freeIPA
identity | policy | audit

FreeIPA 3.3

Trust features

Sumit Bose, Alexander Bokovoy

March 2014

FreeIPA and Active Directory

- FreeIPA and Active Directory both provide identity management solutions on top of the Kerberos infrastructure
- FreeIPA AD Trust feature is designed
  - To give Active Directory users access to FreeIPA resources
  - To allow FreeIPA servers and clients to resolve identities of AD users and groups
- FreeIPA AD Trust feature does not require
  - Synchronizing accounts and passwords with AD
  - Installing any software on AD domain controllers

Cross-realm forest trust: FreeIPA and Active Directory

- FreeIPA exposes its own realm as an Active Directory-compatible forest
- Two Active Directory-compatible forests can trust each other
- As result:
  - Active Directory users can access FreeIPA resources
  - FreeIPA servers and clients can resolve identities of AD users and groups
  - Access to FreeIPA is controlled by FreeIPA rules (HBAC, ...) for Active Directory users and groups
  - All AD user and group management stays at AD side

Active Directory → FreeIPA

- FreeIPA Kerberos infrastructure **cannot be joined** to Active Directory forest as a domain, only trusted as an Active Directory-compatible forest
- FreeIPA **provides access** to its own services to Active Domain's users **by trusting** Active Directory Kerberos infrastructure
- **All FreeIPA access control decisions are done on FreeIPA side**
- FreeIPA **uses Kerberos trust** by an Active Directory **to perform** LDAP and DCE RPC **operations required to support identity mapping** of Active Domain's users and groups
- Thus, **two-way cross-forest trust** is required

FreeIPA → Active Directory

- FreeIPA **requires** only **authenticated read-only access** to Global Catalog and LDAP services
  - FreeIPA does not yet implement own Global Catalog service
- **All access control decisions are done on Active Directory side**
- How to limit access by FreeIPA to Active Directory resources?
  - Use access control means provided by Active Directory
  - As no GC service is (yet) available on FreeIPA side, no additional privileges other than authenticated read-only access can be granted to FreeIPA users and services (yet)

freeIPA
identity | policy | audit

Cross-realm forest trust operations overview

Part I: prepare for trust from FreeIPA side

How to set up trust between FreeIPA and AD

- Short plan:
  - Install freeipa-server-trust-ad package
    ```
    # yum install freeipa-server-trust-ad
    ```
  - Enable FreeIPA for Trust
    ```
    # ipa-adtrust-install
    ```
  - `Make DNS arrangements ...`
  - Add Trust to AD
    ```
    # ipa trust-add ...
    ```

Of course there is a number of requirements ...

Before you enable FreeIPA for Trust

- It is highly recommended to use the internal FreeIPA DNS service
- Check installed Samba packages
  - Trust requires samba package, version 4.x, available in Fedora 19+
  - Package freeipa-server-trust-ad has dependency on the correct samba package so yum will install it automatically
- Install freeipa-server-trust-ad on all your FreeIPA masters
  - Any FreeIPA client to serve AD users has to be connected to an FreeIPA master with trust configured
    - Identity mapping is done by FreeIPA servers, thus if the server doesn't know about the trusts, it can't properly handle AD users for FreeIPA client requests

Plan DNS integration

- DNS is the cornerstone for FreeIPA and AD to discover services in the local and remote domains
  - AD assumes it owns the DNS namespace for Kerberos infrastructure which relies on DNS SRV and TXT records
- For trust, different DNS zones for FreeIPA and AD are required
  - Zone delegation is preferred for production environments
  - Conditional forwarding useful for test environments

ipa-adtrust-install

- Needs to be run on each FreeIPA master that has FreeIPA clients going to provide access to AD users
- Sets up Kerberos principals and Samba configuration to serve as a domain controller using SMB protocol
- Configures FreeIPA LDAP server to add SMB-specific attributes to users and groups (ipaNTSecurityIdentifier and other specific attributes)
- Configures FreeIPA LDAP server to provide access to AD users and groups for legacy clients
- Advertises new domain controller through DNS to Active Directory

Sample output of ipa-adtrust-install

Trust

```
Configuring cross-realm trusts for IPA server requires password for user 'admin'
.
This user is a regular system account used for IPA server administration.

admin password:


The following operations may take some minutes to complete.
Please wait until the prompt is returned.

Configuring CIFS
  [1/20]: stopping smbd
  [2/20]: creating samba domain object
Samba domain object already exists
  [3/20]: creating samba config registry
  [4/20]: writing samba config file
  [5/20]: adding cifs Kerberos principal
  [6/20]: check for cifs services defined on other replicas
  [7/20]: adding cifs principal to S4U2Proxy targets
cifs principal already targeted, nothing to do.
  [8/20]: adding admin(group) SIDs
Admin SID already set, nothing to do
Admin group SID already set, nothing to do
  [9/20]: adding RID bases
```

Q&A for ipa-adtrust-install

Q&A for ipa-adtrust-install

- What is a NetBIOS name and why do I need it?
  - It's a short version of the domain name
  - It's used by various protocols needed by Windows to manage trusts
  - It must be unique (like the DNS domain name)
    - NetBIOS name for FreeIPA domain cannot conflict with NetBIOS names for the FreeIPA or AD servers
    - Ipa-adtrust-install will set NetBIOS name for FreeIPA domain only once, no need to specify it on other replicas explicitly
  - It's automatically generated by ipa-adtrust-install
    - NetBIOS name for FreeIPA server is built from its host name
    - NetBIOS name for FreeIPA domain is built from its domain name

Q&A for ipa-adtrust-install

- –no-msdcs option forces to not advertise FreeIPA server to AD via DNS service records
- Do I want to always use –no-msdcs?
  - No!
  - The special service records required for establishing trust and allowing access to FreeIPA resources
  - However, not all servers need to be exposed to AD, follow your chosen topology design
  - If no special service records created by any ipa-adtrust-install run, they must be managed manually on a different DNS server (see man page and command output)

Q&A for ipa-adtrust-install

Trust

- What is a RID base and why do I need two of them?
  - On Windows, users and group are identified by unique Security Identifiers (SID)
  - A SID for a user or a group is built with the domain SID and a Relative ID (RID)
  - A RID is an unsigned 32-bit integer
  - FreeIPA UIDs and GIDs must be translated into SIDs:
    - RID = RID-Base + (ID – BaseID)
  - Since a UID and a GID can have the same value, a second base is needed to avoid conflicts when creating RID for a group or an user that have the same GID and UID
    - Affects UID/GID changes on FreeIPA side or UID/GID mandated from AD POSIX attributes
  - Details: http://www.freeipa.org/page/V3/ID_Ranges

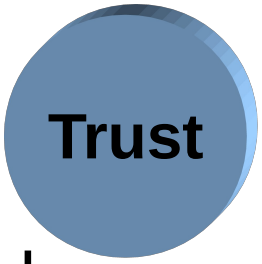
Q&A for ipa-adtrust-install

- Why is the admin password needed?
  - ipa-adtrust-install reconfigures Kerberos KDC and the change requires to obtain new Kerberos ticket afterwards to include MS-PAC information
    - MS-PAC contains group membership information for the user, signed by the Kerberos KDC
  - AD services rely on MS-PAC information in Kerberos tickets to perform authentication and authorization
    - Using MS-PAC greatly reduces network traffic exchange within the domain
  - After ipa-adtrust-install is run, the FreeIPA KDC will add an MS-PAC to the Kerberos tickets
  - The MS-PAC is needed to successfully run 'ipa trust-add ...'
    - ipa-adtrust-install re-inits the admin Kerberos tickets to make sure the admin does not forget to do it

Q&A for ipa-adtrust-install

- –add-sids option generates SIDs for FreeIPA users and groups
- Why is –add-sids not enabled by default
  - –add-sids starts a Directory Server task to add SIDs to all user and group objects
  - The new attributes must be sent to all replica servers
  - With many users, groups and replica servers; the network traffic might lead to temporary performance degradation
  - When adding more FreeIPA servers to serve trust no need to re-run the task

Q&A for ipa-adtrust-install



- How do I start the --add-sids Directory Server task manually?
  - Copy /usr/share/ipa/ipa-sidgen-task-run.ldif
  - Edit nsslapd-basedn and delay:
    - nsslapd-basedn
      - Use value returned by
        'grep basedn /etc/ipa/default.conf | cut -d= -f2-'
    - delay
      - 0 = maximum speed and high CPU and network load
      - Positive integer value, reduced load and speed
  - (as root) ldapmodify -H ldapi://... -f your_copy.ldif

freeIPA
identity | policy | audit

Cross-realm forest trust operations overview

Part II: establish trust between FreeIPA and AD

Establish trust: ipa trust-add ...

- Cross-realm forest trust has two parts:
  - AD side: an account <FreeIPA REALM>$ and trust object for FreeIPA domain have to be created securely
  - FreeIPA side: an account <AD REALM>$ and trust object for AD domain have to be created securely
- “Securely” means
  - Create an account and an object on AD side using AD administrator credentials using MS-RPC protocol
  - Create an account and an object on FreeIPA side using FreeIPA administrator credentials using MS-RPC protocol
  - Trigger verification process between AD and FreeIPA

Establish trust: ipa trust-add ...

- “Verification process” means:
  - FreeIPA server initiates verification on AD domain controller side using MS-RPC protocol stack
  - AD DC checks FreeIPA LDAP for FreeIPA domain information, then
  - AD DC talks to FreeIPA DC using MS-RPC protocol using credentials of the <AD REALM>$ account, then
  - FreeIPA DC connects to AD DC using MS-RPC protocol using credentials of the <FreeIPA REALM>$ account, then
  - AD DC marks the trust object for FreeIPA domain as verified
- Finally, AD DC trusts Kerberos tickets issued by FreeIPA

Establish trust: ipa trust-add ...


```
[root@rh7-01 ~]# kinit admin
Password for admin@IPA7.TEST:
[root@rh7-01 ~]# ipa trust-add ad.test --admin Administrator --password
Active directory domain administrator's password:
----------------------------------------------
Added Active Directory trust for realm "ad.test"
----------------------------------------------
  Realm name: ad.test
  Domain NetBIOS name: AD
  Domain Security Identifier: S-1-5-21-2275361654-3393353068-3720134936
  SID blacklist incoming: S-1-0, S-1-1, S-1-2, S-1-3, S-1-5-1, S-1-5-2, S-1-5-3,
                          S-1-5-4, S-1-5-5, S-1-5-6, S-1-5-7, S-1-5-8, S-1-5-9,
                          S-1-5-10, S-1-5-11, S-1-5-12, S-1-5-13, S-1-5-14,
                          S-1-5-15, S-1-5-16, S-1-5-17, S-1-5-18, S-1-5-19,
                          S-1-5-20
  SID blacklist outgoing: S-1-0, S-1-1, S-1-2, S-1-3, S-1-5-1, S-1-5-2, S-1-5-3,
                          S-1-5-4, S-1-5-5, S-1-5-6, S-1-5-7, S-1-5-8, S-1-5-9,
                          S-1-5-10, S-1-5-11, S-1-5-12, S-1-5-13, S-1-5-14,
                          S-1-5-15, S-1-5-16, S-1-5-17, S-1-5-18, S-1-5-19,
                          S-1-5-20
  Trust direction: Two-way trust
  Trust type: Active Directory domain
  Trust status: Established and verified
[root@rh7-01 ~]# _
```

freeIPA
identity | policy | audit

Q&A for ipa trust-add

Q&A for ipa trust-add

- What credentials are required to establish trust?
  - Either FreeIPA side or AD side can drive the process
- If process is driven from FreeIPA side:
  - AD user from Domain Administrators group of the root domain of an AD forest to be specified as `–admin` option to `'ipa trust-add'` and `–password` option should be given to specify the admin's password

Q&A for ipa trust-add

- If process is driven from AD side:
  - AD Administrator has to pre-create the trust
  - The –trust-secret option should be used to create only the FreeIPA part of the trust and verify it
- A demo of the join process is available at http://www.freeipa.org/page/File:Trust-ad-demo-shared-secret.gif
- If AD DC finds a conflict between AD and FreeIPA DNS domains in the forest topology, the trust will be put under quarantine
  - Use a name suffix routing tab in AD to fix it (slide 29)

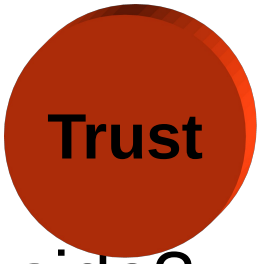
Q&A for ipa trust-add

- What is an ID range?
  - ID ranges are used:
    - to reserve POSIX ID for users and groups from a specific domain
    - to map users and groups from AD domains to a POSIX IDs
  - `ipa trust-add` will find a suitable range automatically
  - Must be only used manually if a specific range should be used, e.g. when migrating from a different product

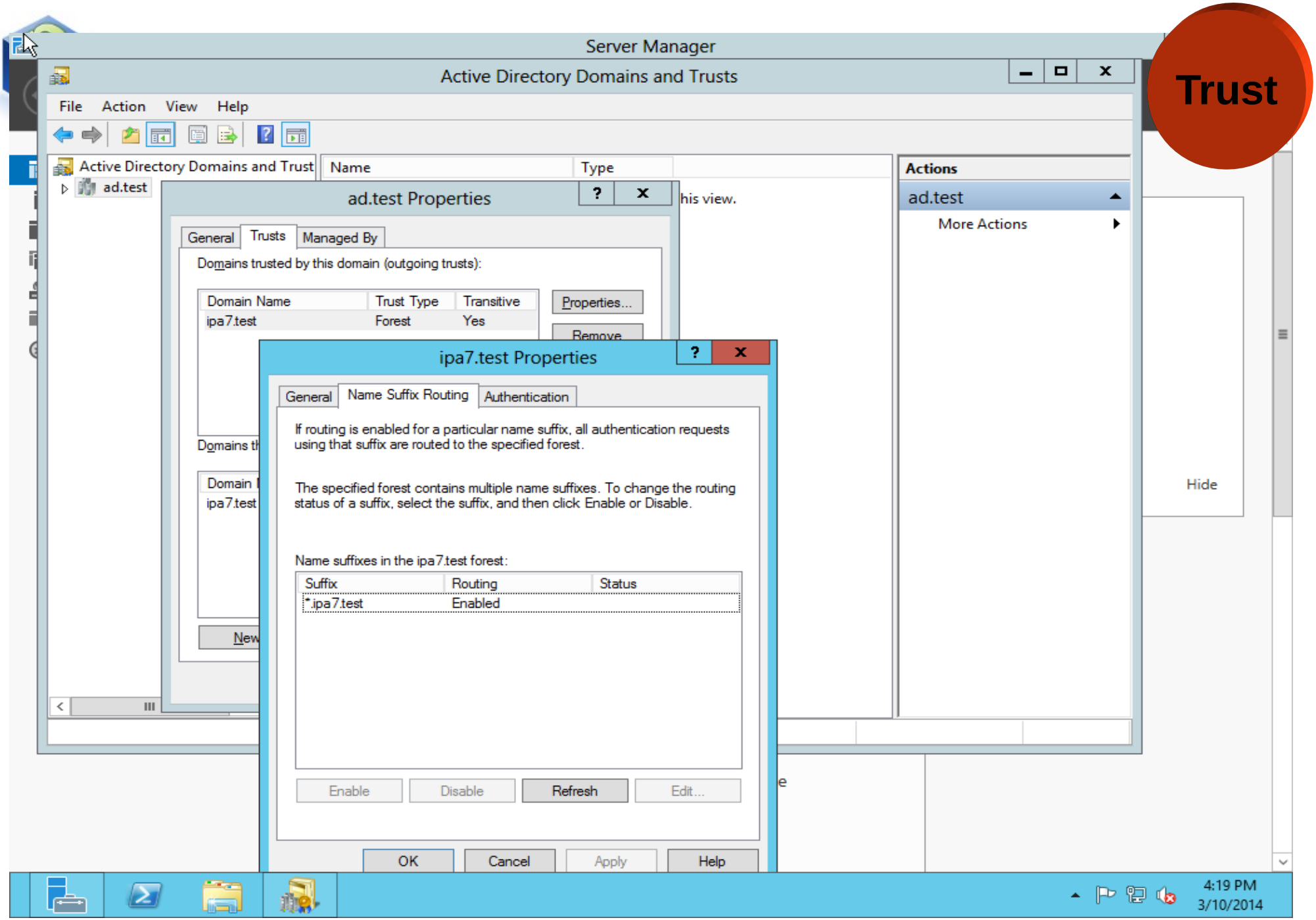
Q&A for ipa trust-add

- What if POSIX ID ranges are already defined on AD side?
  - `ipa trust-add` will attempt to detect this case
    - Can be forced by `–range-type=ipa-ad-trust-posix`
    - AD admins should be in charge to define non-conflicting ranges
  - `ipa trust-add` will create corresponding range for whole forest

Q&A for ipa trust-add



- Do I have to validate the trust from the Windows side?
  - No, validation is done by ipa trust-add
  - However, when AD part of the trust was created in advance, AD may put the trust into quarantine until name suffix routing information is updated to enable the trust
    - See a screen capture on the next slide


Trust
Server Manager
Active Directory Domains and Trusts
File Action View Help
Active Directory Domains and Trust
ad.test
Name
Type
Actions
ad.test
More Actions
ad.test Properties
General Trusts Managed By
Domains trusted by this domain (outgoing trusts):
Domain Name
Trust Type
Transitive
ipa7.test
Forest
Yes
Properties...
Remove
ipa7.test Properties
General Name Suffix Routing Authentication
If routing is enabled for a particular name suffix, all authentication requests using that suffix are routed to the specified forest.
The specified forest contains multiple name suffixes. To change the routing status of a suffix, select the suffix, and then click Enable or Disable.
Name suffixes in the ipa7.test forest:
Suffix
Routing
Status
*.ipa7.test
Enabled
Enable
Disable
Refresh
Edit...
OK
Cancel
Apply
Help
Hide
4:19 PM
3/10/2014

Q&A for ipa trust-add

- `ipa trust-add` was successful, but `getent passwd` does not return external users what do I miss?
  - SSSD is used behind the scenes on the FreeIPA server to lookup up users in trusted AD domains
    - SSSD on FreeIPA clients will forward resolution requests to FreeIPA servers through FreeIPA LDAP server plugin
  - SSSD caches Kerberos ticket it uses to talk to LDAP servers
  - Restart SSSD to force it to refresh the cached view

freeIPA
identity | policy | audit

Cross-realm forest trust operations overview

Part III: manage trust topology

Trust topology: FreeIPA → AD

- FreeIPA realm may span multiple DNS domains
- FreeIPA exposes knowledge about FreeIPA DNS domains to AD with the help of `ipa realmdomains`
  - ipa realmdomains-show
  - ipa realmdomains-mod
- Any new DNS domain managed by FreeIPA will be automatically reflected in realmdomains.
- AD side needs manual refresh, see screenshot on slide 29

Trust topology: AD → FreeIPA

- AD forest may contain multiple AD domains
  - Top domain is called *forest root domain*
  - Other domains are *sub-domains* or *child domains*
- Forest topology can be updated with `ipa trust-fetch-domains`
  - `ipa trust-add` runs it automatically but further updates should be done manually
- FreeIPA manages knowledge about AD domains with the help of `ipa trustdomain`
    - ipa trustdomain-find / -enable / -disable
- Domains other than the AD forest root can be disabled for the trust view. Their users cannot then access FreeIPA resources

freeIPA
identity | policy | audit

Q&A for managing trust topology

Q&A for trust topology

- Should I add additional DNS domains to realmdomains manually?
  - No, DNS management commands in FreeIPA already do that every time new DNS zone is added or deleted
  - In case DNS is not managed by FreeIPA, `ipa realmdomains-mod` can be used to modify the list of the associated DNS domains
- Why FreeIPA presents AD forest's domains as a flat list, they could be arranged in hierarchy in Active Directory?
  - Cross-realm forest trust only requires that Kerberos KDC of forest root domains trust each other. Authentication always goes through the forest root domain's Kerberos KDC.
  - Information about sub-domains is only used for identity mapping and access control purposes. Their relationship topology is not important for FreeIPA.

Q&A for trust topology

- Why a user from a sub-domain cannot login with putty to an FreeIPA client? The sub-domain is listed as enabled by `ipa trustdomain-find`.
  - Rules for enabling access to sub-domain uses are the same as for the forest root domain. See slide 47 “Q&A SSO with putty from Windows” in the next part and add auth_to_local rule specific for the sub-domain.
- Why I cannot disable forest root domain?
  - Disabling forest root domain is equivalent to removing the trust

freeIPA
identity | policy | audit

Cross-realm forest trust operations overview

Part IV: using FreeIPA services via trust

Q&A: Adding external user to local groups

Adding users from trusted domains to local groups

- Create a group for external users
  - ipa group-add –external gr_ext
- Add external users or groups
  - ipa group-add-member –external 'ADDOM\user' gr_ext
- Add group for external users to a local group
  - ipa group-add-member –groups=gr_ext local_group
- If the local group is used e.g. in a HBAC rule the rule applies to the remote users as well
- If the local group is a POSIX group, the remote user will be a member of this POSIX group

Q&A External users and local groups

- Why do I need a special group for external users?
  - Only objects managed by FreeIPA can be a member of an FreeIPA group
  - External users and groups must first be associated with an object managed by IdM
  - Groups created with `ipa group-add –external` provide an object to satisfy this requirement

Q&A External users and local groups

- Can I add external users before I call `ipa trust-add`?
  - No!
  - The given user or group name or the SID of the external object is checked on the remote AD server
  - Since AD in general does not allow anonymous access, this can only be done when the trust is established

Q&A External users and local groups

- What are the strange S-1-5-21-... strings listed as 'External member' by ipa group-show?
  - They are the SIDs of the external users and groups
  - SIDs are unique and cannot be changed and therefore used to reference an external object
  - If there is a network issue at the time `ipa group-show` is called, SIDs may be left not translated in the output of the command

Q&A External users and local groups

- Why does '`getent group local_group`' not show external user ADDOM\XYZ?
  - The full group membership of an external user is only evaluated when the user logs in
  - Full group membership of an external user is not stored on the server but only cached on the client
  - The full group membership of an external user can be found in the PAC of the Kerberos ticket
  - Sorry, currently there is no tool which can display the PAC in tickets stored in the credential cache

Q&A: SSH access with users from trusted domains

SSH access for users from trusted domains

- Putty is a widely used SSH client for Windows
  - http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
- AD users can also login to IdM clients directly at console or from any other IdM client via OpenSSH

Q&A SSO with putty from Windows

- SSO with putty does not work; what am I doing wrong?
  - The fully qualified host name must be used
    - To find the matching Kerberos service ticket the host name entered in putty's 'Host name (or IP address)' field is used
  - User principal name (UPN) or Down-Level Logon Name should be used
    - Both standard Windows name types[1] are supported
      - UPN: username@domain.name
      - Down-Level Logon Name: DOMAIN\username
- Remember to save the session!

[1] http://msdn.microsoft.com/de-de/library/windows/desktop/aa380525(v=vs.85).aspx

Q&A SSO with putty from Windows

- SSO with putty does not work; what am I doing wrong?
  - Destination host must be able to map Kerberos UPN to POSIX user name
    - Users can create .k5login file with the UPN in their home-directory, i.e.
      - Log in with password first
      - Create .k5login file
      - Now log in with SSO
    - Admin can add auth_to_local mapping in krb5.conf for each AD domain (separate auth_to_local lines per each domain in AD forest):

```
[realms]
  IPA.DOMAIN = { ...
   auth_to_local = RULE:[1:$1@$0](^.*@AD_DOMAIN$)s/@AD_DOMAIN/@ad_domain/
   auth_to_local = DEFAULT
  }
```

Q&A SSO with putty from Windows

- SSO with putty does not work; what am I doing wrong?
  - If HBAC is used in the IdM domain trusted users must be added to HBAC rules
    - Add user to a group for external users
    - Add this group to a local group
    - Use the local group in a HBAC rule to allow access

Q&A SSO with putty from Windows

- My credentials are not forwarded/delegated; what is missing?
  - Putty's Checkbox 'Allow GSSAPI credential delegation' must be checked
  - Windows requires the ok_as_delegate Kerberos flag in the service ticket to delegate credentials
    - ipa service-mod –ok-as-delegate=true \ host/destinationhost.domain@REALM
  - Flags can be checked with klist

POSIX attributes defined in Active Directory

- POSIX attributes can be retrieved from AD side
  - UID and GID can be defined on AD side
  - IdM uses special ID range type '*ipa-ad-trust-posix*' to define ranges for users with POSIX attributes coming from Active Directory
  - For users from ID range with type '*ipa-ad-trust-posix*' SSSD will pick up POSIX attributes from Active Directory
  - Home directory will always be overridden by **SSSD on FreeIPA client**, controlled by sssd.conf
    - `subdomain_homedir` option
    - Default value: /home/%d/%u
  - AD user's shell will be set to default one by **SSSD on FreeIPA server or client**, controlled by sssd.conf
    - default_shell option
    - Default value: <empty>
      - Empty value means using default IdM shell, defined with '`ipa config-mod --defaultshell=..`' command
        - Default IdM shell is /bin/sh
  - Thus, user's shell can be overridden for all AD users per-FreeIPA client

Q&A: Legacy clients access from trusted domains

Legacy clients access from trusted domains

- Legacy clients are those machines which don't have SSSD 1.9.2 or later installed
  - SSSD 1.9.2 or later knows how to handle trusted domains when configured with 'ipa' identity provider
  - Full trusts feature set is available with SSSD 1.11
- Separate training material is available for details of legacy clients configuration, see next slide for short summary

Legacy clients access from trusted domains

- If Schema Compatibility plugin was enabled with `ipa-adtrust-install –enable-compat`
  - AD users can be searched over cn=compat,$SUFFIX tree by nss_ldap/nss-pam-ldapd/Solaris/AIX/etc
  - AD users can authenticate against IdM LDAP server using uid=<user>,cn=users,cn=compat,$SUFFIX
- Use `ipa-advise` tool to generate recipes how to configure legacy clients to authenticate and fetch identity information from the compatibility tree. Both IdM and AD users will be available through the single tree.
- ipa-advise tool has advices for RHEL5, RHEL6, older Fedora, generic Linux, and FreeBSD already.


freeIPA
identity | policy | audit