Red Hat

Version 1.0
August 2023

Validation Workflow Guide

Red Hat Ansible Automation Platform

Red Hat

Version 1.0
August 2023

Contents

Red Hat

Version 1.0
August 2023

Introduction

Ansible validated content is a program dedicated to providing curated but non-supported content through Ansible Automation Platform (AAP).

Validated content is intended to help customers get started with use-case focused content and to show examples and best practices from Red Hat and technology partners.
Validated content is available on Ansible Automation Hub with the release of the Ansible Automation Platform 2.4.

This step-by-step guide will help describe the Ansible validated content workflow to partners who are interested in building this type of content.

Red Hat

Version 1.0
August 2023

STEP 1: Create Your Validated Content

Validated content is packaged inside of a collection, which is a standardized format for Ansible content. The Collections Developer Guide will help guide you through the entire process of developing collections, how the collection structure is organized and what each folder inside a collection means and should contain.

We do not require that partners publish their validated content on Galaxy, the open-source collection repository equivalent of Automation Hub. The intent of validated content is to allow partner ISVs to provide value to Ansible Automation Platform customers, and therefore the only business requirement is for it to be available through Automation Hub. This approach allows Red Hat and partners to collect targeted feedback that can be used to evolve, re-assess or even retire content at a faster pace.

Also, the Ansible Collections Checklist is for maintainers of Collections to provide them help, advice, and guidance on making sure their Collections are validated for correctness.

Namespace requirements

Validated Content should fall into one of the primary Ansible content domains to enable a better UX on Automation Hub.

Since Validated Content targets higher order use cases we expect more variety on the content itself compared to Certified Content.

The ABU maintains and makes available domain-specific namespaces to help differentiate Validated Content at a collection level.

cloud	For public cloud, private cloud and container native use cases
edge	For edge use cases
infra	For infrastructure use cases (e.g. OS and enterprise applications automation)
network	For network automation use cases
security	For security automation use cases

The suggested naming convention for Validated Content is *domain.platform*. The *platform* is the product the content is intended to be used on, and the *domain* must be one in the list above. Where complexity or scope requires it, another option is *domain.platform_usecase*. However, partners who write validated content and wish to keep that use-case specific content in their namespace are free to do so.

Validated Content is a suitable venue also for platform-agnostic content, in that case the suggested naming convention is *domain.use_case*.

STEP 2: Review Content Business Requirements

Validated Content must adhere to all business requirements shown below. Refer to this checklist for a summarization of the business and technical requirements.

Content Requirements

- Content in validated content collections must be YAML only:
    - Content must never duplicate content found in a certified collection.
    - A supported plugin or module used in validated roles must always be added as a dependency;
    - We recommend that your collections depend exclusively on certified or validated content collections, or having an equivalent support level provided by a Red Hat partner. Collections that fulfill this requirement will have a preferred/quicker route to validation;
    - Collections depending on community content will be evaluated case by case, and are required to have these dependencies clearly documented as “community”.
- **NOTE:** The only plugins that should be in a validated content collection are the ones that are highly use-case specific and are required for the YAML use-case content to function. If they're generic/reusable they should be in a separate collection. If Python content is included, the validated content must be tested with *ansible-test sanity*.

Repository and Maintainership Requirements

- The content is in a self certified repository, against the OpenSSF Best Practices.
- Validated content created by partners will remain in the partner’s repository. Validated content created by internal Red Hat teams and by trusted sources** will be migrated into the Ansible Community of Practice Github organization.

- Source of the collection is placed under an approved open source license and respects its prerequisites (e.g. code is freely available).
- README contains information about the purpose of the collection, how to install it, a link to the license file, and general usage and requirements information such as required versions of ansible-core and Python, and other required libraries or SDKs.
- Has 2 or more active maintainers listed in a CODEOWNERS file.

Other Policy Requirements

- There is no policy or legal reason for why the content shouldn't be published;
- The use case addressed by the content is not already covered by another validated content collection;
- The Collection follows semantic versioning requirements and is at or above version 1.0.0.

**Trusted-sources - Red Hat must be aware of who the maintainers are and where the content is originating.

STEP 3: Review Content Technical Requirements

The technical review requires content candidates to pass all *ansible-lint* "Production" profile checks, and to pass ansible-test sanity checks if Python content is included.

Installing and Running Ansible-lint

As a Github Action:
On GHA, the pre-built GHA for Ansible Lint can be used to run the linter as a Github Action.

Locally:
To install ansible lint, you can install from dmf, pip3, or from source.

```
$ pip3 install ansible-lint
```

```
$ dnf install ansible-lint
```

```
$ pip3 install git+https://github.com/ansible/ansible-lint
```

To run ansible-lint with all tests enforced, specify the Production profile.

```
$ ansible-lint --profile production
```

For info on all the available options, run `ansible-lint --help` and refer to the ansible-lint docs here: ansible-lint readthedocs.

Running Ansible-test sanity (if including Python content)

Sanity tests are made up of scripts and tools used to perform static code analysis. The primary purpose of these tests is to enforce Ansible coding standards and requirements. Please refer to the below commands for common ways to run `ansible-test`:

```
$ cd {...}/ansible_collections/{namespace}/{collection_name}/  # navigate to collection
```

```
$ ansible-test sanity   # run all sanity tests
```

You can see all sanity tests in the Sanity Testing section on the Developer Guide.

Any issue found in a piece of content candidate will be reported to the sponsor/author of the content to be addressed. Delays in addressing an issue may result in content rejection or removal. See the Validated Content Removal Policy.

The new collection must be placed under validation and testing of a CI/CD pipeline and guarantee a high test coverage of the collection. For a start, each role/module/plugin must be involved at least once in the testing cycle.

STEP 4: Submit Validated Content Candidate

After following the standards for creating validated content, you can submit your content candidate to ***this form***.

Once a candidate is submitted it gets added to a list and a notification will be sent to representatives of the Automation Community of Practice and Partner Engineering.

Candidates will be reviewed bi-weekly (or monthly, depending on the volume of submissions). If there are concerns with the submitted content, the reviewing team will reach out to the submitter with a list of items to address for resubmission.

Once candidates have been accepted, internal candidates will be asked to migrate their collections into the redhat-cop organization.

Note: Partner ISVs don't have to migrate their collections and may keep them in their current repository.

STEP 5: Content Approval and Distribution

If the content passes all requirements, it can be approved for distribution in private Automation Hub and on Ansible Automation Hub.

When validated content is uploaded to Automation Hub, Partner Engineering will do a final review of the content. If the content is approved, it will be signed and published on Automation Hub for consumption and the author will be notified via available channels.

If the content is rejected or there are concerns that need to be addressed before approval, Partner Engineering will reach out to the author with an explanation of the issue and details around why the content was not approved.

Validated Content Standalone Tarball Inclusion

In order for customers to be able to pre-load the validated content into their air-gapped instances of Private Automation Hub, the validated content needs to get included in the validated content tarball provided to our customers on access.redhat.com.

Contrary to the Certified Content standalone release, there is no specific due date for validated content; this is an artifact that lives in a rolling release fashion. Whenever a new version of a collection or a new collection is available, Partner Engineering will relay the information to our productization team. The inclusion will then be made and a new artifact be made available to our customers.

Validated Content Removal Policy

Red Hat reserves the right to remove a piece of validated content from Automation Hub at any time. Situations that could result in content removal include but are not limited to:

- If there are functionality issues with the content and the creator does not resolve the issue in a timely manner;
- Bug reports and bugfix PRs start piling up without being reviewed;
- If the content is outdated or otherwise no longer relevant to a platform or product;
- There has been no maintainer's activity in the collection repository for several months;
- CI has stopped passing (or even has not been running) for several months.