Ansible I

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

Section I – Ansible Basics

What is Ansible?

How Ansible works

Installing Ansible (and what you get with it)

Ansible settings

Inventories

Module basics

GÉANT

Framework comparison - Ansible vs. SaltStack

Ansible advantages

- Simple to setup
- Large library of modules
- Large community
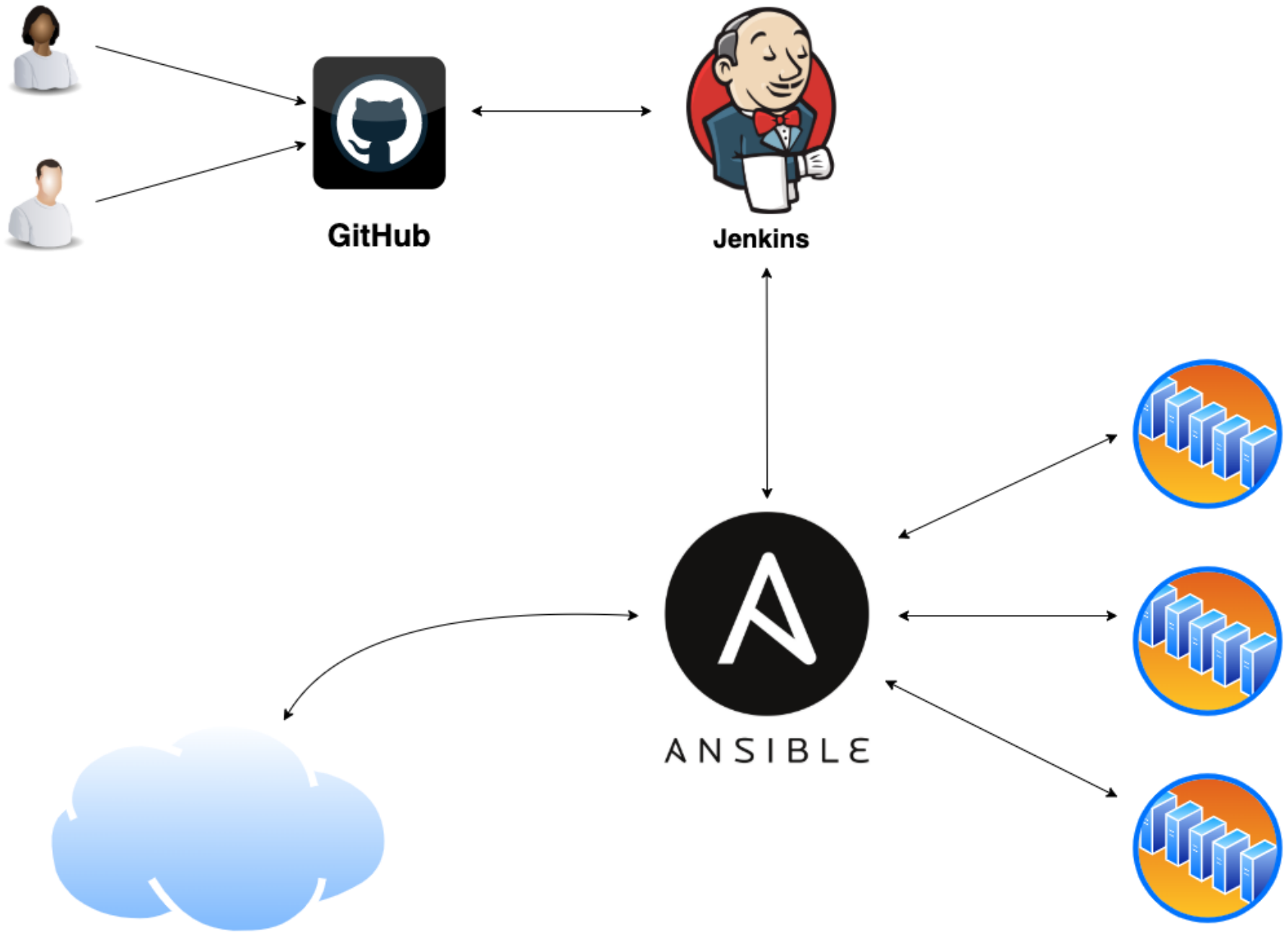- Run-anywhere design

Ansible disadvantages

- Everything done via ssh
- Limited Windows support
- Limited parallelisation by default

SaltStack advantages

- Fast
- Good Windows support
- Large community
- Distributed design

SaltStack disadvantages

- Setup non-trivial
- Proxies needed for network hosts

GitHub

Jenkins

ANSIBLE

www.geant.org

GÉANT

Automation and Networking in the 2020s

Configure via frameworks, not CLI

Network as Code

Declarative vs. Imperative

Idempotence

What Is Ansible?

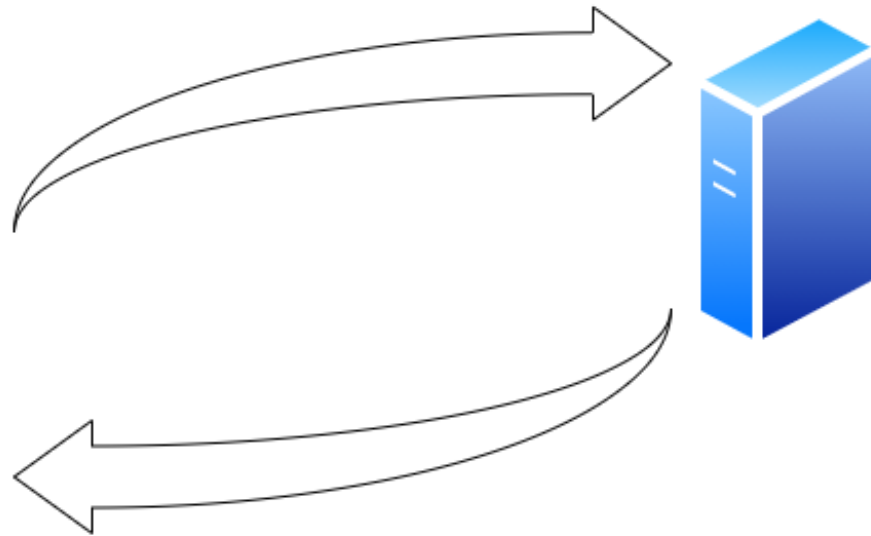Automation Framework

Manages nodes & services
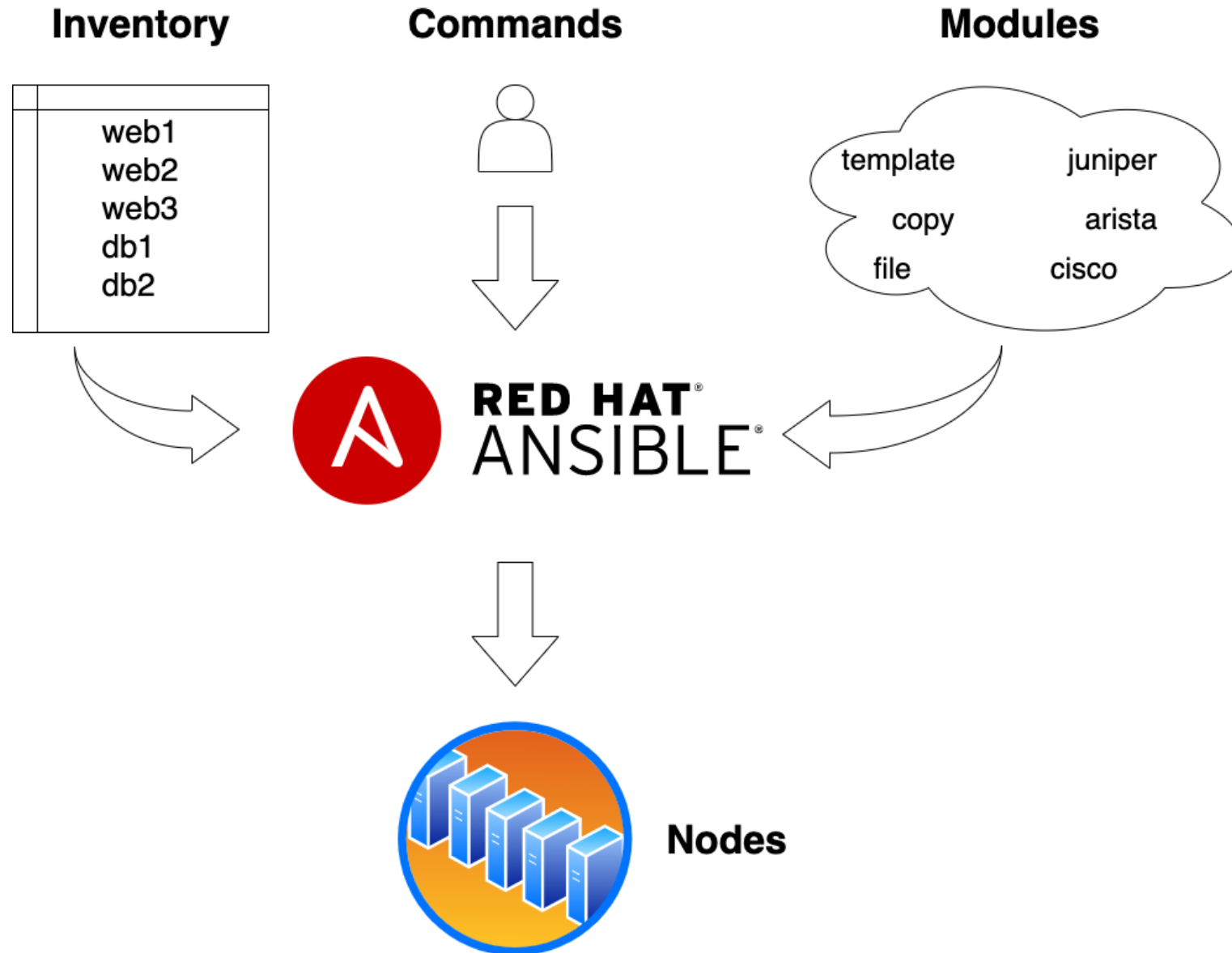
Runs on a UNIX host

CLI commands

YAML for tasks

6

GÉANT

How Ansible Works

1 - copy Python code to server

2 - run Python code on server

3 - retrieve results from server (and delete code)

www.geant.org

How Ansible Works (ctd.)

Ansible Settings

```
# My own config file, thank goodness!

[defaults]


host_key_checking = False
stdout_callback = yaml


inventory = hosts
```

Example ansible configuration file

www.geant.org

GÉANT

Warning: Cows Ahead

```
 _____
/ PLAY [Play 1 - Show the software version on an Arista \
\ switch]                                                /
 ------------------------------------------------------------------
        \   ^__^
         \  (oo)_____
            (__)\       )\/\
                ||----w |
                ||     ||
```

```
# To supress, put this in your ansible.cfg

nocows=True
```

Installing Ansible

https://docs.ansible.com/ansible/latest/installation_guide/

- ansible-core: built-in plugins
  - versions are 2.10 → 2.11 → 2.12

- ansible : selected Collections
  - versions are 2.10 → 3.0 → 4.0

Ansible Common Options

- `-i INVENTORY_NAME`

- `-m MODULE_NAME`
- `-a ARGUMENTS_FOR_A_MODULE`

- `--version`

- `--help`

```
$ ansible --version
-bash: ansible: command not found


$ ansible --version
ansible [core 2.11.6]
    config file = /Users/donal/ansible.cfg
…
    executable location = /usr/local/bin/ansible
    python version = 3.10.0
    jinja version = 3.0.2
```

GÉANT

```
$ ansible-playbook --help
usage: ansible [-h] [--version] [-v] [-b] [--become-method BECOME_METHOD] [--become-user BECOME_USER]

               [-K] [-i INVENTORY] [--list-hosts]

               [-l SUBSET] [-P POLL_INTERVAL] [-B SECONDS] [-o] [-t TREE] [-k] [--private-key PRIVATE_KEY_FILE]

                 [-u REMOTE_USER] [-c CONNECTION] [-T TIMEOUT] [--ssh-common-args SSH_COMMON_ARGS]

               [--sftp-extra-args SFTP_EXTRA_ARGS] [--scp-extra-args SCP_EXTRA_ARGS]

               [--ssh-extra-args SSH_EXTRA_ARGS] [-C] [--syntax-check] [-D] [-e EXTRA_VARS]

               [--vault-id VAULT_IDS] [--ask-vault-password | --vault-password-file VAULT_PASSWORD_FILES]

               [-f FORKS] [-M MODULE_PATH] [--playbook-dir BASEDIR]

               [--task-timeout TASK_TIMEOUT] [-a MODULE_ARGS] [-m MODULE_NAME]

               pattern


Define and run a single task 'playbook' against a set of hosts
positional arguments:
  pattern               host pattern
optional arguments:
  --ask-vault-password, --ask-vault-pass

                        ask for vault password
  --list-hosts          outputs a list of matching hosts; does not execute

                        anything else
```

```
$ tree /etc/ansible
/etc/ansible
├── ansible.cfg
├── hosts
└── roles


$ ansible<TAB>
ansible                 ansible-connection
ansible-doc             ansible-inventory
ansible-pull            ansible-config
ansible-console         ansible-galaxy
ansible-playbook        ansible-vault
```

Inventories

```
loadbalancer1

[webservers]
web[1:10]_dublin

[dbservers]
db[1:2]_dublin

[all_servers:children]
webservers
dbservers
```

Example INI ansible inventory file

GÉANT

Inventories (ctd.)

```yaml
all:
  hosts:
    loadbalancer1:
  children:
    webservers:
      hosts:
        web[01:10]_dublin:
    dbservers:
      hosts:
        db[01:02]_dublin:
    all_servers:
      children:
        webservers:
        dbservers:
```

Example YAML ansible inventory file

GÉANT

```
[all:vars]
ansible_user=zork
ansible_ssh_pass=swordfish

[webservers:vars]
ansible_user=nginx_test


[veos_hosts]
veos        ansible_host=localhost ansible_port=9022
```

Section1/show_version/hosts

```yaml
all:
  vars:
    ansible_user: zork
    ansible_ssh_pass: swordfish
  hosts:
    veos:
      vars:
        ansible_host: localhost
        ansible_port: 9022
  children:
    webservers:
      vars:
        ansible_user: nginx_test
```

Example YAML ansible variables

GÉANT

```
$ ansible-inventory --graph -i hosts.yaml
@all:
  |--@all_servers:
  |    |--@dbservers:
  |    |    |--db01_dublin
  |    |    |--db02_dublin
  |    |--@webservers:
  |    |    |--web01_dublin
  |    |    |--web02_dublin
  |--@dbservers:
  |    |--db01_dublin
  |    |--db02_dublin
  |--@ungrouped:
  |    |--loadbalancer1
  |--@webservers:
  |    |--web01_dublin
  |    |--web02_dublin
```

Example ansible inventory – graph format

Modules

- Built-in
  - raw, template, file, copy, shell
- Industry-contributed
  - Cisco   `ios_command`
  - Juniper   `junos_config`
  - Arista   `eos_lacp`
  - Cloud   `aws_config_rule`
- Ansible Galaxy
  - http://galaxy.ansible.com/

Module Naming – Implicit vs. Explicit

```
$ ansible localhost -m command -a "uptime"
localhost | CHANGED | rc=0 >>
15:56  up  7:02, 2 users, load averages: 1.43 1.67 1.80


$ ansible localhost -m ansible.builtin.command -a "uptime"
localhost | CHANGED | rc=0 >>
15:56  up  7:02, 2 users, load averages: 1.57 1.68 1.80
```

```
$ ansible -i hosts -m ping testhost

testhost | SUCCESS => {

    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python"
    },

    "changed": false,

    "ping": "pong"
}
```

www.geant.org

```
$ ansible -i hosts -m ping veos
veos | UNREACHABLE! => {
    "changed": false,
    "msg": Failed to create temporary directory...
…
…
…

    "unreachable": true
}
```

```
$ ansible -i hosts -m raw -a "show version" veos

veos | CHANGED | rc=0 >>

vEOS
Hardware version:
Serial number:
Hardware MAC address: 0800.270e.c3d7
System MAC address: 0800.270e.c3d7

Software image version: 4.25.2F
…
```

GÉANT

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