



Ansible II

Dónal Cunningham, HEAnet
Orchestration, Automation and Virtualisation (OAV)
Training Development Team

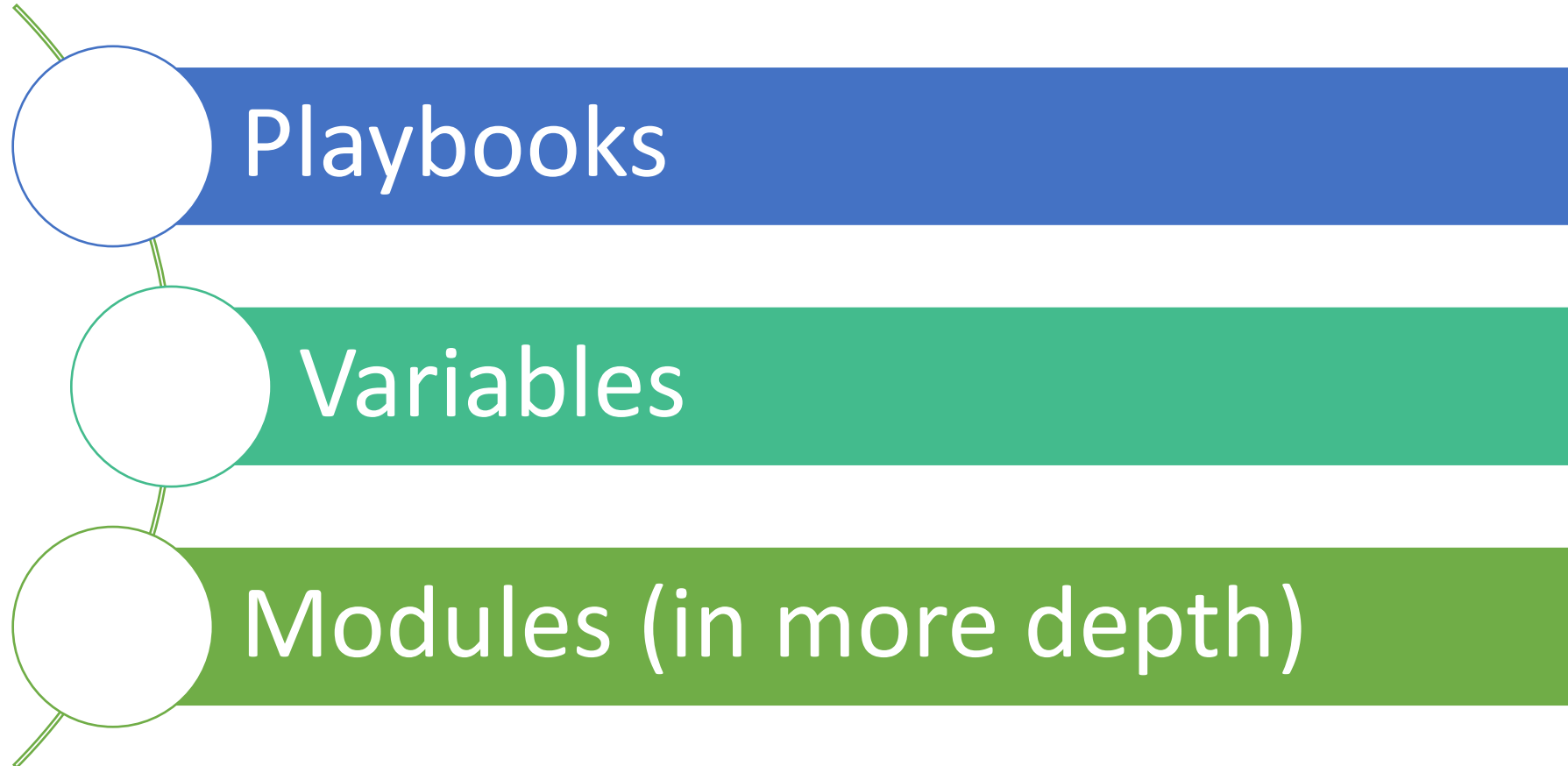
OAV training by the GÉANT project:

<https://wiki.geant.org/display/NETDEV/OAV+Training+Portal>

oav@lists.geant.org

www.geant.org

In This Section...



YAML recap

```
---
devices:
  - routers
  - switches
  - firewalls

vlan:
  name:      DMZ
  id:        10
  subnet:    10.1.20.0/24
  acl:       DMZ_from_internet_v4
```

v lans:

- Management:

```
  vlan_id:          99
  description:      'Management OOB'
  ipv4_subnets:    193.1.219.0/24
```

- DNS_AUTH:

```
  vlan_id:          10
  description:      'Authoritative DNS servers'
  ipv4_subnets:    193.1.220.0/26
  dhcp_servers:
    - 193.1.2.1
    - 193.1.3.1
```

Playbooks

```
---  
# Oh look, a comment...  
# ...spread out over multiple lines  
  
- name: Set up Apache           # Or nginx, or Mongoose  
  hosts: webservers  
  tasks:  
    - name: install Apache  
    - name: generate Apache config file  
    - name: download web content to relevant directory  
    - name: restart Apache  
    - name: eat cake
```

```
$ ansible-playbook -i hosts setup_apache.yml
PLAY [Set up Apache] *****

TASK [install Apache] *****
ok: [web1]
ok: [web2]

TASK [generate Apache config file] *****
ok: [web1]
ok: [web2]

PLAY RECAP *****
web1 : ok=2  changed=2  unreachable=0  failed=0
web2 : ok=2  changed=2  unreachable=0  failed=0
```

Common Errors

- Inventory problems
 - Typos/missing hosts
 - Hosts not in groups
- SSH problems
 - Incorrect IP address
 - Incorrect credentials
 - Incorrect permissions to run commands

```
$ ssh ansible@cw-leaf2
```

```
The authenticity of host 'cw-leaf2' can't be established.
```

```
ECDSA key fingerprint is ...
```

```
Are you sure you want to continue connecting (yes/no)?
```

```
$ ssh ansible@pw-leaf3
```

```
Unable to negotiate with 1.2.3.4 port 22: no matching key  
exchange method found.
```

```
Their offer: diffie-hellman-group14-sha1
```

```
$ ssh -vvv ansible@wt-leaf3
```

```
...
```

```
debug1: Connecting to wt-leaf2 port 22.
```

```
ssh: connect to host wt-leaf2 port 22: Operation timed out
```



```
$ ansible-playbook ping_unknown_host.yaml
[WARNING]: Could not match supplied host pattern,
ignoring: fakeserver

PLAY [A play to run the ping module] *****

TASK [Gathering Facts] *****
ok: [web1]

TASK [A task to invoke the ping module] *****
ok: [web1]

PLAY RECAP *****
web1: ok=2    changed=0    unreachable=0    failed=0
```

```
$ ansible-playbook ping_unreachable_host.yaml
PLAY [A play to run the ping module]
*****

TASK [Gathering Facts] *****
fatal: [veos]: UNREACHABLE! => changed=false
  msg: 'Failed to connect to the host via ssh: blah blah'
ok: [web1]

TASK [A task to invoke the ping module] *****
ok: [web1]

PLAY RECAP *****
web1 : ok=2    changed=0    unreachable=0    failed=0
veos  : ok=0    changed=0    unreachable=1    failed=0
```

Variables

```
- name: Set up Apache
  hosts: webservers
  vars:
    http_port: 80
```

```
- name: Set up Apache
  hosts: webservers
  vars_files:
    - external_vars.yml
```

Variables on the Command Line

```
$ ansible-playbook --extra-vars "hosts=dbservers"
```

```
$ ansible-playbook --extra-vars "@my_variables.json"
```

```
$ tree
.
├── ansible.cfg
├── hosts
├── group_vars
│   ├── all.yaml
│   └── webservers.yaml
├── host_vars
│   ├── web1.yaml
│   ├── web2.yaml
│   ├── db1.yaml
│   └── db2.yaml
├── setup_apache_playbook.yaml
└── setup_mysql_playbook.yaml
```

```
logging_level: 4
syslog_level: 4
log_buffer: 45000
```

radius_host:

- host: 193.1.219.33
- host: 193.1.248.38

logging:

- host: 193.1.219.117
- host: 193.1.219.36
- host: 193.1.248.123

dns:

- server: 193.1.186.2
- server: 193.1.186.3

```
domain_name: services.hea.net
```

```
vlans:
- name: vlan 1234
  description: "[UNI] Panopticon"
  port_type: svi
  ip_address: 192.0.2.2/28
  ipv6_address: 2001:db8:6:1705::1/64
  ip_helper_address:
    - address: 192.0.2.254
    - address: 192.0.2.253
#
  vrrp_group: 1
  vrrp_ip: 192.0.2.1
  vrrp_priority: 100
  vrrp_auth: sooper_seekrit
  vrrp_preempt: True
#
  shutdown: False
  default: False
```

Variable Referencing

```
physical_attributes:  
  mtu:          1500  
  speed:        auto  
  rootguard:    True
```

```
physical_attributes.speed
```

...is the same thing as...

```
physical_attributes['speed']
```

...which is "auto"

- **name**: Copy the relevant files to the target

vars:

source_file: foo.conf

destination_directory: /tmp/foo

tasks:

- **name**: copy foo.conf to server

copy:

src: "{{ source_file }}"

dest: "{{ destination_directory }}"

owner: foo

group: foo

mode: "0644"

backup: yes


Modules

A Documentation ANSIBLEFEST PRODUCTS

- Release and maintenance
- Testing Strategies
- Sanity Tests
- Frequently Asked Questions
- Glossary
- Ansible Reference: Module Utilities
- Special Variables
- Red Hat Ansible Tower
- Ansible Automation Hub
- Logging Ansible output

ROADMAPS

- Ansible Roadmap

 **Red Hat**
Ansible Automation Platform

Extend the power of Ansible to your entire team

[Try it free](#)

```
- name: Copy a new "sudoers" file into place, after passing validation with visudo
ansible.builtin.copy:
  src: /mine/sudoers
  dest: /etc/sudoers
  validate: /usr/sbin/visudo -csf %s

- name: Copy a "sudoers" file on the remote machine for editing
ansible.builtin.copy:
  src: /etc/sudoers
  dest: /etc/sudoers.edit
  remote_src: yes
  validate: /usr/sbin/visudo -csf %s

- name: Copy using inline content
ansible.builtin.copy:
  content: '# This file was moved to /etc/other.conf'
  dest: /etc/mine.conf

- name: If follow=yes, /path/to/file will be overwritten by contents of foo.conf
ansible.builtin.copy:
  src: /etc/foo.conf
  dest: /path/to/link # link to /path/to/file
  follow: yes

- name: If follow=no, /path/to/link will become a file and be overwritten by contents of foo.conf
ansible.builtin.copy:
  src: /etc/foo.conf
  dest: /path/to/link # link to /path/to/file
  follow: no
```

Return Values

Common return values are documented [here](#), the following are the fields unique to this module:

Key	Returned	Description
backup file	changed and	Name of backup file created.

Playbook with Modules

```
---  
- name: Install Apache2 on all webservers  
  hosts: webservers  
  become: true # Must have root privileges  
  tasks:  
    - name: Install Apache  
      apt: pkg=apache2 state=latest # httpd if using yum  
    - name: Generate config file  
      copy: src=httpd.conf dest=/etc/httpd.conf  
    - name: Restart Apache  
      service: name=apache2 state=started
```

```
---  
- name: Install mod_rewrite on all webservers  
  hosts: webservers  
  become: true  
  tasks:  
    - name: Install Apache  
      apt: pkg=apache2 state=latest  
  
    - name: enable mod_rewrite  
      apache2_module: name=rewrite state=present  
  notify:  
    - restart_apache2  
  
handlers:  
  - name: restart_apache2  
    service: name=apache2 state=restarted
```

```
---  
- name: Check the version of code on a switch  
  hosts: veos  
  gather_facts: no          # or False  
  connection: local  
  tasks:  
    - name: Gather version from switches  
      raw: 'show version'  
  
# If using more recent versions of Ansible, use...  
# delegate_to: localhost  
#           ...instead of...  
# connection: local
```

```
---  
- name: Check the version of code on all switches  
  hosts: veos  
  gather_facts: no # or False  
  connection: local  
  tasks:  
    - name: Gather version from switches  
      raw: 'show version| include software'  
      register: show_version  
  
    - name: Display the version  
      debug: var=show_version
```

```
TASK [Task 1.1 - Gather version from switches] *****
changed: [veos]

TASK [Task 1.2 - Print output] *****
ok: [veos] => {
  "showvers": {
    "changed": true,
    "failed": false,
    "rc": 0,
    "stderr": "Shared connection to localhost closed.\r\n",
    "stderr_lines": [
      "Shared connection to localhost closed."
    ],
    "stdout": "Software image version: 4.25.2F\r\n",
    "stdout_lines": [
      "Software image version: 4.25.2F"
    ]
  }
}
```

Indexing into stdout_lines

```
- name:          Display the version  
  debug:        var=showversion.stdout_lines[0]
```

```
ok: [veos] => {  
  "showvers.stdout_lines[0]": "Software image version: 4.25.2F"  
}
```



```
- name: Print out PIM Rendezvous Point
  debug:
    msg:
      - Detected a PTP server.
      - Its address is {{ ptp_server }}
  when: ptp_server is defined

- name: Print out a variable
  ansible.builtin.debug:
    var: command_result
```

playbook ↑

ansible.cfg ↓

```
stdout_callback = yaml # Or minimal, or slack, or log_plays
```

Vendor modules – Config and Commands

```
- name:          Get the version
  arista.eos.eos_command:
    commands:
      - 'show version | include software'
      - 'show uptime'
  register: eos_commands
```

- **name:** Get the JunOS BGP neighbours

juniper.junos.junos_command:

commands: 'show bgp summary'

register: junos_bgp_summary

when: ansible_network_os == 'junos'

- **name:** Get the IOS BGP neighbours

cisco.ios.ios_command:

commands: 'show ip bgp summary'

register: ios_bgp_summary

when: ansible_network_os == 'ios'

```
- name: Task 1.1 - Gather version from switches
  arista.eos.eos_command:
    commands:
      - 'show version | include software'
      - 'show uptime'
  register: eos_commands
```

```
TASK [Task 1.1 - Gather version from switches] *****
fatal: [veos]: FAILED! => {"changed": false, "msg": "Connection
type ssh is not valid for this module"}
```

```
---  
- name: Play 1 - Show the software version on an Arista switch  
  hosts: veos  
  gather_facts: no  
  vars:  
    ansible_connection: network_cli  
    ansible_network_os: eos  
  tasks:  
    - name: Task 1.1 - Gather version from switches  
      eos_command:  
        commands:  
          - 'show version | include software'  
          - 'show uptime'  
      register: eos_commands
```

```
$ ansible-playbook -i ./hosts ./show_version_and_uptime.yaml
```

```
PLAY [Play 1 - Show the software version on an Arista switch] *****
```

```
TASK [Task 1.1 - Gather version from switches] *****
```

```
ok: [veos]
```

```
TASK [Task 1.2 - Print output] *****
```

```
ok: [veos] => {
```

```
  "eos_commands.stdout_lines": [
```

```
    [
```

```
      "Software image version: 4.25.2F"
```

```
    ],
```

```
    [
```

```
      "00:18:25 up 9:07, 1 user, load average: 0.04, 0.05, 0.00"
```

```
    ]
```

```
  ]
```

```
}
```

```
PLAY RECAP *****
```

```
veos-1: ok=2    changed=0    unreachable=0    failed=0
```

Setting System Variables – Inventory and Playbook Examples

```
[junos_devices:vars]
ansible_connection: netconf
ansible_network_os: junos
```

```
---
- name: Set hostname
  hosts: veos
  vars:
    ansible_connection: netconf
    ansible_network_os: eos
  eos_config:
    backup: True
    commands: 'hostname {{ hostname }}'
```

Config vs. Command

```
---  
- name: Set the loopback description  
  cisco.ios.ios_config:  
    lines:      description "Loopback for BGP sessions"  
    parents:    interface Loopback0  
    register:   ios_set_description  
    backup:     True  
  
- name: Save the config  
  cisco.ios.ios_command:  
    commands:   copy running-config startup-config  
    register:   ios_save_config
```


Thank you

If you have any questions, or would like to discuss any of these topics further, please email:

oav@lists.geant.org

This material is distributed under a
CC BY-NC-SA licence

www.geant.org

