
ANSIBLE

Agenda

- Optimizing Ansible Execution

Where do I start to optimize Ansible?

- There isn’t a single way to optimize Ansible.
- From a networking perspective, the closest the target host are from the control node, the better.
- Try to use recent versions.
- Disable facts gathering if you are not using them
- Reuse Gathered facts with Fact caching
- Increase parallelism
- Avoid Loops with the Package Manager Modules

Where do I start to optimize Ansible?

- Copy files to target hosts efficiently
- Templates
- Enable pipelining
- Use plug-ins and callbacks to measure the time each task is taking

Reuse gathered facts with Fact caching

- Ansible uses cache plugins to store gathered facts gathered by a playbook execution.
- Fact cache can be used to limit how many times you need to gather facts by reusing facts gathered in previous executions.
- If you have a playbook with multiple plays, the first play could gather facts about all hosts, and then facts gathering from subsequent plays are disabled and use the facts from the first play.
- Alternatively you can use smart gathering by setting the gathering attribute in the ansible config file to smart, e.g:

```
[defaults]
gathering=smart
```

Limit facts that are gathered

- Another possibility is to explicitly gather a subset of all possible facts.
- Subsets available are:
  - all
  - min
  - hardware
  - network
  - virtual
  - ohai
  - facter

Modify Parallelism

- The forks parameter controls how many collections in parallel Ansible establishes when connecting to target hosts.
- It’s set to 5 hosts by default.
- Increasing this value will cause that Ansible “covers” more hosts at once
- Setting this value too high may case that the control host slows down, therefore it’s recommended to start with 50 for example, and then adjust accordingly after trial and error.

Avoid loops with package manager modules

- It’s recommended to not use loops to iterate over a list of packages that should be passed to any package manage module.
- Most of these modules allow you to pass a list of items instead in case multiple packages should be installed at once.

```yaml
tasks:
  - name: Install base packages
    ansible.builtin.yum:
      name:
        - vim
        - jq
        - podman
        - skopeo
      state: present
```

```
$ yum install -y vim jq podman skopeo
```

```yaml
tasks:
  - name: Install base packages
    ansible.builtin.yum:
      name: "{{ item }}"
      state: started
      enabled: true
    loop:
      - vim
      - jq
      - podman
      - skopeo
```

Copy files to target hosts efficiently

- The ansible.builtin.copy module recursively copies the files and directories to the target hosts.

- The ansible.posix.synchronize module is more efficient since it uses rsync in the background

Templates

- While the ansible.builtin.lineinfile modify files to either eliminate or add a line to a file, when used with a loop it is not optimal.

- Therefore it’s recommended to substitute it in those cases with the ansible.builtin.template module instead to copy a template of a file.

Pipelining

- Pipelining in Ansible consist of enabling this feature which causes that Ansible establishes fewer SSH connections to the target hosts in order to run a task.
- To enable pipelining set the ANSIBLE_PIPELINING environment variable to true in the execution-environment section of the ansible-navigator.yml config file:

```yaml
execution-environment:
  container-engine: podman
  enabled: true
  image: ghcr.io/ansible/community-ansible-dev-tools:latest
  environment-variables:
    set:
      ANSIBLE_PIPELINING: true
```

Use plug-ins and callbacks to measure tasks

- Both plug-ins and callbacks are used to extend Ansible functionality.
- This may be from changing the output of command line tools to other functionality.
- The timer plug-in shows the playbook execution time in the output of the ansible-navigator command.
- Ansible ships with callbacks by default, you can check them by running: ansible-navigator doc -t callback -l -m stdout command.
- To enable a specific callback add it like this in ansible.cfg:

```
[defaults]
callbacks_enabled=timer, profile_tasks
```

Exercise