# OpenStack-Ansible Documentation: os_heat role

*Release 18.1.0.dev236*

**OpenStack-Ansible Contributors**

**May 11, 2024**

# CONTENTS


1 Dependencies 1

2 Default variables 3

3 Example playbook 11

4 Tags 13

5 Heat client endpoints 15

# CHAPTER ONE

# DEPENDENCIES

This role needs pip >= 7.1 installed on the target host.

To use this role, define the following variables:

```
# password of the keystone service user for heat
heat_service_password: "secrete"
# password of the admin user for the keystone heat domain
heat_stack_domain_admin_password: "secrete"
# key used for encrypting credentials stored in the heat db
heat_auth_encryption_key: "32characterslongboguskeyvaluefoo"
# password for heat database
heat_container_mysql_password: "secrete"
# password for heat RabbitMQ vhost
heat_rabbitmq_password: "secrete"
# comma-separated list of RabbitMQ hosts
rabbitmq_servers: 10.100.100.101
# Keystone admin user for service, domain, project, role creation
keystone_admin_user_name: "admin"
# Keystone admin password for service, domain, project, role creation
keystone_auth_admin_password: "secrete"
```

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

CHAPTER

TWO

# DEFAULT VARIABLES

```
# The variables file used by the playbooks in the Heat-api group.
# These don't have to be explicitly imported by vars_files: they are␣
↪autopopulated.

# Enable/Disable Ceilometer
heat_ceilometer_enabled: "{{ (groups['ceilometer_all'] is defined) and␣
↪(groups['ceilometer_all'] | length > 0) }}"

## Verbosity Options
debug: False

# Set the host which will execute the shade modules
# for the service setup. The host must already have
# clouds.yaml properly configured.
heat_service_setup_host: "{{ openstack_service_setup_host | default('localhost
↪') }}"
heat_service_setup_host_python_interpreter: "{{ openstack_service_setup_host_
↪python_interpreter | default((heat_service_setup_host == 'localhost') |␣
↪ternary(ansible_playbook_python, ansible_facts['python']['executable'])) }}"

# Set the package install state for distribution packages
# Options are 'present' and 'latest'
heat_package_state: "{{ package_state | default('latest') }}"

# Set installation method.
heat_install_method: "{{ service_install_method | default('source') }}"
heat_venv_python_executable: "{{ openstack_venv_python_executable | default(
↪'python3') }}"

heat_git_repo: https://opendev.org/openstack/heat
heat_git_install_branch: master
heat_upper_constraints_url: "{{ requirements_git_url | default('https://
↪releases.openstack.org/constraints/upper/' ~ requirements_git_install_
↪branch | default('master')) }}"
heat_git_constraints:
  - "--constraint {{ heat_upper_constraints_url }}"

heat_pip_install_args: "{{ pip_install_options | default('') }}"
```

(continues on next page)

(continued from previous page)

```
# Name of the virtual env to deploy into
heat_venv_tag: "{{ venv_tag | default('untagged') }}"
heat_bin: "{{ _heat_bin }}"

heat_fatal_deprecations: False

heat_clients_endpoint: internalURL
heat_clients_heat_endpoint: publicURL

## Database info
heat_db_setup_host: "{{ openstack_db_setup_host | default('localhost') }}"
heat_db_setup_python_interpreter: "{{ openstack_db_setup_python_interpreter |␣
↪default((heat_db_setup_host == 'localhost') | ternary(ansible_playbook_
↪python, ansible_facts['python']['executable'])) }}"
heat_galera_address: "{{ galera_address | default('127.0.0.1') }}"
heat_galera_user: heat
heat_galera_database: heat
heat_galera_use_ssl: "{{ galera_use_ssl | default(False) }}"
heat_galera_ssl_ca_cert: "{{ galera_ssl_ca_cert | default('') }}"
heat_galera_port: "{{ galera_port | default('3306') }}"
heat_db_max_overflow: "{{ openstack_db_max_overflow | default('50') }}"
heat_db_max_pool_size: "{{ openstack_db_max_pool_size | default('5') }}"
heat_db_pool_timeout: "{{ openstack_db_pool_timeout | default('30') }}"
heat_db_connection_recycle_time: "{{ openstack_db_connection_recycle_time |␣
↪default('600') }}"

## Oslo Messaging Info
# RPC
heat_oslomsg_rpc_host_group: "{{ oslomsg_rpc_host_group | default('rabbitmq_
↪all') }}"
heat_oslomsg_rpc_setup_host: "{{ (heat_oslomsg_rpc_host_group in groups) |␣
↪ternary(groups[heat_oslomsg_rpc_host_group][0], 'localhost') }}"
heat_oslomsg_rpc_transport: "{{ oslomsg_rpc_transport | default('rabbit') }}"
heat_oslomsg_rpc_servers: "{{ oslomsg_rpc_servers | default('127.0.0.1') }}"
heat_oslomsg_rpc_port: "{{ oslomsg_rpc_port | default('5672') }}"
heat_oslomsg_rpc_use_ssl: "{{ oslomsg_rpc_use_ssl | default(False) }}"
heat_oslomsg_rpc_userid: heat
heat_oslomsg_rpc_vhost: /heat
heat_oslomsg_rpc_ssl_version: "{{ oslomsg_rpc_ssl_version | default('TLSv1_2
↪') }}"
heat_oslomsg_rpc_ssl_ca_file: "{{ oslomsg_rpc_ssl_ca_file | default('') }}"

# Notify
heat_oslomsg_notify_host_group: "{{ oslomsg_notify_host_group | default(
↪'rabbitmq_all') }}"
heat_oslomsg_notify_setup_host: "{{ (heat_oslomsg_notify_host_group in␣
↪groups) | ternary(groups[heat_oslomsg_notify_host_group][0], 'localhost') }}
↪"
```

(continues on next page)

(continued from previous page)

```
heat_oslomsg_notify_transport: "{{ oslomsg_notify_transport | default('rabbit
↪') }}"
heat_oslomsg_notify_servers: "{{ oslomsg_notify_servers | default('127.0.0.1
↪') }}"
heat_oslomsg_notify_port: "{{ oslomsg_notify_port | default('5672') }}"
heat_oslomsg_notify_use_ssl: "{{ oslomsg_notify_use_ssl | default(False) }}"
heat_oslomsg_notify_userid: "{{ heat_oslomsg_rpc_userid }}"
heat_oslomsg_notify_password: "{{ heat_oslomsg_rpc_password }}"
heat_oslomsg_notify_vhost: "{{ heat_oslomsg_rpc_vhost }}"
heat_oslomsg_notify_ssl_version: "{{ oslomsg_notify_ssl_version | default(
↪'TLSv1_2') }}"
heat_oslomsg_notify_ssl_ca_file: "{{ oslomsg_notify_ssl_ca_file | default('')␣
↪}}"

## (Qdrouterd) integration
# TODO(ansmith): Change structure when more backends will be supported
heat_oslomsg_amqp1_enabled: "{{ heat_oslomsg_rpc_transport == 'amqp' }}"

## Heat User / Group
heat_system_user_name: heat
heat_system_group_name: heat
heat_system_shell: /bin/false
heat_system_comment: heat system user
heat_system_home_folder: "/var/lib/{{ heat_system_user_name }}"

## Default domain
heat_project_domain_name: Default
heat_project_name: admin
heat_user_domain_name: Default

## Stack
heat_stack_domain_admin: stack_domain_admin
heat_stack_owner_name: heat_stack_owner
heat_stack_domain_description: Owns users and projects created by heat
heat_stack_user_domain_name: heat
heat_max_nested_stack_depth: 5

heat_deferred_auth_method: trusts
heat_trusts_delegated_roles: []

## Cinder backups
heat_cinder_backups_enabled: false

# osprofiler
heat_profiler_enabled: false
heat_profiler_trace_sqlalchemy: false

## Auth
heat_service_region: "{{ service_region | default('RegionOne') }}"
```

(continues on next page)

(continued from previous page)

```
heat_service_project_name: "service"
heat_service_user_name: "heat"
heat_service_role_names:
  - admin
  - service
heat_service_token_roles:
  - service
heat_service_token_roles_required: "{{ openstack_service_token_roles_required␣
↪| default(True) }}"
heat_service_project_domain_id: default
heat_service_user_domain_id: default
heat_keystone_auth_plugin: password

## Trustee Auth
heat_service_trustee_project_name: "service"
heat_service_trustee_user_name: "heat"
heat_service_trustee_password: "{{ heat_service_password }}"
heat_service_trustee_project_domain_id: "default"
heat_service_trustee_user_domain_id: "default"
heat_keystone_trustee_auth_plugin: "{{ heat_keystone_trustee_auth_type }}"
heat_keystone_trustee_auth_type: password

## Heat api service type and data
heat_service_name: heat
heat_service_description: "Heat Orchestration Service"
heat_service_port: 8004
heat_service_proto: http
heat_service_publicuri_proto: "{{ openstack_service_publicuri_proto |␣
↪default(heat_service_proto) }}"
heat_service_adminuri_proto: "{{ openstack_service_adminuri_proto |␣
↪default(heat_service_proto) }}"
heat_service_internaluri_proto: "{{ openstack_service_internaluri_proto |␣
↪default(heat_service_proto) }}"
heat_service_type: orchestration
heat_service_publicuri: "{{ heat_service_publicuri_proto }}://{{ external_lb_
↪vip_address }}:{{ heat_service_port }}"
heat_service_publicurl: "{{ heat_service_publicuri }}/v1/%(tenant_id)s"
heat_service_adminuri: "{{ heat_service_adminuri_proto }}://{{ internal_lb_
↪vip_address }}:{{ heat_service_port }}"
heat_service_adminurl: "{{ heat_service_adminuri }}/v1/%(tenant_id)s"
heat_service_internaluri: "{{ heat_service_internaluri_proto }}://{{ internal_
↪lb_vip_address }}:{{ heat_service_port }}"
heat_service_internalurl: "{{ heat_service_internaluri }}/v1/%(tenant_id)s"

## Heat api cfn service type and data
heat_cfn_service_name: heat-cfn
heat_cfn_service_description: "Heat CloudFormation Service"
heat_cfn_service_port: 8000
heat_cfn_service_proto: http
```

(continues on next page)

(continued from previous page)

```
heat_cfn_service_publicuri_proto: "{{ openstack_service_publicuri_proto |␣
↪default(heat_cfn_service_proto) }}"
heat_cfn_service_adminuri_proto: "{{ openstack_service_adminuri_proto |␣
↪default(heat_cfn_service_proto) }}"
heat_cfn_service_internaluri_proto: "{{ openstack_service_internaluri_proto |␣
↪default(heat_cfn_service_proto) }}"
heat_cfn_service_type: cloudformation
heat_cfn_service_publicuri: "{{ heat_cfn_service_publicuri_proto }}://{{␣
↪external_lb_vip_address }}:{{ heat_cfn_service_port }}"
heat_cfn_service_publicurl: "{{ heat_cfn_service_publicuri }}/v1"
heat_cfn_service_adminuri: "{{ heat_cfn_service_adminuri_proto }}://{{␣
↪internal_lb_vip_address }}:{{ heat_cfn_service_port }}"
heat_cfn_service_adminurl: "{{ heat_cfn_service_adminuri }}/v1"
heat_cfn_service_internaluri: "{{ heat_cfn_service_internaluri_proto }}://{{␣
↪internal_lb_vip_address }}:{{ heat_cfn_service_port }}"
heat_cfn_service_internalurl: "{{ heat_cfn_service_internaluri }}/v1"

## Heat wait and metadata server
heat_waitcondition_server_uri: "{{ heat_cfn_service_publicuri_proto }}://{{␣
↪external_lb_vip_address }}:{{ heat_cfn_service_port }}"
heat_waitcondition_server_url: "{{ heat_waitcondition_server_uri }}/v1/
↪waitcondition"
heat_metadata_server_url: "{{ heat_cfn_service_publicuri_proto }}://{{␣
↪external_lb_vip_address }}:{{ heat_cfn_service_port }}"

# If the following variables are unset in user_variables, the value set will␣
↪be half the number of available VCPUs
# heat_engine_workers: 4
# heat_api_workers: 4

## Cap the maximum number of threads / workers when a user value is␣
↪unspecified.
heat_api_threads_max: 16
heat_api_threads: "{{ [[(ansible_facts['processor_vcpus']//ansible_facts[
↪'processor_threads_per_core'])|default(1), 1] | max * 2, heat_api_threads_
↪max] | min }}"

heat_service_in_ldap: "{{ service_ldap_backend_enabled | default(False) }}"

## Plugin dirs
heat_plugin_dirs:
  - /usr/lib/heat
  - /usr/local/lib/heat

## Policy vars
# Provide a list of access controls to update the default policy.json with.␣
↪These changes will be merged
# with the access controls in the default policy.json. E.g.
#heat_policy_overrides:
```

(continues on next page)

(continued from previous page)

```
#  "cloudformation:ListStacks": "rule:deny_stack_user"
#  "cloudformation:CreateStack": "rule:deny_stack_user"

# Common pip packages
heat_pip_packages:
  - cryptography
  - "git+{{ heat_git_repo }}@{{ heat_git_install_branch }}#egg=openstack-heat"
  - keystonemiddleware
  - osprofiler
  - PyMySQL
  - pymemcache
  - python-glanceclient
  - python-heatclient
  - python-keystoneclient
  - python-memcached
  - python-neutronclient
  - python-novaclient
  - python-openstackclient
  - python-swiftclient
  - python-troveclient
  - systemd-python

# Memcached override
heat_memcached_servers: "{{ memcached_servers }}"

# Specific pip packages provided by the user
heat_user_pip_packages: []

heat_optional_oslomsg_amqp1_pip_packages:
  - oslo.messaging[amqp1]

heat_api_init_overrides: {}
heat_api_cfn_init_overrides: {}
heat_engine_init_overrides: {}

## Service Name-Group Mapping
heat_services:
  heat-api:
    group: heat_api
    service_name: heat-api
    init_config_overrides: "{{ heat_api_init_overrides }}"
    start_order: 2
    wsgi_app: True
    wsgi_name: heat-wsgi-api
    uwsgi_overrides: "{{ heat_api_uwsgi_ini_overrides }}"
    uwsgi_port: "{{ heat_service_port }}"
    uwsgi_bind_address: "{{ heat_api_uwsgi_bind_address }}"
  heat-api-cfn:
    group: heat_api_cfn
```

(continues on next page)

(continued from previous page)

```
    service_name: heat-api-cfn
    init_config_overrides: "{{ heat_api_cfn_init_overrides }}"
    start_order: 3
    wsgi_app: True
    wsgi_name: heat-wsgi-api-cfn
    uwsgi_overrides: "{{ heat_api_cfn_uwsgi_ini_overrides }}"
    uwsgi_port: "{{ heat_cfn_service_port }}"
    uwsgi_bind_address: "{{ heat_api_cfn_uwsgi_bind_address }}"
  heat-engine:
    group: heat_engine
    service_name: heat-engine
    execstarts: "{{ heat_bin }}/heat-engine"
    init_config_overrides: "{{ heat_engine_init_overrides }}"
    start_order: 1

# Required secrets for the role
heat_required_secrets:
  - keystone_auth_admin_password
  - heat_stack_domain_admin_password
  - heat_auth_encryption_key
  - heat_container_mysql_password
  - heat_rabbitmq_password
  - heat_service_password
  - memcached_encryption_key

# uWSGI Settings
heat_api_uwsgi_ini_overrides: {}
heat_api_cfn_uwsgi_ini_overrides: {}
heat_wsgi_processes_max: 16
heat_wsgi_processes: "{{ [[ansible_facts['processor_vcpus']|default(1), 1] |␣
↪max * 2, heat_wsgi_processes_max] | min }}"
heat_wsgi_threads: 1
heat_api_uwsgi_bind_address: "{{ openstack_service_bind_address | default('0.
↪0.0.0') }}"
heat_api_cfn_uwsgi_bind_address: "{{ openstack_service_bind_address | default(
↪'0.0.0.0') }}"

## Tunable overrides
heat_heat_conf_overrides: {}
heat_api_paste_ini_overrides: {}
heat_default_yaml_overrides: {}
heat_aws_cloudwatch_alarm_yaml_overrides: {}
heat_aws_rds_dbinstance_yaml_overrides: {}
heat_policy_overrides: {}
```

CHAPTER

THREE

# EXAMPLE PLAYBOOK

```yaml
---
- name: Install heat server
  hosts: heat_all
  user: root
  roles:
    - { role: "os_heat", tags: [ "os-heat" ] }
  vars:
    external_lb_vip_address: 172.16.24.1
    internal_lb_vip_address: 192.168.0.1
    heat_galera_address: "{{ internal_lb_vip_address }}"
    keystone_admin_user_name: admin
    keystone_admin_tenant_name: admin
    galera_root_user: root
  vars_prompt:
    - name: "galera_root_password"
      prompt: "What is galera_root_password?"
```

# CHAPTER FOUR

# TAGS

This role supports two tags: `heat-install` and `heat-config`. The `heat-install` tag can be used to install and upgrade. The `heat-config` tag can be used to maintain the configuration of the service.

CHAPTER

FIVE

# HEAT CLIENT ENDPOINTS

When your VMs need to talk to your API, you might have to change the Heat config. By default Heat is configured to use the internal API endpoints. Should instances or created containers need to access the API (e.g. Magnum, Heat Signaling) the public endpoints will need to be used as in the following example:

```
heat_heat_conf_overrides:
  clients_keystone:
    endpoint_type: publicURL
    auth_uri: "{{ keystone_service_publicurl }}"
```