OpenStack-Ansible Documentation: repo_server role

Release 18.1.0.dev196

OpenStack-Ansible Contributors

Mar 04, 2026

CONTENTS

CHAPTER

ONE

ABSTRACT

Ansible role that deploys a repository server for built python packages (wheels), requirements and constraints for specific builds.

To clone or view the source code for this repository, visit the role repository for repo_server.

CHAPTER TWO

ROLE PURPOSE

Repo container is used as a target by other OpenStack-Ansible roles and collections when *venv_wheel_build_enable : true*.

In this scenario it is expected, that there will be a repo instance per each Operating System family and major version, as well as for each CPU architecture in the deployments.

CHAPTER

THREE

WEB SERVER

In order to serve pre-built content to clients (like `pip` or `uv`), an Apache Web server is being used.

We leverage httpd role to set up a web server, and to manage a corresponding Virtual Host.

CHAPTER FOUR

USING SHARED FILE SYSTEM

When multiple instances of a repo server exist, is designed to leverage a shared filesystem, mounted for `/var/www/repo` directory.

This filesystem is used to store built results and ensure that each repo server is able to serve wheels for all available variants in the deployment.

There are no requirements to filesystem performance or reliability, as stored cache data can be re-built from scratch, in case of filesystem failure.

By default, `openstack.osa.repo` playbook will install a GlusterFS as a shared filesystem directly on all repo servers. You can disable this behavior by setting `openstack_repo_server_enable_glusterfs: false`.

You can also use any existing shared filesystem by defining `repo_server_systemd_mounts` variable - in this case it will be mounted via systemd_mount role.

CHAPTER FIVE

DEFAULT VARIABLES

```
## Verbosity Options
debug: false

## APT Cache Options
cache_timeout: 600

# Set the package install state for distribution and pip packages
# Options are 'present' and 'latest'
repo_server_package_state: "latest"

repo_server_name: openstack-slushee

repo_service_home_folder: "{{ _repo_service_home_folder }}"
repo_service_user_name: "{{ _repo_service_user_name }}"
repo_service_group_name: "{{ _repo_service_group_name }}"

# Main web server port
repo_server_bind_address: "{{ openstack_service_bind_address | default('0.0.0.
↪0') }}"
repo_server_port: 8181
repo_server_directory_root: /var/www/repo
repo_apache_log_level: info

## Cap the maximum number of threads / workers when a user value is␣
↪unspecified.

# This directory is used on the deploy host to create u-c files which are then
# copied to the repo server and served by http. Any other files in this
# directory placed by the deployer will also be transferred
repo_upper_constraints_path: "/etc/openstack_deploy/upper-constraints"

# Multiple repo servers must have a shared /var/www/repo
repo_server_systemd_mounts: []

# Example using remote shared filesystem to synchronise the repo contents␣
↪between
# several repo servers
# repo_server_systemd_mounts:
```

(continues on next page)

(continued from previous page)

```
#   - what: "gluster-server:gluster-volume-name"
#     where: "/var/www/repo"
#     type: glusterfs
#     state: 'started'
#     enabled: true

###
### Backend TLS
###

# Define if communication between haproxy and service backends should be
# encrypted with TLS.
repo_backend_ssl: "{{ openstack_service_backend_ssl | default(False) }}"

# Storage location for SSL certificate authority
repo_pki_dir: "{{ openstack_pki_dir | default('/etc/openstack_deploy/pki') }}"

# Delegated host for operating the certificate authority
repo_pki_setup_host: "{{ openstack_pki_setup_host | default('localhost') }}"

# repo server certificate SAN if user did not provide own certs
repo_pki_san: "{{ openstack_pki_san | default('DNS:' ~ ansible_facts['hostname
↪'] ~ ',IP:' ~ management_address) }}"
repo_pki_regen_cert: ""

repo_ssl_protocol: "{{ ssl_protocol | default('ALL -SSLv2 -SSLv3 -TLSv1 -
↪TLSv1.1') }}"
# TLS v1.2 and below
repo_ssl_cipher_suite_tls12: "{{ ssl_cipher_suite | default(
↪'ECDH+AESGCM:ECDH+CHACHA20:ECDH+AES256:ECDH+AES128:!aNULL:!SHA1:!AESCCM') }}
↪"
# TLS v1.3
repo_ssl_cipher_suite_tls13: "{{ ssl_cipher_suite_tls13 | default('TLS_AES_
↪128_GCM_SHA256:TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256') }}"

## Define user-provided SSL certificates. Otherwise self-signed certificates
## will be generated for domains defined in ``repo_pki_san`` variables.
# repo_user_ssl_cert: <path to cert on ansible deployment host>
# repo_user_ssl_key: <path to cert on ansible deployment host>
# repo_user_ssl_ca_cert: <path to cert on ansible deployment host>
```

CHAPTER SIX

REQUIRED VARIABLES

None.

CHAPTER

SEVEN

EXAMPLE PLAYBOOK

```yaml
---
- name: Setup repo servers
  hosts: repo_all
  user: root
  roles:
    - role: "repo_server"
      tags: "repo-server"
```