Cross-realm trusts
with FreeIPA v3

Alexander Bokovoy, Andreas Scheider
Alexander Bokovoy
about:me
- Member of Samba Team since 2003
- Principal Software Engineer, Red Hat
  - FreeIPA project

Andreas Schneider
about:me
- Member of Samba Team since 2010
- Software Engineer, Red Hat
  - Samba support and development
Cross-realm trusts with Active Directory

- Active Directory
  - CIFS authentication protocols used by Windows 2000 and above
  - Kerberos protocol
  - LDAP storage and protocol, CLDAP protocol
  - PAC kerberos extension
Cross-realm trusts with Active Directory

- PAC: Privilege Attribute Certificate
  - Extension to Kerberos to convey CIFS information
  - Contains
    - Authorization data (security identifiers and relative identifiers of group membership, etc)
    - User profile information (home directory, logon script, etc)
    - Service for User data
    - Password credentials (for smartcards)
Cross-realm trusts with Active Directory

- **PAC: Privilege Attribute Certificate**
  - Relatively well described in [MS-PAC]
  - Allows to cache and pass through important account details for kerberized services
  - Usually not inspected and not expected by traditional Kerberos-based services/applications
  - Tickets with PAC may grow large, up to 65KiB
FreeIPA v3

- FreeIPA v3 introduces few extensions:
  - New Kerberos database back end
  - Support for filling in PAC structure
  - CLDAP responder to complement LDAP queries
  - New Samba SAM back end
FreeIPA v3

- FreeIPA v3 relies on “merged” Samba build
  - Samba 3 daemon with external RPC processing
    - End-point mapper
    - LSA SS, LSA SD, LSA RPC
    - SAMR, NETLOGON
  - IPA SAM back end to use LDAP/Kerberos combined information from FreeIPA
  - Samba 4 client libraries and Python bindings for trust management
FreeIPA v3

- net rpc trust
  - Samba 3 'net' utility
  - Connects to remote DC and issues LSA calls to setup the trust part
  - Modifies Samba databases directly
    - Requires root access
    - Can't be run within FreeIPA WSGI process
  - Complex command line arguments
FreeIPA v3

Challenges

- FreeIPA management process runs under unprivileged user
- Uses delegated Kerberos credentials to access managed services

Thus, trust relationship should be queried and managed via remote CIFS calls

- Use of Samba client libraries
  - Python bindings to Samba 4
FreeIPA v3 Trust User Experience

- 'net rpc trust create'
  - Asks for 6 specialized parameters including those not exposed in Windows UI
  - Windows Admins usually don’t give up their credentials easily
  - Creates the trust in “one shot”

- Quest to simplify the experience
  - Trust has two halves:
    - Trust information in local realm
    - Trust information in remote realm
    - Can be set independently
FreeIPA v3 Trust User Experience

- Quest to simplify the experience
  - Trust has two halves:
    - Trust information in local realm
    - Trust information in remote realm
    - Can be set independently
    - Require verification to finalize
  - All trust information is possible to autodiscover using LSA and CLDAP requests
Trust setup with existing admin credentials

- `ipa trust-add-ad --server=winda.ad.local --realm-admin=AD\Administrator`
  - My credentials (kerberos ticket)
  - AD server
  - AD admin creds (will be asked for the password)
  - Miss shared trust secret

- Resulted actions:
  - Generate shared secret
  - Discover AD domain/realm with AD admin creds
  - Setup local trust part
  - Setup remote trust part
  - Verify remote trust working
  - Display info, result of verification, and instructions how to use the trust
Local trust setup before remote trust is done

- `ipa trust-add-ad --server=winda.ad.local`
  - My credentials (kerberos ticket)
  - AD server
  - Miss AD admin creds
  - Miss shared trust secret

- Resulted actions:
  - Generate shared trust secret
  - Discover anonymously AD domain/realm
  - Set up local trust part
  - Display instructions and info for windows admin to setup the remote trust part
Local trust setup after remote trust is done

- `ipa trust-add-ad --server=winda.ad.local --shared-secret=ABCD`
  - My credentials (kerberos ticket)
  - AD server
  - Shared trust secret
  - Miss AD admin creds

- Resulted actions:
  - Discover anonymously AD domain/realm
  - Setup local trust part
  - Verify remote trust working
  - Display info and result of verification
Challenges and issues

- Samba 4 python bindings
  - Largely shaped by use in setting up Samba 4 DC
  - Auto-generated for majority of DCE RPC calls
  - Common processing
  - Very brief error reporting (RuntimeError)
  - Crash immediately if used incorrectly

- Samba 4 Credentials library
  - Makes wild assumptions of ways to authenticate
    - Tries to discover based on environment, does not work for cases like “running within WSGI process with multiple Kerberos credentials caches”
Challenges and issues

- Samba 3
  - Not everything in new RPCs for Active Directory is supported
    - Can't complete Windows-initiated trust request due to lack of DRSU API
    - Samba 4 is there for a reason!

- Samba 4
  - Samba relies on Heimdal a lot
  - Work to make sure MIT Kerberos library is usable is being done. A must for distribution integration
  - Will require fairly new MIT Kerberos (1.9)
Questions?