



(/rol/app/)

Search

Red Hat Enterprise Linux Automation with Ansible

 **FEEDBACK**

TRANSLATIONS ▾

CERTIFICATE OF ATTENDANCE





P (/rol/app/courses/rh294-8.4/pages/pr01)	(/rol/app/courses/rh294-8.4/pages/pr01s02)	1 (/rol/app/courses/rh294-8.4/pages/ch01)
(/rol/app/courses/rh294-8.4/pages/ch01s02)	(/rol/app/courses/rh294-8.4/pages/ch01s03)	(/rol/app/courses/rh294-8.4/pages/ch01s04)
8.4/pages/pr01)	(/rol/app/courses/rh294-8.4/pages/ch02)	3.4/pages/ch01)
(/rol/app/courses/rh294-8.4/pages/ch01s05)	(/rol/app/courses/rh294-8.4/pages/ch02s04)	(/rol/app/courses/rh294-8.4/pages/ch02s02)
(/rol/app/courses/rh294-8.4/pages/ch02s03)	3.4/pages/ch02)	(/rol/app/courses/rh294-8.4/pages/ch02s05)
(/rol/app/courses/rh294-8.4/pages/ch02s06)	(/rol/app/courses/rh294-8.4/pages/ch02s07)	(/rol/app/courses/rh294-8.4/pages/ch02s08)
(/rol/app/courses/rh294-8.4/pages/ch02s09)	(/rol/app/courses/rh294-8.4/pages/ch02s10)	(/rol/app/courses/rh294-8.4/pages/ch02s11)
(/rol/app/courses/rh294-8.4/pages/ch02s12)	3 (/rol/app/courses/rh294-8.4/pages/ch03)	(/rol/app/courses/rh294-8.4/pages/ch03s02)
(/rol/app/courses/rh294-8.4/pages/ch03s03)	(/rol/app/courses/rh294-8.4/pages/ch03s04)	(/rol/app/courses/rh294-8.4/pages/ch03s05)
(/rol/app/courses/rh294-8.4/pages/ch03s06)	8.4/pages/ch03)	(/rol/app/courses/rh294-8.4/pages/ch03s08)
(/rol/app/courses/rh294-8.4/pages/ch04)	(/rol/app/courses/rh294-8.4/pages/ch04s02)	4 (/rol/app/courses/rh294-8.4/pages/ch04s03)
(/rol/app/courses/rh294-8.4/pages/ch04s04)	(/rol/app/courses/rh294-8.4/pages/ch04s05)	(/rol/app/courses/rh294-8.4/pages/ch04s06)
(/rol/app/courses/rh294-8.4/pages/ch04s07)	(/rol/app/courses/rh294-8.4/pages/ch04s08)	5 (/rol/app/courses/rh294-8.4/pages/ch05)
(/rol/app/courses/rh294-8.4/pages/ch05s02)	(/rol/app/courses/rh294-8.4/pages/ch05s03)	(/rol/app/courses/rh294-8.4/pages/ch05s04)
(/rol/app/courses/rh294-8.4/pages/ch05s05)	(/rol/app/courses/rh294-8.4/pages/ch05s06)	8.4/pages/ch05)
(/rol/app/courses/rh294-8.4/pages/ch06s02)	(/rol/app/courses/rh294-8.4/pages/ch06s03)	(/rol/app/courses/rh294-8.4/pages/ch06)
(/rol/app/courses/rh294-8.4/pages/ch06s05)	(/rol/app/courses/rh294-8.4/pages/ch06s06)	(/rol/app/courses/rh294-8.4/pages/ch06s04)
(/rol/app/courses/rh294-8.4/pages/ch07s02)	(/rol/app/courses/rh294-8.4/pages/ch07s03)	8.4/pages/ch06)
(/rol/app/courses/rh294-8.4/pages/ch07s05)	(/rol/app/courses/rh294-8.4/pages/ch07s06)	(/rol/app/courses/rh294-8.4/pages/ch07)
(/rol/app/courses/rh294-8.4/pages/ch07s08)	(/rol/app/courses/rh294-8.4/pages/ch07s09)	(/rol/app/courses/rh294-8.4/pages/ch07s04)
(/rol/app/courses/rh294-8.4/pages/ch07s11)	(/rol/app/courses/rh294-8.4/pages/ch07s12)	(/rol/app/courses/rh294-8.4/pages/ch07s07)
(/rol/app/courses/rh294-8.4/pages/ch08s02)	(/rol/app/courses/rh294-8.4/pages/ch08s03)	(/rol/app/courses/rh294-8.4/pages/ch07s10)
(/rol/app/courses/rh294-8.4/pages/ch08s05)	(/rol/app/courses/rh294-8.4/pages/ch08s06)	8 (/rol/app/courses/rh294-8.4/pages/ch08)
(/rol/app/courses/rh294-8.4/pages/ch09s02)	(/rol/app/courses/rh294-8.4/pages/ch09s03)	(/rol/app/courses/rh294-8.4/pages/ch08s04)
(/rol/app/courses/rh294-8.4/pages/ch09s05)	(/rol/app/courses/rh294-8.4/pages/ch09s06)	8.4/pages/ch08)
(/rol/app/courses/rh294-8.4/pages/ch09s08)	(/rol/app/courses/rh294-8.4/pages/ch09s09)	(/rol/app/courses/rh294-8.4/pages/ch09)
(/rol/app/courses/rh294-8.4/pages/ch09s11)	(/rol/app/courses/rh294-8.4/pages/ch09s12)	(/rol/app/courses/rh294-8.4/pages/ch09s04)
(/rol/app/courses/rh294-8.4/pages/ch10s02)	(/rol/app/courses/rh294-8.4/pages/ch10s03)	8.4/pages/ch09)
A (/rol/app/courses/rh294-8.4/pages/apa)	(/rol/app/courses/rh294-8.4/pages/apa)	(/rol/app/courses/rh294-8.4/pages/ch09s07)
	(/rol/app/courses/rh294-8.4/pages/apb)	(/rol/app/courses/rh294-8.4/pages/ch09s10)
		10 (/rol/app/courses/rh294-8.4/pages/ch10)
		(/rol/app/courses/rh294-8.4/pages/ch10s04)
		8.4/pages/ch10)
		(/rol/app/courses/rh294-8.4/pages/apb)

Deploying Roles with Ansible Galaxy



Objectives

After completing this section, you should be able to select and retrieve roles from Ansible Galaxy or other sources such as a Git repository, and use them in your playbooks.

Introducing Ansible Galaxy

Ansible Galaxy (<https://galaxy.ansible.com>) is a public library of Ansible content written by a variety of Ansible administrators and users. It contains thousands of Ansible roles and it has a searchable database that helps Ansible users identify roles that might help them accomplish an administrative task. Ansible Galaxy includes links to documentation and videos for new Ansible users and role developers.

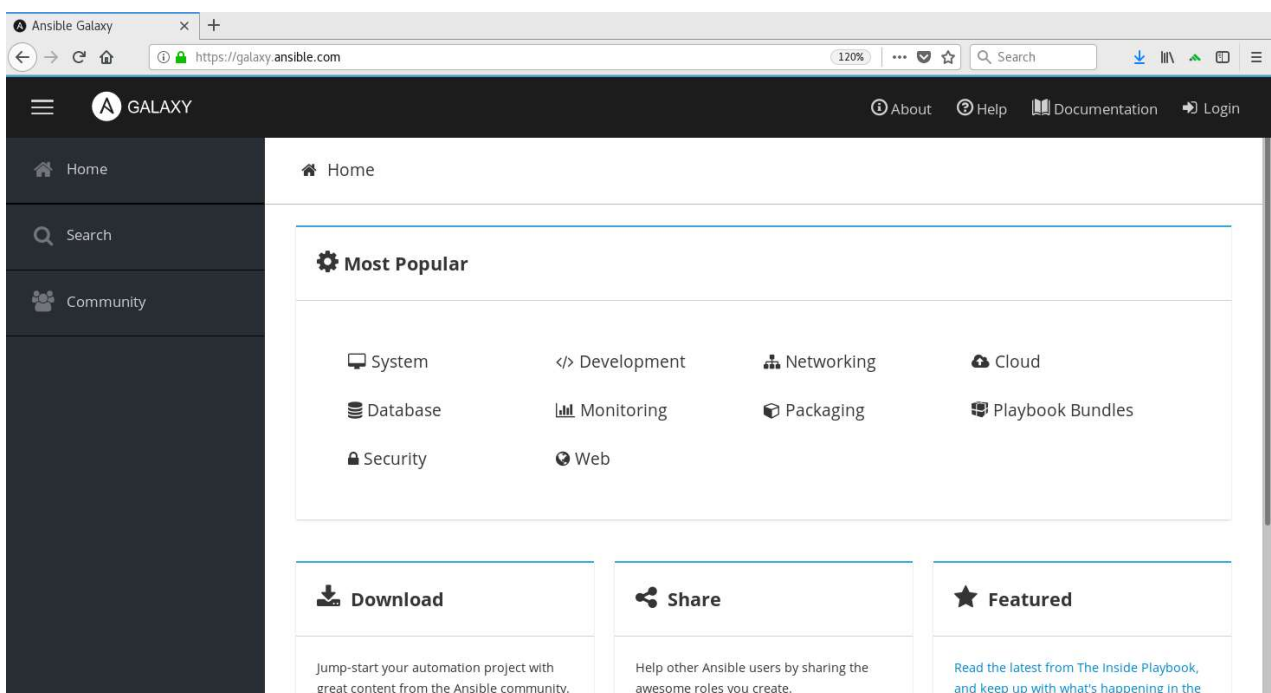


Figure 7.1: Ansible Galaxy home page

In addition, the `ansible-galaxy` command that you use to get and manage roles from Ansible Galaxy can also be used to get and manage roles your projects need from your own Git repositories.

Getting Help with Ansible Galaxy

The **Documentation** tab on the Ansible Galaxy website home page leads to a page that describes how to use Ansible Galaxy. There is content that describes how to download and use roles from Ansible Galaxy. Instructions on how to develop roles and upload them to Ansible Galaxy are also on that page.

Browsing Ansible Galaxy for Roles

The **Search** tab on the left side of the Ansible Galaxy website home page gives users access to information about the roles published on Ansible Galaxy. You can search for an Ansible role by its name, using tags, or by other role attributes. Results are presented in descending order of the **Best Match** score, which is a computed score based on role quality, role popularity, and search criteria.

NOTE

Content Scoring (https://galaxy.ansible.com/docs/contributing/content_scoring.html) in the documentation has more information on how roles are scored by Ansible Galaxy.

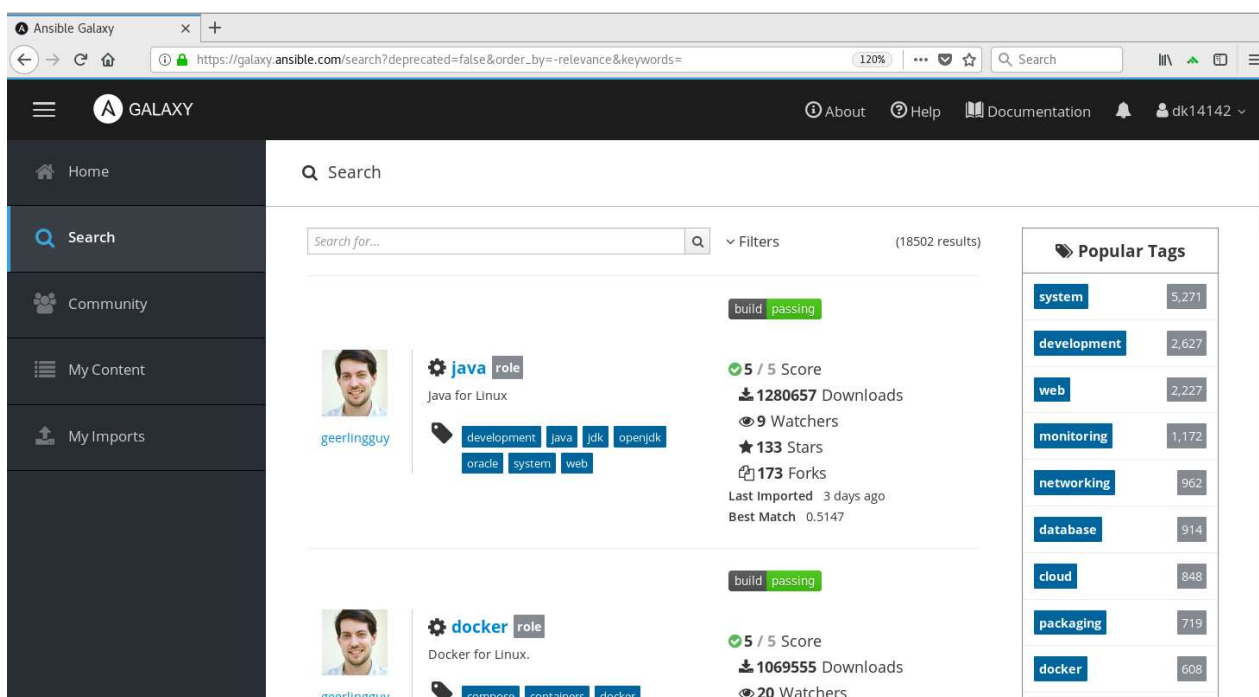


Figure 7.2: Ansible Galaxy search screen

Ansible Galaxy reports the number of times each role has been downloaded from Ansible Galaxy. In addition, Ansible Galaxy also reports the number of watchers, forks, and stars the role's GitHub repository has. Users can use this information to help determine how active development is for a role and how popular it is in the community.

The following figure shows the search results that Ansible Galaxy displayed after a keyword search for `redis` was performed. Notice the first result has a **Best Match** score of 0.9009.

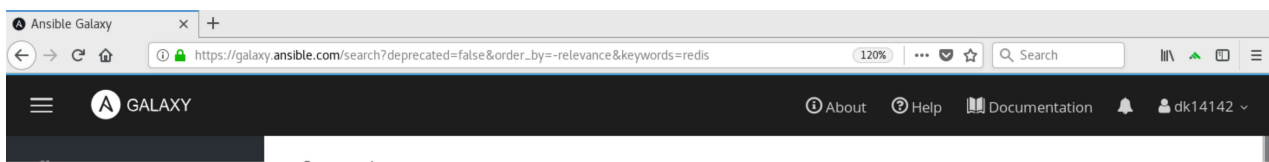


Figure 7.3: Ansible Galaxy search results example

The **Filters** pulldown menu to the right of the search box allow searches to be performed on keywords, author IDs, platform, and tags. Possible platform values include EL for Red Hat Enterprise Linux (and closely related distributions such as CentOS) and Fedora, among others.

Tags are arbitrary single-word strings set by the role author that describe and categorize the role. Users can use tags to find relevant roles. Possible tag values include `system`, `development`, `web`, `monitoring`, and others. A role can have up to 20 tags in Ansible Galaxy.

IMPORTANT

In the Ansible Galaxy search interface, keyword searches match words or phrases in the README file, content name, or content description. Tag searches, by contrast, specifically match tag values set by the author for the role.

The Ansible Galaxy Command-Line Tool

The `ansible-galaxy` command line tool can be used to search for, display information about, install, list, remove, or initialize roles.

Searching for Roles from the Command Line

The `ansible-galaxy search` subcommand searches Ansible Galaxy for roles. If you specify a string as an argument, it is used to search Ansible Galaxy for roles by keyword. You can use the `--author`, `--platforms`, and `--galaxy-tags` options to narrow the search results. You can also use those options as the main search key. For example, the command `ansible-galaxy search --author geerlingguy` will display all roles submitted by the user `geerlingguy`.

Results are displayed in alphabetical order, not by descending `Best Match` score. The following example displays the names of roles that include `redis`, and are available for the Enterprise Linux (EL) platform.

```
[user@host ~]$ ansible-galaxy search 'redis' --platforms EL

Found 124 roles matching your search:
```

Name	Description
----	-----
1it.sudo	Ansible role for managing sudoers
AerisCloud.librato	Install and configure the Librato Agent
AerisCloud.redis	Installs redis on a server
AlbanAndrieu.java	Manage Java installation
andrewrothstein.redis	builds Redis from src and installs
...output omitted...	
geerlingguy.php-redis	PhpRedis support for Linux
geerlingguy.redis	Redis for Linux
gikoluo.filebeat	Filebeat for Linux.
...output omitted...	

The `ansible-galaxy info` subcommand displays more detailed information about a role. Ansible Galaxy gets this information from a number of places including the role's `meta/main.yml` file and its GitHub repository. The following command displays information about the `geerlingguy.redis` role, available from Ansible Galaxy.

```
[user@host ~]$ ansible-galaxy info geerlingguy.redis

Role: geerlingguy.redis
  description: Redis for Linux
  active: True
...output omitted...
  download_count: 146209
  forks_count: 82
  github_branch: master
  github_repo: ansible-role-redis
  github_user: geerlingguy
...output omitted...
  license: license (BSD, MIT)
  min_ansible_version: 2.4
  modified: 2018-11-19T14:53:29.722718Z
  open_issues_count: 11
  path: [u'/etc/ansible/roles', u'/usr/share/ansible/roles']
  role_type: ANS
  stargazers_count: 98
...output omitted...
```

Installing Roles from Ansible Galaxy

The `ansible-galaxy install` subcommand downloads a role from Ansible Galaxy, then installs it locally on the control node.

By default, roles are installed into the first directory that is writable in the user's `roles_path`. Based on the default `roles_path` set for Ansible, normally the role will be installed into the user's `~/.ansible/roles` directory. The default `roles_path` might be overridden by your current Ansible configuration file or by the environment variable `ANSIBLE_ROLES_PATH`, which affects the behavior of `ansible-galaxy`.

You can also specify a specific directory to install the role into by using the `-p DIRECTORY` option.

In the following example, `ansible-galaxy` installs the `geerlingguy.redis` role into a playbook project's `roles` directory. The command's current working directory is `/opt/project`.

```
[user@host project]$ ansible-galaxy install geerlingguy.redis -p roles/
- downloading role 'redis', owned by geerlingguy
- downloading role from https://github.com/geerlingguy/...output omitted...
- extracting geerlingguy.redis to /opt/project/roles/geerlingguy.redis
- geerlingguy.redis (1.6.0) was installed successfully
[user@host project]$ ls roles/
geerlingguy.redis
```

Installing Roles Using a Requirements File

You can also use `ansible-galaxy` to install a list of roles based on definitions in a text file. For example, if you have a playbook that needs to have specific roles installed, you can create a `roles/requirements.yml` file in the project directory that specifies which roles are needed. This file acts as a dependency manifest for the playbook project which enables playbooks to be developed and tested separately from any supporting roles.

For example, a simple `requirements.yml` to install `geerlingguy.redis` might read like this:

```
- src: geerlingguy.redis
  version: "1.5.0"
```

The `src` attribute specifies the source of the role, in this case the `geerlingguy.redis` role from Ansible Galaxy. The `version` attribute is optional, and specifies the version of the role to install, in this case `1.5.0`.

IMPORTANT

You should specify the version of the role in your `requirements.yml` file, especially for playbooks in production.

If you do not specify a version, you will get the latest version of the role. If the upstream author makes changes to the role that are incompatible with your playbook, it may cause an automation failure or other problems.

To install the roles using a role file, use the `-r REQUIREMENTS-FILE` option:

```
[user@host project]$ ansible-galaxy install -r roles/requirements.yml \
> -p roles
- downloading role 'redis', owned by geerlingguy
- downloading role from https://github.com/geerlingguy/ansible-role-redis/archive/1.6.0.tar.gz
- extracting geerlingguy.redis to /opt/project/roles/geerlingguy.redis
- geerlingguy.redis (1.6.0) was installed successfully
```

You can use `ansible-galaxy` to install roles that are not in Ansible Galaxy. You can host your own proprietary or internal roles in a private Git repository or on a web server. The following example shows how to configure a `requirements` file using a variety of remote sources.

```
[user@host project]$ cat roles/requirements.yml
# from Ansible Galaxy, using the latest version
- src: geerlingguy.redis

# from Ansible Galaxy, overriding the name and using a specific version
- src: geerlingguy.redis
  version: "1.5.0"
  name: redis_prod

# from any Git-based repository, using HTTPS
- src: https://gitlab.com/guardianproject-ops/ansible-nginx-acme.git
  scm: git
  version: 56e00a54
  name: nginx-acme

# from any Git-based repository, using SSH
- src: git@gitlab.com:guardianproject-ops/ansible-nginx-acme.git
  scm: git
  version: master
  name: nginx-acme-ssh

# from a role tar ball, given a URL;
# supports 'http', 'https', or 'file' protocols
- src: file:///opt/local/roles/myrole.tar
  name: myrole
```

The `src` keyword specifies the Ansible Galaxy role name. If the role is not hosted on Ansible Galaxy, the `src` keyword indicates the role's URL.

If the role is hosted in a source control repository, the `scm` attribute is required. The `ansible-galaxy` command is capable of downloading and installing roles from either a Git-based or mercurial-based software repository. A Git-based repository requires an `scm` value of `git`, while a role hosted on a mercurial repository requires a value of `hg`. If the role is hosted on Ansible Galaxy or as a tar archive on a web server, the `scm` keyword is omitted.

The `name` keyword is used to override the local name of the role. The `version` keyword is used to specify a role's version. The `version` keyword can be any value that corresponds to a branch, tag, or commit hash from the role's software repository.

To install the roles associated with a playbook project, execute the `ansible-galaxy install` command:

```
[user@host project]$ ansible-galaxy install -r roles/requirements.yml \
> -p roles
- downloading role 'redis', owned by geerlingguy
- downloading role from https://github.com/geerlingguy/ansible-role-redis/archive/1.6.0.tar.gz
- extracting geerlingguy.redis to /opt/project/roles/geerlingguy.redis
- geerlingguy.redis (1.6.0) was installed successfully
- downloading role 'redis', owned by geerlingguy
- downloading role from https://github.com/geerlingguy/ansible-role-redis/archive/1.5.0.tar.gz
- extracting redis_prod to /opt/project/roles/redis_prod
- redis_prod (1.5.0) was installed successfully
- extracting nginx-acme to /opt/project/roles/nginx-acme
- nginx-acme (56e00a54) was installed successfully
- extracting nginx-acme-ssh to /opt/project/roles/nginx-acme-ssh
- nginx-acme-ssh (master) was installed successfully
- downloading role from file:///opt/local/roles/myrole.tar
- extracting myrole to /opt/project/roles/myrole
- myrole was installed successfully
```

Managing Downloaded Roles

The `ansible-galaxy` command can also manage local roles, such as those roles found in the `roles` directory of a playbook project. The `ansible-galaxy list` subcommand lists the roles that are found locally.

```
[user@host project]$ ansible-galaxy list
- geerlingguy.redis, 1.6.0
- myrole, (unknown version)
- nginx-acme, 56e00a54
- nginx-acme-ssh, master
- redis_prod, 1.5.0
```

A role can be removed locally with the `ansible-galaxy remove` subcommand.

```
[user@host ~]$ ansible-galaxy remove nginx-acme-ssh
- successfully removed nginx-acme-ssh
[user@host ~]$ ansible-galaxy list
- geerlingguy.redis, 1.6.0
- myrole, (unknown version)
- nginx-acme, 56e00a54
- redis_prod, 1.5.0
```

Use downloaded and installed roles in playbooks like any other role. They may be referenced in the `roles` section using their downloaded role name. If a role is not in the project's `roles` directory, the `roles_path` will be checked to see if the role is installed in one of those directories, first match being used. The following `use-role.yml` playbook references the `redis_prod` and `geerlingguy.redis` roles:

```
[user@host project]$ cat use-role.yml
---
- name: use redis_prod for Prod machines
  hosts: redis_prod_servers
  remote_user: devops
  become: true
  roles:
    - redis_prod

- name: use geerlingguy.redis for Dev machines
  hosts: redis_dev_servers
  remote_user: devops
  become: true
  roles:
    - geerlingguy.redis
```

This playbook causes different versions of the `geerlingguy.redis` role to be applied to the production and development servers. In this manner, changes to the role can be systematically tested and integrated before deployment to the production servers. If a recent change to a role causes problems, using version control to develop the role allows you to roll back to a previous, stable version of the role.

REFERENCES

Ansible Galaxy – Ansible Documentation (<https://docs.ansible.com/ansible/2.9/cli/ansible-galaxy.html>)

← PREVIOUS (/ROL/APP/COURSES/RH294-8.4/PAGES/CH07S06) → NEXT (/ROL/APP/COURSES/RH294-8.4/PAGES/CH07S08)

Privacy Policy (http://s.bl-l.com/h/cZrgWbQn?url=https://www.redhat.com/en/about/privacy-policy?extldCarryOver=true&sc_cid=701f2000001D8QoAAK)

Red Hat Training Policies (<http://s.bl-l.com/h/cZrb2DXG?url=https://www.redhat.com/en/about/red-hat-training-policies>)

Terms of Use (<https://www.redhat.com/en/about/terms-use>)

All policies and guidelines (<https://www.redhat.com/en/about/all-policies-guidelines>)

Release Notes (<https://learn.redhat.com/t5/Red-Hat-Learning-Subscription/Red-Hat-Learning-Subscription-Release-Notes/ba-p/22952>)

[Cookie Preferences](#)

