FREEIPA INSTALLATION USING ANSIBLE-FREEIPA

FOSDEM - 2018-02-03

Thomas Wörner
Senior Software Engineer, Red Hat Inc.

`https://github.com/freeipa/ansible-freeipa/`

AGENDA

- Project goals
- IPA installers vs. ansible-freeipa
- IPA client installation steps
- Enrollment workflow with ipa-client-install vs. with ansible-freeipa
- IPA client OTP use case
- IPA client domain configuration with ipa-client-install vs. with ansible-freeipa
- IPA server installation steps
- Examples of Ansible inventory files and playbooks

PROJECT GOALS

- Allow automation of FreeIPA installations and configuration using ansible-freeipa
- Same results using normal FreeIPA installers or ansible-freeipa
  - ansible-freeipa can provide additional features
- Provide Ansible roles and modules for server, client and replica installations
  - The replica installation is still work in progress and not part of the repository yet
- Support FreeIPA 4.5+ for ipaserver, ipareplica and ipaclient roles

FREEIPA INSTALLER SCRIPTS VS. ANSIBLE-FREEIPA

INSTALLATION USING FREEIPA INSTALLERS

- Log in to every machine, start installation process manually
- Use either principal/password or keytab
- Wait till installation is done

INSTALLATION USING ANSIBLE-FREEIPA

- Simple installation on more than one machine
- One configuration file (inventory file) per domain or realm
- One place for configuration options
- Simple use of OTP for client installation and update, more secure: Admin password not transferred to the clients
- Advanced auto detection for clients
- Repair of broken client configurations with one known limitation:
    - Missing /etc/krb5.keytab

FREEIPA CLIENT INSTALLATION STEPS

- Domain discovery and validation of parameters
- Time synchronization (ntp, chrony)
- IPA enrollment (Creation of host entry and keytab)
- SSSD, PAM, NSS configuration
- Kerberos client configuration
- PKI configuration
- DNS configuration

CLIENT CONFIGURATION WITH ANSIBLE-FREEIPA

- Full autodiscovery: No need to provide domain or realm
  - Using DNS SRV/TXT records for ldap and kerberos
- Autodiscovery of IPA servers: Provide IPA domain
- Enhanced discovery: Provide only server
- No discovery: Provide server and domain
- Realm is usually derived from upper-cased name of the IPA domain, or can be forced to a different value
- Supported enrollment types
  - OTP
  - Admin principal and password
  - Existing host keytab

CLIENT INVENTORY FILE

```
# Example minimal inventory file using full auto-detection
[ipaclients]
ipaclient.ipadomain.com

# ipaclient_password can be provided by a Vault-protected file
```

ipaservers	Group of IPA server hostnames
ipaclients	Group of IPA client hostnames
ipaadmin_keytab	The path to the admin keytab used for alternative authentication
ipaadmin_password	The password for the kerberos admin principal
ipaadmin_principal	The authorized kerberos principal used to join the IPA realm
ipaclient_domain	The primary DNS domain of an existing IPA deployment
ipaclient_realm	The Kerberos realm of an existing IPA deployment
ipaclient_keytab	The path to a backed-up host keytab from previous enrollment
ipaclient_force_join	Set force_join to yes to join the host even if it is already enrolled
ipaclient_use_otp	Generate a one-time-password
ipaclient_allow_repair	Allow repair of already joined hosts
ipaclient_kinit_attempts	Repeat the request for host Kerberos ticket
ipaclient_ntp	Set to no to not configure and enable NTP
ipaclient_mkhomedir	Create users home dir

CLIENT PLAYBOOKS

install-client.yml

```
---
- name: Playbook to configure IPA clients with username/password
  hosts: ipaclients
  become: true
  vars_files:
  - playbook_sensitive_data.yml

  roles:
  - role: ipaclient
    state: present
```

uninstall-client.yml

```
---
- name: Playbook to configure IPA clients with username/password
  hosts: ipaclients
  become: true
  vars_files:
  - playbook_sensitive_data.yml

  roles:
  - role: ipaclient
    state: absent
```

IPA SERVER INSTALLATION STEPS

- Domain discovery and validation of parameters
- (Configure firewall)
- Time synchronization and configuration (ntpd)
- Directory server configuration (dirsrv)
- Kerberos configuration (krb5kdc, kadmin)
- Certificate Server configuration (pki-tomcatd)
- Further directory server configuration (dirsrv)
- OTPD configuration (ipa-otpd)
- Custodia configuration (ipa-custodia)
- HTTP configuration (httpd)
- Kerberos KDC configuration (krb5kdc)
- KRA (Key Recovery Authority) configuration
- DNS configuration (named)
- AD trust configuration (smb, winbind)
- Client configuration on master
- Enable IPA service

SERVER INVENTORY FILE

```
# Example minimal server inventory file
[ipaserver]
ipaserver.ipadomain.com

[ipaserver:vars]
ipaserver_domain=ipadomain.com
ipaserver_realm=IPADOMAIN.COM
# Passwords can be provided by a Vault-protected file
ipaadmin_password=SomePassword1
ipadm_password=SomePassword2
```

ipaserver	Group with IPA server hostname
ipaadmin_password	The password for the kerberos admin principal
ipaserver_domain	The primary DNS domain for the IPA deployment
ipaserver_realm	The Kerberos realm for the IPA deployment
ipaserver_setup_kra	Install and configure a KRA on this server
ipaserver_setup_dns	Configure an integrated DNS server
ipaserver_setup_adtrust	Configure AD Trust capability
ipaserver_auto_forwarders	Add DNS forwarders configured in /etc/resolv.conf
ipaserver_no_reverse	Do not create reverse DNS zone
ipaclient_no_ntp	Set to no to not configure and enable NTP
ipaclient_mkhomedir	Create users home dir

(excerpt)

SERVER PLAYBOOKS

install-server.yml

```yaml
---
- name: Playbook to configure IPA server with username/password
  hosts: ipaserver
  become: true

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

uninstall-server.yml

```yaml
---
- name: Playbook to configure IPA clients with username/password
  hosts: ipaserver
  become: true

  roles:
  - role: ipaserver
    state: absent
```

CLUSTER INVENTORY FILE

```
[ipaserver]
ipaserver.ipadomain.local

[ipaserver:vars]
ipadm_password=SomePassword123
#ipaserver_setup_dns=yes
#ipaserver_auto_forwarders=yes

[ipaclients]
ipaclient1.ipadomain.local
ipaclient2.ipadomain.local
ipaclient3.ipadomain.local

[ipaclients:vars]
#ipaclient_use_otp=yes
ipaclient_allow_repair=yes

[ipa:children]
ipaserver
ipaclients

[ipa:vars]
ipaadmin_password=SomePassword456
ipaserver_domain=ipadomain.local
ipaserver_realm=IPADOMAIN.LOCAL
```

CLUSTER PLAYBOOKS (1)

install-cluster.yml

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

  roles:
  - role: ipaserver
    state: present

- name: Install IPA clients
  hosts: ipaclients
  become: true

  roles:
  - role: ipaclient
    state: present
```

Note: Please remember to register the client IP addresses and names if DNS will be setup in the IPA server. This needs to be done before the clients are enrolled.

CLUSTER PLAYBOOKS (2)

uninstall-cluster.yml

```
---
- name: Uninstall IPA clients
  hosts: ipaclients
  become: true

  roles:
  - role: ipaclient
    state: absent

- name: Uninstall IPA servers
  hosts: ipaserver
  become: true

  roles:
  - role: ipaserver
    state: absent
```

Q/A

THANK YOU