreeIPA as a trusted realm to AD, passed 106/117 tests
- FreeIPA as a local realm to a Windows client, passed 69/117 tests
- Main issues:
  - FreeIPA cannot return the correct error code (8 cases)
  - FreeIPA does not contain attributes in AD (11 cases)
  - FreeIPA does not support Claims or Compound Identity (11 cases)
  - FreeIPA KDCOption and Flags Issue (7 cases)
  - FreeIPA missing PA-DATA (1 case)
  - FreeIPA configuration issue (2 cases)
  - FreeIPA KDC is case sensitive (1 case)
- We plan to make a new test run with FreeIPA 4.0 in autumn 2014

DEMO

Questions & Answers

- Slides `http://www.samba.org/~ab/`