

Cheat Sheet — Ansible for Proxmox VE

credativ GmbH

Command line

```
ansible <pattern> -m <module> -a '<args>'
ansible-playbook site.yml
ansible-playbook site.yml --check --diff
ansible-playbook site.yml --limit pve02
ansible-playbook site.yml --tags repos
ansible-inventory --graph
ansible-inventory --host pve01
ansible-doc <module>
ansible-doc -l community.proxmox
ansible-galaxy collection install -r requirements.yml
ansible-lint site.yml roles/
```

Run states

State	Meaning
ok	already in the desired state
changed	something was changed
skipped	a when: excluded the host
failed	the task did not succeed
unreachable	no connection at all

changed=0 on the second run = idempotent.

Inventory

```
all:
  children:
    pve:
      hosts:
        pve01: {ansible_host: 192.168.0.1}
      vars:
        ansible_user: root
    pve_primary:
      hosts: {pve01: null}
```

Variables: inventory/group_vars/<group>.yaml, inventory/host_vars/<host>.yaml

ansible.cfg

```
[defaults]
inventory          = inventory/hosts.yml
host_key_checking  = False
interpreter_python  = auto_silent

[ssh_connection]
pipelining         = True
```

Play skeleton

```
- name: What this play does
  hosts: pve
  gather_facts: true
```

```
module_defaults: {}
handlers:
  - name: Reload something
    ansible.builtin.command: ifreload -a
tasks:
  - name: What this task does
    ansible.builtin.apt:
      name: "{{ pve_packages }}"
      state: present
      notify: Reload something
```

Guards for command / shell

```
args:
  creates: /etc/pve/corosync.conf
when: some_condition
changed_when: false
failed_when: rc not in [0, 2]
check_mode: false
```

Role layout

```
roles/pve_node/
defaults/main.yml  # overridable values
vars/main.yml      # internal values
tasks/main.yml     # task list only
handlers/main.yml
templates/*.j2
files/
```

Precedence (weak → strong): defaults → group_vars → host_vars → vars/ → play vars: → --extra-vars

Secrets

In this lab: pve_root_password set directly in inventory/group_vars/pve.yaml.

In production: Ansible Vault or your CI/CD secret store.

API modules: the pattern

```
- hosts: localhost
  connection: local
  module_defaults:
    group/community.proxmox.proxmox:
      api_host: 192.168.0.1
      api_user: automation@pve
      api_token_id: ansible
      api_token_secret: "{{ pve_api_token_secret }}"
      validate_certs: false
```

Or per task on a Proxmox host: delegate_to: localhost

community.proxmox: the ones you need

Module	Purpose
proxmox_cluster	create / join a cluster
proxmox_cluster_join_info	fingerprint, node list
proxmox_cluster_status_info	quorum, membership
proxmox_storage	dir, nfs, pbs, zfspool, ...
proxmox_user / _group / _role	accounts, tokens
proxmox_access_acl	permissions
proxmox_kvm	virtual machines
proxmox	LXC containers
proxmox_template	ISO / container templates
proxmox_backup	run a backup
proxmox_node_network	bridges, bonds, VLANs
proxmox_pool / _pool_member	resource pools
*_info	read-only variants

Cluster, in one playbook

```
- hosts: pve_primary
  tasks:
    - community.proxmox.proxmox_cluster:
      state: present
      cluster_name: training
      link0: "{{ ansible_host }}"
      api_host: "{{ ansible_host }}"
      api_user: root@pam
      api_password: "{{ pve_root_password }}"
      validate_certs: false
      delegate_to: localhost

- hosts: pve_secondary
  serial: 1 # one at a time
  tasks:
    - community.proxmox.proxmox_cluster:
      state: present
      master_ip: 192.168.0.1
      fingerprint: "{{ fp }}"
      ...
      delegate_to: localhost
```

Dynamic inventory

File name **must** end in .proxmox.yml:

```
plugin: community.proxmox.proxmox
url: https://192.168.0.1:8006
user: automation@pve
token_id: ansible
token_secret:
  "{{ lookup('env', 'PVE_TOKEN_SECRET') }}"
validate_certs: false
want_facts: true
```

Groups generated: proxmox_nodes, proxmox_all_qemu, proxmox_all_lxc, proxmox_all_running, proxmox_all_stopped, proxmox_<node>_qemu, proxmox_pool_<pool>

Proxmox VE commands for verification

```
pvecm status # quorum, votes, cluster name
pvecm nodes # members
pvesm status # storages
pveum user list # users
pveum acl list # permissions
qm list ; qm config <vmid>
pct list
vzdump --help
ifreload -a # apply network changes
ls /etc/pve/nodes # pmxcfs replication
```

When something fails

```
ansible-playbook site.yml -vvv # incl. SSH conversation
ansible-inventory --graph -vvv # why a source is ignored
ansible <host> -m ansible.builtin.setup # all facts
ansible-doc -s <module> # option skeleton
journalctl -u pveproxy -u pvedaemon # on the node
```

Documentation

- Ansible: <https://docs.ansible.com/ansible/latest/>
- Collection: <https://docs.ansible.com/ansible/latest/collections/community/proxmox/>
- Collection source: <https://github.com/ansible-collections/community.proxmox>
- Proxmox VE: <https://pve.proxmox.com/pve-docs/>
- Your own cluster's API: <https://<node>:8006/pve-docs/api-viewer/>
- ansible-lint: <https://ansible.readthedocs.io/projects/lint/>

Golden rules

- Describe the **state**, not the steps
- A **module** beats a command; a command needs a guard
- Node = Debian** over SSH, **node = Proxmox VE** over the API
- check --diff **before** the real run
- Networking changes: **one node at a time**, console at hand
- Tokens, not root passwords