

ANSIBLE

Introduction to Ansible Collections

Ganesh Nalawade
Principal Software Engineer
Ansible Engineering

Agenda

- What's a Collection?
- Why Collections?
- Deep dive with Collections
- Collection demo
- Ansible galaxy for Collections
- Reference for Collection

Who am I

Ganesh Nalawade

- * Principal Software Engineer at Ansible by Red Hat
- * Work primarily as upstream developer in Ansible Networking
- * Worked extensively on Network management plane developing software for on/box automation and programmability infra.
- * Co Organiser for Ansible meetup group in Pune

- Bundle of Ansible content
 - Any kind of plugins
 - modules
 - module_utils
 - roles
 - playbooks (TBD)
- With a namespace + name

- Difficult distribution of non-role content
- Plugin/role name collisions
- Difficult to do code-sharing for most plugins

- [ansible-network.network-engine](#)

- Multiple roles with dependencies
 - Maintain version dependencies across roles and Ansible version
- Roles already distribute plugins but the problem is
 - Requires role to be added in playbooks “entered” to use plugin content
 - Plugins are not “first-class citizens”
 - Versioning/maintenance issues

- Collection may contain 1-N:
 - Roles/Modules/Plugins
 - Playbooks (TBD)
- Collection is the unit of distribution and versioning
- Installable:
 - globally
 - per-user
 - content-adjacent (ie per-project)

- Default paths (searched in this order)
 - (current playbooks)/collections
 - ~/.ansible/collections
 - /usr/share/ansible/collections
- ansible-galaxy can be used to create/install/distribute collections
 - From Ansible 2.9 onwards (tech preview)

- Sample collection layout:
 - [ansible-security.ibm_qradar](#)
 - [ansible-network.arista.eos](#)

- Ansible has a flat plugins namespace
- Role1 and Role2 both distribute my_module
 - Which one will get used?
 - Can't use both of them in same run?
 - If ansible core later includes a my_module?
 - What if project has in library/?
 - What if the system has ib /usr/share/ansible?
 - Issue exists for roles and all modules/plugins

- Collections have a namespace and name
 - For distribution, namespace == Galaxy user/org
- Content is collections is accessed by namespace
 - mynamespace.mycollection.my_module
 - f5.bigip.provisioning_role
- Built-in collections
 - ansible.builtin
 - ansible.legacy

- Collections have a namespace and name
 - For distribution, namespace == Galaxy user/org
- Content in collections is accessed by namespace
 - mynamespace.mycollection.my_module
 - f5.bigip.provisioning_role
- Built-in collections
 - ansible.builtin (stuff shipped with ansible)
 - ansible.legacy (mimics old lookup behaviour)

- **hosts**: somehosts
- tasks**:
 - `myns.mycollection.athing`
 - `ansible.builtin.ping` *# use only the ping packaged in core*
 - `ansible.legacy.ping` *# use core or library(etc)/ping.py*
`when: thing | myns.mycollection.filter == 42`
 - **ping**: *# still works, == ansible.legacy.ping:*

- Ordered list of collections for unqualified name
- `ansible.legacy` auto-appended
 - if `ansible.builtin` or `ansible.legacy` isn't explicitly listed
 - eases adding collection search to existing plays
- settable on `plays/blocks/tasks`

- `module_utils` in roles only works for modules
 - inventory plugins can't use `module_utils` code
 - unless you're using ansible shipped code

- Entire collection is a Python namespace package
- All installed collection content is accessible
 - even across collections!
 - `ansible_collections` root Python namespaces
 - `ansible_collections.mynamespace.mycollection`

DEMO

- [Collection support with ansible-galaxy \(2.9\)](#)

Contributing to Ansible

- [Community groups](#)
- [IRC channels](#)
- Google Groups:
<https://groups.google.com/forum/#!forum/ansible-project>
<https://groups.google.com/forum/#!forum/ansible-devel>
- [Ansible galaxy](#)

Thank You

Github/IRC: @ganeshrn