You’re Doing Ansible Roles All Wrong

Tim Appnel
Product Manager, Ansible
GitHub: tima
Matrix: @tima:ansible.im

ANSIBLE

Agenda

- A brief history of roles in Ansible
- Why Ansible roles?
- The state of Ansible roles
- How You’re Doing Ansible Roles All Wrong
- (Reverse) Q&A
  - How to improve and enhance role-based content today and tomorrow?

Not Your Typical “Best” Practices Talk

Ansible Community of Practice (CoP)
“Good” Practices

The Zen of Ansible

Brief History of Roles in Ansible

- Introduced in November 2013 with Ansible version 1.4
- Reusable building blocks for automation
- Encapsulate tasks, variables, handlers, templates, and files
- Promote portability and organization
- Use human-readable YAML

Why Ansible Roles

- Promote consistency and standardization
- Scale automation across large environments
- Foster collaboration and best practices
- Shareable, versioned, and support dependencies (sort of)
- Reusable by different teams
- Crucial for efficient and effective automation solutions

The State of Ansible Roles

- Initially part of playbook projects
- Basic support from Ansible Galaxy/GitHub
- Collections introduced in Ansible 2.8 (Dec 2018), bundling modules, plugins, roles with namespaces
- Continued use of standalone roles
- Extensive use in large deployments (hundreds, even thousands)
- Roles lag behind modules/plugins in terms of functionality

How You’re Doing Ansible Roles All Wrong

Roles Should Be In Collections

Advantages of Roles in Collections

- Share plugins between roles (plugins in root of collection)
- Improved namespace
- Group roles in one distribution package (easier install)
- Devtools support for creating new collections

Advantages of Roles in Collections

- Python namespace for collections
- Collections included in `ansible` community package
- Future Ansible content development is collection-based
- Easier maintenance of smaller, purpose-built roles
- Ansible Development Environment (ADE) for dependency management

Advantages of Roles in Collections

- AAP Enhancements
  - Collections added to execution environments (EEs), auto-install of Python dependencies
  - Storage of collections in automation hub/secure supply chain
  - Colocate role with EDA content

Documentation

- Meaningful README for Every Role:
  - Summary of purpose and actions
  - Facts needed upon invocation
  - Facts set post execution
  - Rollback capabilities
  - Check mode capabilities
  - Examples for common and not so obvious invocations
- License File: Define terms for use, modification, and distribution
- Collection README: Concise descriptions of each role's purpose

Other Overlooked Ansible Role Practices

- Always Define an Argument Spec:
  - Ensure inputs are well-defined and communicated effectively
  - Fail fast (and more friendly)
- Use Minimal Well-Defined Entry Points:
  - Just rely on include_role and avoid complex dispatching logic with task_from etc.
- Implement Sophisticated Logic in Collection Modules/Plugins:
  - Avoid programming in roles (YAML suck for programming)

Shortcomings in Ansible Roles Today

- Poor Documentation:
  - Difficult to find and understand, if it exists at all
- Fuzzy/Complex Interface:
  - Multiple entry points, confusing syntax
- Portability & Maintainability Challenges:
  - Accumulation of shortcomings impacts efficiency and creates barriers
  - Harder for large-scale users to manage and reuse

Ideas to Enhance Ansible Roles

- `ansible-creator`:
  - “role2collection” import feature?
  - docs subcommand - inserts argument spec (argspec) in README template?
- `ansible-lint` rules:
  - Looks for recommended sections?
  - Require role argspec use?
  - Lint role invocation against its argument spec prior to execution?
- Ansible VS Code extension:
  - Collections tree view of plugins and roles docs?
  - Render and display docs for each collection, module, plugin, and role?
  - VS Code completion for roles having an argument spec?

Thoughts? Questions? What did we miss?

Anyone can contribute.
We are all community.

Join the forum to participate in discussions and get help!

Want to contribute? Find out how to get involved.

Thanks!

GitHub: **tima**
Matrix: **#community:ansible.com #social:ansible.com**

Ansible community forum:
https://forum.ansible.com/

ANSIBLE