



Linux Ubuntu 26.04 Manual Pages on command 'ansible-galaxy.1'

\$ man ansible-galaxy.1

ansible-galaxy(1) System administration commands ansible-galaxy(1)

Name

ansible-galaxy - Perform various Role and Collection related operations.

SYNOPSIS

usage: ansible-galaxy [-h] [--version] [-v] TYPE ...

DESCRIPTION

Command to manage Ansible roles and collections.

None of the CLI tools are designed to run concurrently with themselves. Use an external scheduler and/or locking to ensure there are no clashing operations.

COMMON OPTIONS

--version

show program's version number, config file location, configured module search path, mod?

ule location, executable location and exit

-h, --help

show this help message and exit

-v, --verbose

Causes Ansible to print more debug messages. Adding multiple -v will increase the ver?

bosity, the builtin plugins currently evaluate up to -vvvvvv. A reasonable level to start

is -vvv, connection debugging might require -vvvv. This argument may be specified multi?

ple times.

ACTIONS

collection

Perform the action on an Ansible Galaxy collection. Must be combined with a further

action like init/install as listed below.

collection download

Download collections and their dependencies as a tarball for an offline install.

--clear-response-cache

Clear the existing server response cache.

--no-cache

Do not use the server response cache.

--pre

Include pre-release versions. Semantic versioning pre-releases are ignored by default.

fault

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-n, --no-deps

Don't download collection(s) listed as dependencies.

-p 'DOWNLOAD_PATH', --download-path 'DOWNLOAD_PATH'

The directory to download the collections to.

-r 'REQUIREMENTS', --requirements-file 'REQUIREMENTS'

A file containing a list of collections to be downloaded.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection init

Creates the skeleton framework of a role or collection that complies with the Galaxy metadata format. Requires a role or collection name. The collection name must be in the format <namespace>.<collection>.

--collection-skeleton 'COLLECTION_SKELETON'

The path to a collection skeleton that the new collection should be based upon.

--init-path 'INIT_PATH'

The path in which the skeleton collection will be created. The default is the current directory.

rent working directory.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-e, --extra-vars

set additional variables as key=value or YAML/JSON, if filename prepend with @.

This argument may be specified multiple times.

-f, --force

Force overwriting an existing role or collection

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection build

Build an Ansible Galaxy collection artifact that can be stored in a central repository like Ansible Galaxy. By default, this command builds from the current working directory. You can optionally pass in the collection input path (where the galaxy.yml file is).

--output-path 'OUTPUT_PATH'

The path in which the collection is built to. The default is the current working directory.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-f, --force

Force overwriting an existing role or collection

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection publish

Publish a collection into Ansible Galaxy. Requires the path to the collection tarball to publish.

--import-timeout 'IMPORT_TIMEOUT'

The time to wait for the collection import process to finish.

--no-wait

Don't wait for import validation results.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection install

Install one or more roles(ansible-galaxy role install), or one or more collections(ansible-galaxy collection install). You can pass in a list (roles or collections) or use the file option listed below (these are mutually exclusive). If you pass in a list, it can be a name (which will be downloaded via the galaxy API and github), or it can be a local tar archive file.

--clear-response-cache

Clear the existing server response cache.

--disable-gpg-verify

Disable GPG signature verification when installing collections from a Galaxy server

--force-with-deps

Force overwriting an existing collection and its dependencies.

--ignore-signature-status-code

==SUPPRESS==. This argument may be specified multiple times.

--ignore-signature-status-codes

A space separated list of status codes to ignore during signature verification (for example, NO_PUBKEY FAILURE). Descriptions for the choices can be seen at [L\(? <https://github.com/gpg/gnupg/blob/master/doc/DETAILS#general-status-codes>\)](https://github.com/gpg/gnupg/blob/master/doc/DETAILS#general-status-codes).Note>: specify these after positional arguments or use -- to separate them. This argument may be specified multiple times.

--keyring 'KEYRING'

The keyring used during signature verification

--no-cache

Do not use the server response cache.

--offline

Install collection artifacts (tarballs) without contacting any distribution servers. This does not apply to collections in remote Git repositories or URLs to remote tarballs.

--pre

Include pre-release versions. Semantic versioning pre-releases are ignored by default

--required-valid-signature-count 'REQUIRED_VALID_SIGNATURE_COUNT'

The number of signatures that must successfully verify the collection. This should be a positive integer or -1 to signify that all signatures must be used to verify the collection. Prepend the value with + to fail if no valid signatures are found for the collection (e.g. +all).

--signature

An additional signature source to verify the authenticity of the MANIFEST.json before installing the collection from a Galaxy server. Use in conjunction with a positional collection name (mutually exclusive with --requirements-file). This argument may be specified multiple times.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

-U, --upgrade

Upgrade installed collection artifacts. This will also update dependencies unless

--no-deps is provided

-c, --ignore-certs

Ignore SSL certificate validation errors.

-f, --force

Force overwriting an existing role or collection

-i, --ignore-errors

Ignore errors during installation and continue with the next specified collection.

This will not ignore dependency conflict errors.

-n, --no-deps

Don't download collections listed as dependencies.

-p 'COLLECTIONS_PATH', --collections-path 'COLLECTIONS_PATH'

The path to the directory containing your collections.

-r 'REQUIREMENTS', --requirements-file 'REQUIREMENTS'

A file containing a list of collections to be installed.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection list

List installed collections or roles

--format 'OUTPUT_FORMAT'

Format to display the list of collections in.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-p, --collections-path

One or more directories to search for collections in addition to the default

COLLECTIONS_PATHS. Separate multiple paths with ':'. This argument may be specified multiple times.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

collection verify

Compare checksums with the collection(s) found on the server and the installed copy.

This does not verify dependencies.

`--ignore-signature-status-code`

`==SUPPRESS==`. This argument may be specified multiple times.

`--ignore-signature-status-codes`

A space separated list of status codes to ignore during signature verification

(for example, `NO_PUBKEY FAILURE`). Descriptions for the choices can be seen at [L\(? <https://github.com/gpg/gnupg/blob/master/doc/DETAILS#general-status-codes>](https://github.com/gpg/gnupg/blob/master/doc/DETAILS#general-status-codes).Note>:

specify these after positional arguments or use `--` to separate them. This argument

may be specified multiple times.

`--keyring 'KEYRING'`

The keyring used during signature verification

`--offline`

Validate collection integrity locally without contacting server for canonical manifest hash.

ifest hash.

`--required-valid-signature-count 'REQUIRED_VALID_SIGNATURE_COUNT'`

The number of signatures that must successfully verify the collection. This should

be a positive integer or `all` to signify that all signatures must be used to verify

the collection. Prepend the value with `+` to fail if no valid signatures are found

for the collection (e.g. `+all`).

`--signature`

An additional signature source to verify the authenticity of the `MANIFEST.json` before

using it to verify the rest of the contents of a collection from a Galaxy

server. Use in conjunction with a positional collection name (mutually exclusive

with `--requirements-file`). This argument may be specified multiple times.

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

`-i, --ignore-errors`

Ignore errors during verification and continue with the next specified collection.

`-p, --collections-path`

One or more directories to search for collections in addition to the default COL?

LECTIONS_PATHS. Separate multiple paths with ':'. This argument may be specified multiple times.

`-r 'REQUIREMENTS', --requirements-file 'REQUIREMENTS'`

A file containing a list of collections to be verified.

`-s 'API_SERVER', --server 'API_SERVER'`

The Galaxy API server URL

role Perform the action on an Ansible Galaxy role. Must be combined with a further action like delete/install/init as listed below.

role init

Creates the skeleton framework of a role or collection that complies with the Galaxy metadata format. Requires a role or collection name. The collection name must be in the format <namespace>.<collection>.

`--init-path 'INIT_PATH'`

The path in which the skeleton role will be created. The default is the current working directory.

`--offline`

Don't query the galaxy API when creating roles

`--role-skeleton 'ROLE_SKELETON'`

The path to a role skeleton that the new role should be based upon.

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

`--type 'ROLE_TYPE'`

Initialize using an alternate role type. Valid types include: 'container', 'apb' and 'network'.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

-e, --extra-vars

set additional variables as key=value or YAML/JSON, if filename prepend with @.

This argument may be specified multiple times.

-f, --force

Force overwriting an existing role or collection

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

role remove

removes the list of roles passed as arguments from the local system.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-p, --roles-path

The path to the directory containing your roles. The default is the first writable one configured via DEFAULT_ROLES_PATH: {{ ANSIBLE_HOME ~ "/roles:/usr/share/ansible/roles:/etc/ansible/roles" }} . This argument may be specified multiple times.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

role delete

Delete a role from Ansible Galaxy.

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

role list

List installed collections or roles

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-p, --roles-path

The path to the directory containing your roles. The default is the first writable one configured via DEFAULT_ROLES_PATH: {{ ANSIBLE_HOME ~ "/roles:/usr/share/ansible/roles:/etc/ansible/roles" }} . This argument may be specified multiple times.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

role search

searches for roles on the Ansible Galaxy server

--author 'AUTHOR'

GitHub username

--galaxy-tags 'GALAXY_TAGS'

list of galaxy tags to filter by

--platforms 'PLATFORMS'

list of OS platforms to filter by

--timeout 'TIMEOUT'

The time to wait for operations against the galaxy server, defaults to 60s.

--token 'API_KEY', --api-key 'API_KEY'

The Ansible Galaxy API key which can be found at <<https://galaxy.ansible.com/me/preferences>>.

-c, --ignore-certs

Ignore SSL certificate validation errors.

-s 'API_SERVER', --server 'API_SERVER'

The Galaxy API server URL

role import

used to import a role into Ansible Galaxy

`--branch 'REFERENCE'`

The name of a branch to import. Defaults to the repository's default branch (usually master)

`--no-wait`

Don't wait for import results.

`--role-name 'ROLE_NAME'`

The name the role should have, if different than the repo name

`--status`

Check the status of the most recent import request for given github_user/github_repo.

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

`-s 'API_SERVER', --server 'API_SERVER'`

The Galaxy API server URL

role setup

Setup an integration from Github or Travis for Ansible Galaxy roles

`--list`

List all of your integrations.

`--remove 'REMOVE_ID'`

Remove the integration matching the provided ID value. Use `--list` to see ID values.

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

`-p, --roles-path`

The path to the directory containing your roles. The default is the first writable one configured via `DEFAULT_ROLES_PATH: {{ ANSIBLE_HOME ~ "/roles:/usr/share/ansible/roles:/etc/ansible/roles" }}`. This argument may be specified multiple times.

`-s 'API_SERVER', --server 'API_SERVER'`

The Galaxy API server URL

role info

prints out detailed information about an installed role as well as info available from the galaxy API.

`--offline`

Don't query the galaxy API when creating roles

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

`-p, --roles-path`

The path to the directory containing your roles. The default is the first writable one configured via `DEFAULT_ROLES_PATH: {{ ANSIBLE_HOME ~ "/roles:/usr/share/ansible/roles:/etc/ansible/roles" }}`. This argument may be specified multiple times.

`-s 'API_SERVER', --server 'API_SERVER'`

The Galaxy API server URL

role install

Install one or more roles(`ansible-galaxy role install`), or one or more collections(`ansible-galaxy collection install`). You can pass in a list (roles or collections) or use the file option listed below (these are mutually exclusive). If you pass in a list, it can be a name (which will be downloaded via the galaxy API and github), or it can be a local tar archive file.

`--force-with-deps`

Force overwriting an existing role and its dependencies.

`--timeout 'TIMEOUT'`

The time to wait for operations against the galaxy server, defaults to 60s.

`--token 'API_KEY', --api-key 'API_KEY'`

The Ansible Galaxy API key which can be found at <https://galaxy.ansible.com/me/preferences>.

`-c, --ignore-certs`

Ignore SSL certificate validation errors.

`-f, --force`

Force overwriting an existing role or collection

`-g, --keep-scm-meta`

Use tar instead of the scm archive option when packaging the role.

`-i, --ignore-errors`

Ignore errors and continue with the next specified role.

`-n, --no-deps`

Don't download roles listed as dependencies.

`-p, --roles-path`

The path to the directory containing your roles. The default is the first writable one configured via `DEFAULT_ROLES_PATH`: `{{ ANSIBLE_HOME ~ "/roles:/usr/share/ansible/roles:/etc/ansible/roles" }}`. This argument may be specified multiple times.

`-r 'REQUIREMENTS', --role-file 'REQUIREMENTS'`

A file containing a list of roles to be installed.

`-s 'API_SERVER', --server 'API_SERVER'`

The Galaxy API server URL

ENVIRONMENT

The following environment variables may be specified.

`ANSIBLE_CONFIG` -- Specify override location for the ansible config file

Many more are available for most options in `ansible.cfg`

For a full list check <https://docs.ansible.com/>. or use the `ansible-config` command.

FILES

`/etc/ansible/ansible.cfg` -- Config file, used if present

`~/.ansible.cfg` -- User config file, overrides the default config if present

`./ansible.cfg` -- Local config file (in current working directory) assumed to be 'project specific' and overrides the rest if present.

As mentioned above, the `ANSIBLE_CONFIG` environment variable will override all others.

AUTHOR

Ansible was originally written by Michael DeHaan.

COPYRIGHT

Copyright ? 2018 Red Hat, Inc | Ansible. Ansible is released under the terms of the GPLv3 license.

SEE ALSO

ansible (1), ansible-config (1), ansible-console (1), ansible-doc (1), ansible-inventory (1), ansible-playbook (1), ansible-pull (1), ansible-vault (1)

Extensive documentation is available in the documentation site: <<<https://docs.ansible.com>>>. IRC and mailing list info can be found in file CONTRIBUTING.md, available in: <?<https://github.com/ansible/ansible>>>

Ansible 2.20.1

ansible-galaxy(1)