Security Improvement with Infrastructure as Code

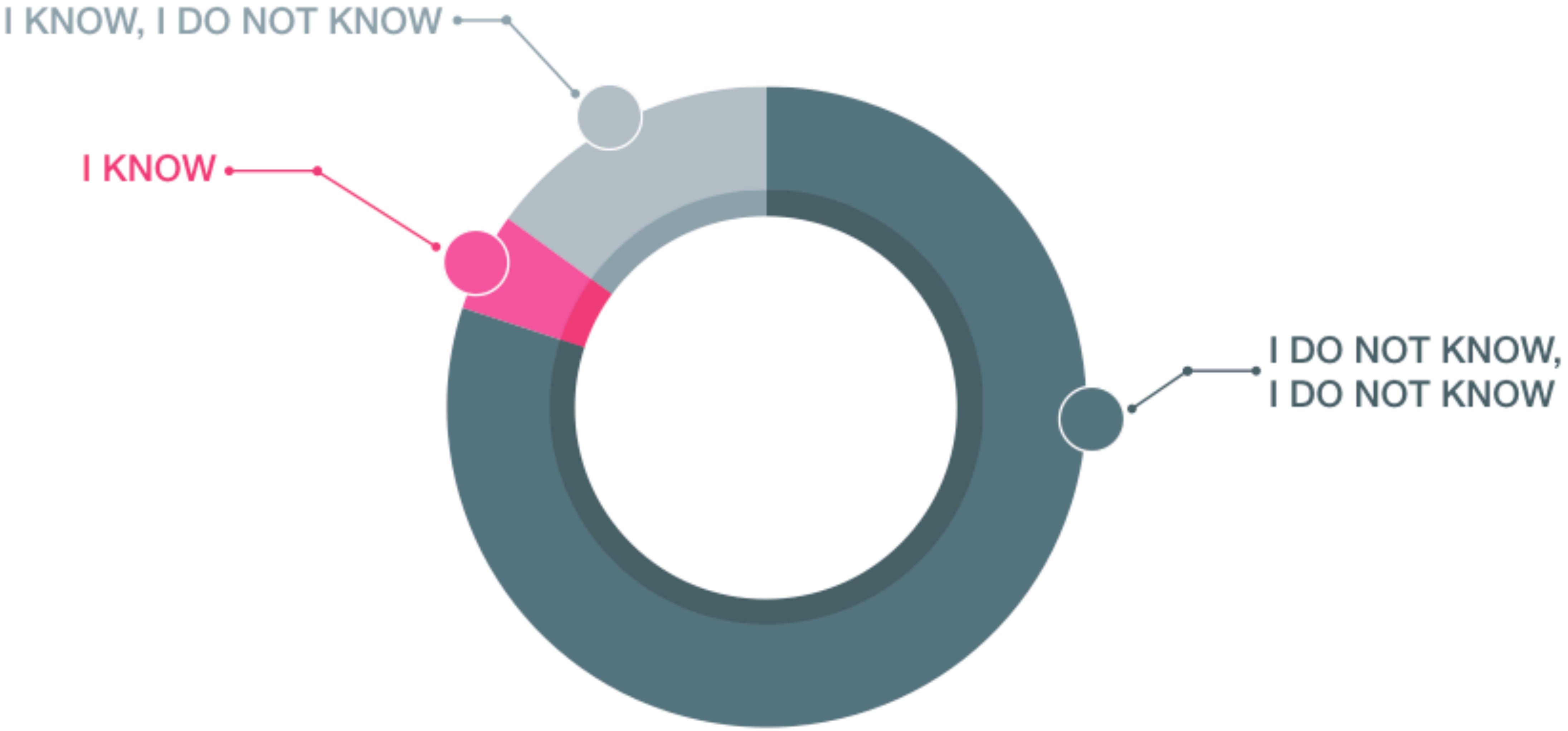
DAMRONGSAK REETANON

MFEC Public Company Limited

MiSSConf(SPS)

DAMRONGSAK REETANON

- Chief Cyber Security Office [at] MFEC
- Open Source Lover
- OpenStack Users and Contributors Thailand Community Co-Founder
- Amateur Photographer
- Red Hat Certified Architect Level III


I KNOW, I DO NOT KNOW
I KNOW
I DO NOT KNOW, I DO NOT KNOW

Human Knowledge Belongs To The Worlds

- Antitrust

Image by ElisaRiva from Pixabay

Security Improvement with Infrastructure as Code

Infrastructure as code (IaC)

the process of **managing and provisioning computer data centers** through **machine-readable definition files**, rather than physical hardware configuration or interactive configuration tools. The IT infrastructure managed by this comprises ***both physical equipment such as bare-metal servers as well as virtual machines and associated configuration resources***. The definitions may be in a version control system. It can use either scripts or declarative definitions, rather than manual processes, but the term is more often used to promote declarative approaches.

WIKIPEDIA
The Free Encyclopedia

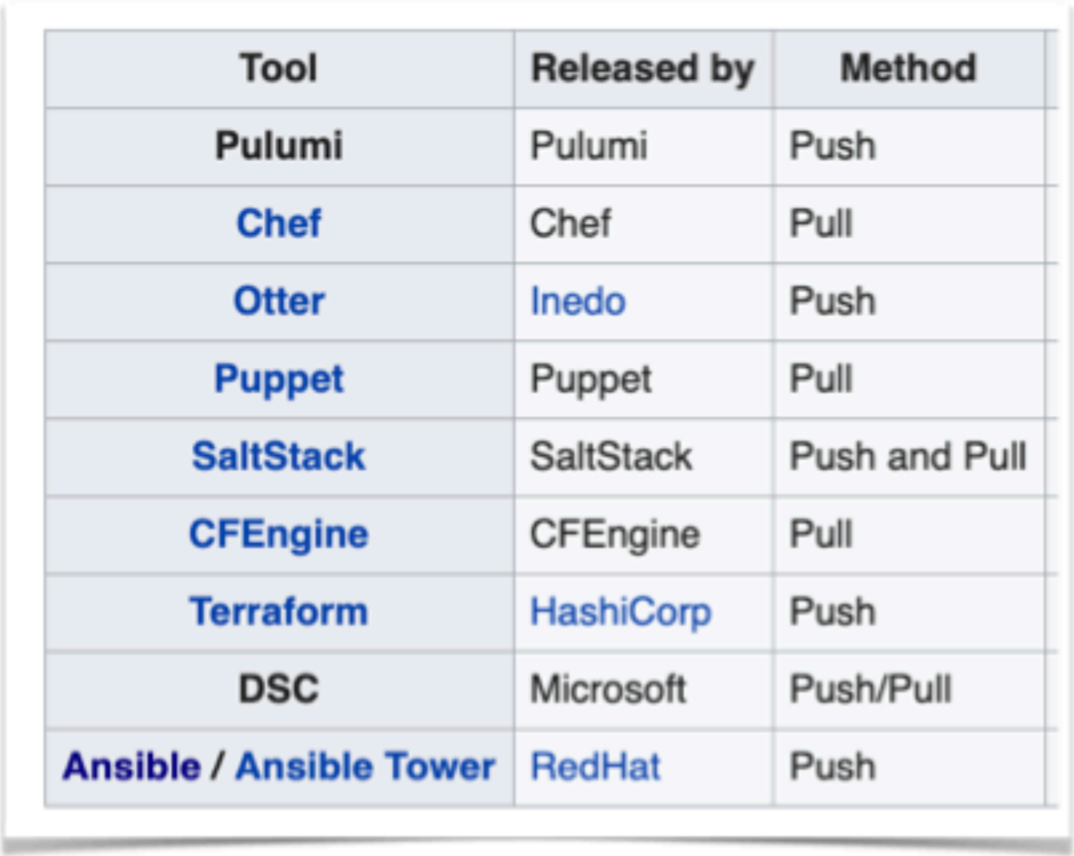
Treat your **Infrastructure** as **Code**

ref: https://www.dtelepathy.com/blog/wp-content/uploads/2015/02/Betterment_Say-No1.jpg puppet labs

TERRAFORM

CHEF™ CODE CAN

ANSIBLE

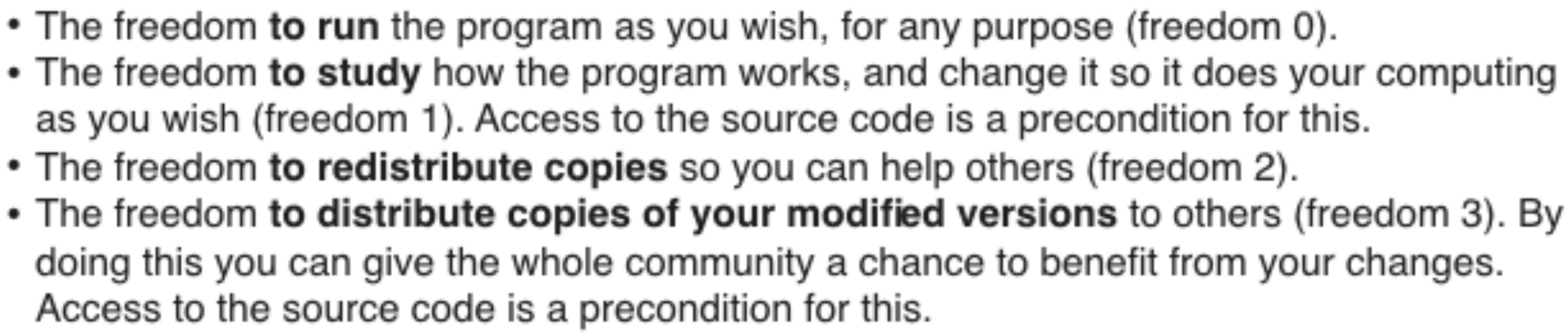


Tool	Released by	Method
Pulumi	Pulumi	Push
Chef	Chef	Pull
Otter	Inedo	Push
Puppet	Puppet	Pull
SaltStack	SaltStack	Push and Pull
CFEngine	CFEngine	Pull
Terraform	HashiCorp	Push
DSC	Microsoft	Push/Pull
Ansible / Ansible Tower	RedHat	Push

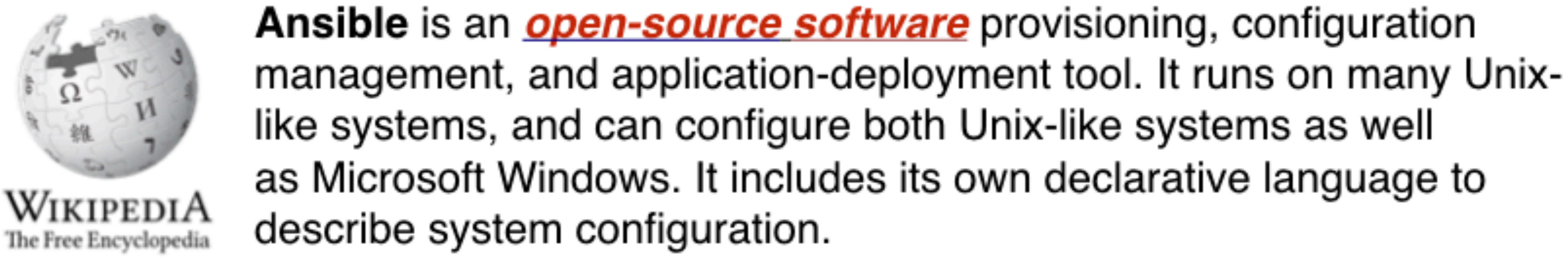
- Patching
- Hardening
- ~~Compliance~~
- Authentication
- Authorization
- Accounting
- ~~Security Orchestrator Automation Response~~

Security Improvement with Infrastructure as Code

ANSIBLE



- The freedom **to run** the program as you wish, for any purpose (freedom 0).
- The freedom **to study** how the program works, and change it so it does your computing as you wish (freedom 1). Access to the source code is a precondition for this.
- The freedom **to redistribute copies** so you can help others (freedom 2).
- The freedom **to distribute copies of your modified versions** to others (freedom 3). By doing this you can give the whole community a chance to benefit from your changes. Access to the source code is a precondition for this.



Ansible is an ***open-source software*** provisioning, configuration management, and application-deployment tool. It runs on many Unix-like systems, and can configure both Unix-like systems as well as Microsoft Windows. It includes its own declarative language to describe system configuration.

ANSIBLE

configuration management

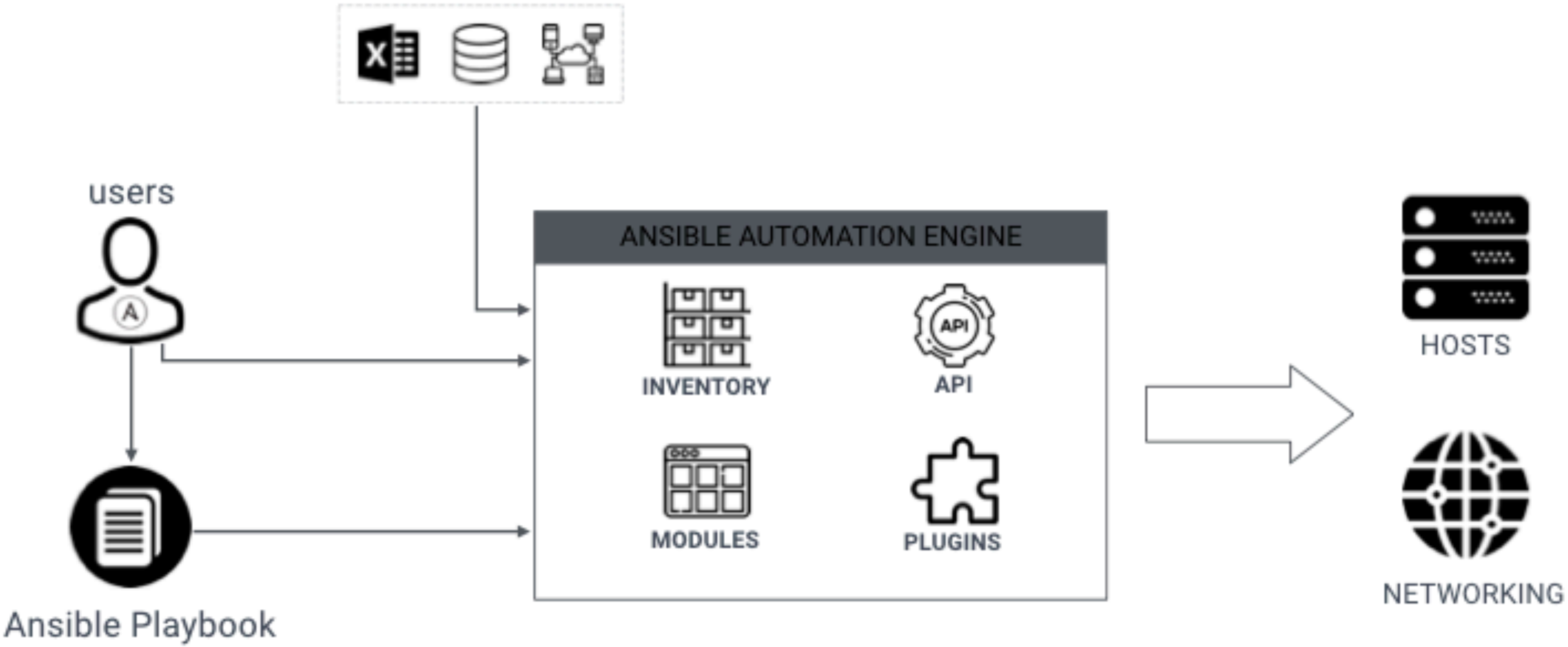
CI / CD

orchestration

application deployment

infrastructure provisioning

ANSIBLE ARCHITECTURE



users

ANSIBLE AUTOMATION ENGINE

INVENTORY

API

MODULES

PLUGINS

HOSTS

NETWORKING

Ansible Playbook

AWX

AWX

The AWX Project -- AWX for short -- **is an open source community project**, sponsored by Red Hat, that enables users to better control their Ansible project use in IT environments. AWX is the upstream project from which the Red Hat Ansible Tower offering is ultimately derived.

AWX is a **web-based solution that makes Ansible even more easy to use for IT teams** of all kinds. It’s designed to be the hub for all of your automation tasks.

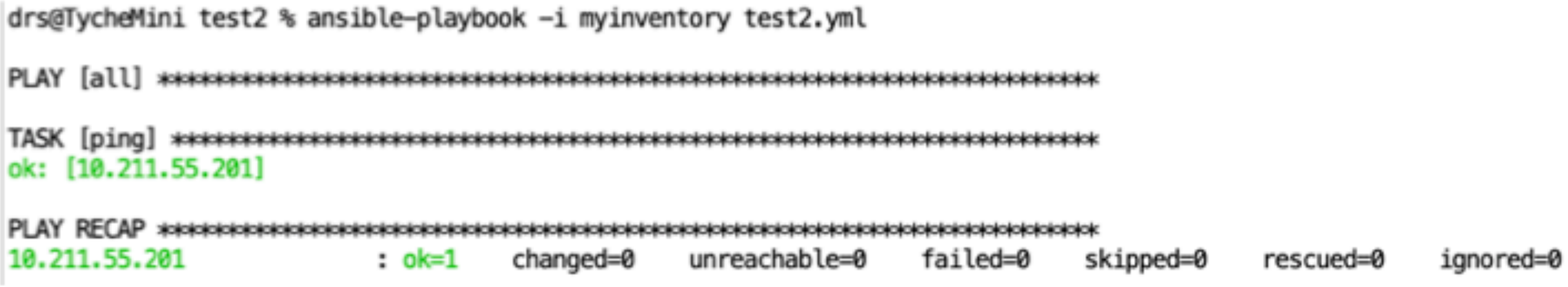
AWX allows you to control access to who can access what, even allowing sharing of SSH credentials without someone being able to transfer those credentials. Inventory can be graphically managed or synced with a wide variety of cloud sources. It logs all of your jobs, integrates well with LDAP, and has an amazing browsable REST API. Command line tools are available for easy integration with Jenkins as well. Provisioning callbacks provide great support for autoscaling topologies.

How to manage credentials in Ansible?

Image by Anemone123 from Pixabay

```
- hosts: all
  gather_facts: off
  remote_user: root
  vars:
    ansible_password: centos
  tasks:
    - ping:
```

```
drs@TycheMini test2 % cat test2.yml
- hosts: all
  gather_facts: off
  remote_user: root
  tasks:
    - ping:
```



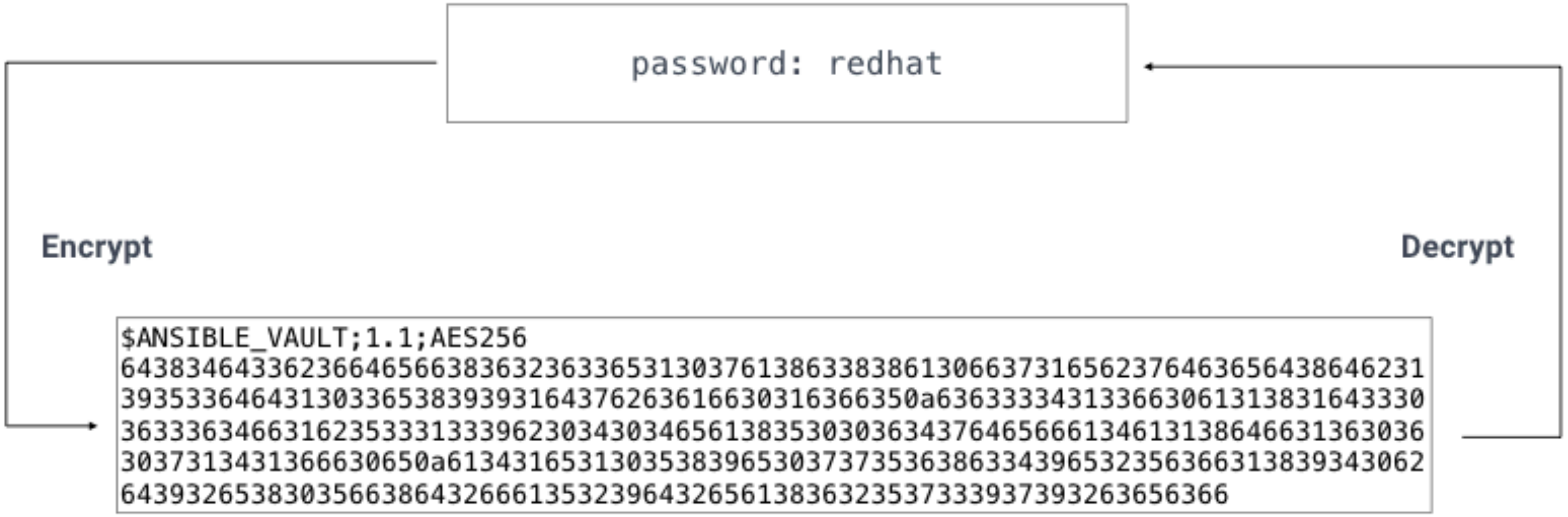
```
drs@TycheMini test2 % ansible-playbook -i myinventory test2.yml

PLAY [all] *********************************************************************

TASK [ping] ********************************************************************
ok: [10.211.55.201]

PLAY RECAP *********************************************************************
10.211.55.201              : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

ANSIBLE VAULT

Ansible Vault is a feature of **ansible that allows you to keep sensitive data such as passwords or keys in encrypted files**, rather than as plaintext in playbooks or roles. These vault files can then be distributed or placed in source control.

To enable this feature, a command line tool **ansible-vault** is used to edit files, and a command line flag (--ask-vault-pass or --vault-password-file) is used. Alternately, you may specify the location of a password file or command Ansible to always prompt for the password in your ansible.cfg file. These options require no command line flag usage.

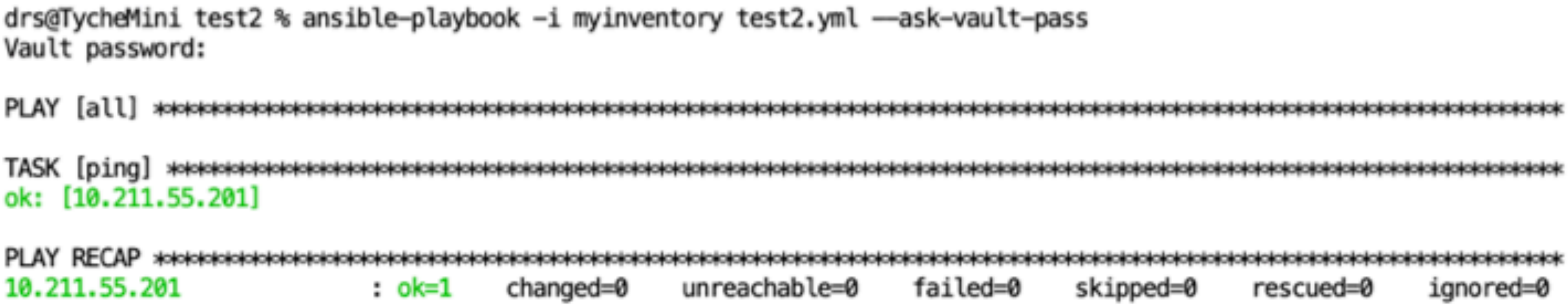
ANSIBLE VAULT



```
password: redhat
```

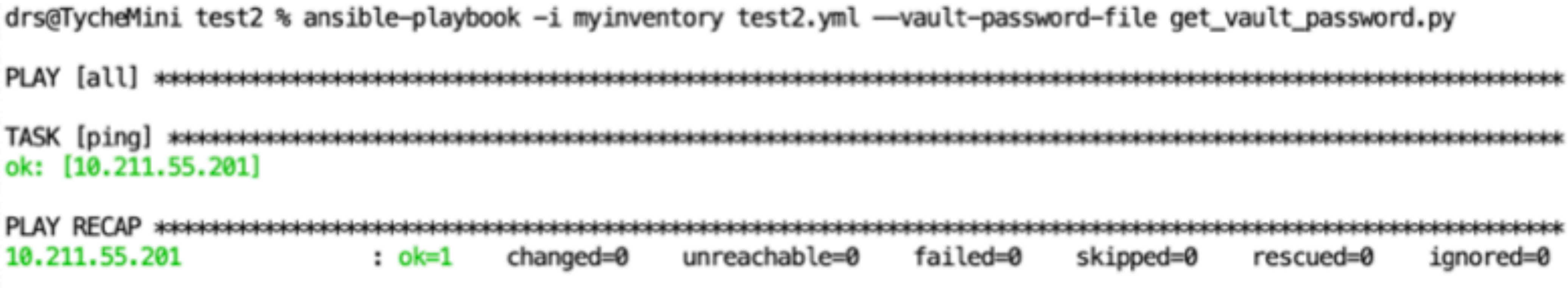
```
$ANSIBLE_VAULT;1.1;AES256
64383464336236646566383632363365313037613863383861306637316562376463656438646231
3935336464313033653839393164376263616630316366350a636333343133663061313831643330
36333634663162353331333962303430346561383530303634376465666134613138646631363036
3037313431366630650a613431653130353839653037373536386334396532356366313839343062
64393265383035663864326661353239643265613836323537333937393263656366
```

```
drs@TycheMini test2 % cat test2.yml
- hosts: all
  gather_facts: off
  remote_user: root
  tasks:
    - ping:
drs@TycheMini test2 % cat group_vars/all
$ANSIBLE_VAULT;1.1;AES256
61326132363836306434323965663762646433306637373563363136643039396337623935636139
64656137343035633466643962366664636362613561333000a616462663232643864666136313034
35653733373233366163366332376538386463653133623339376639343266643136383835353537
3737356263346561360a303061336261323364313864363862383166316465656339386334333834
353461366235613934346462313739346337363438383065353235643136393935 32
```



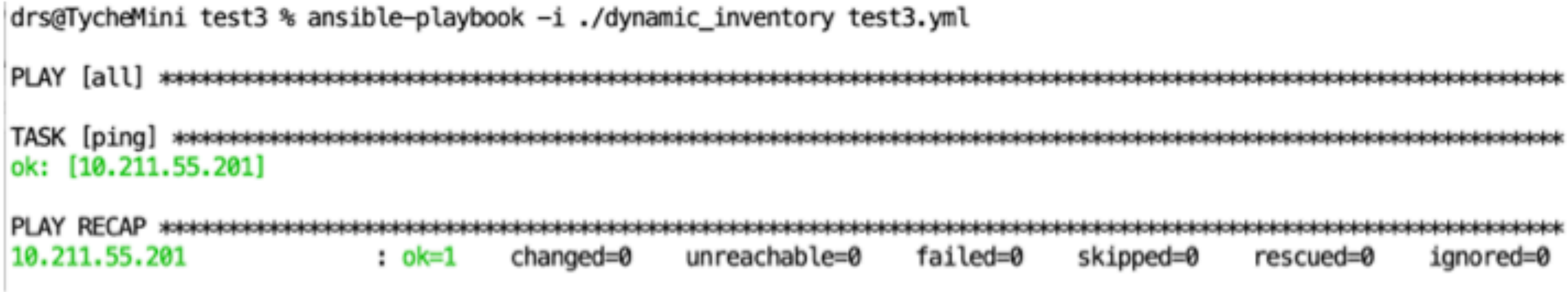
```
drs@TycheMini test2 % ansible-playbook -i myinventory test2.yml --ask-vault-pass
Vault password:

PLAY [all] *********************************************************************************************

TASK [ping] ********************************************************************************************
ok: [10.211.55.201]

PLAY RECAP *********************************************************************************************
10.211.55.201              : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
drs@TycheMini test2 % cat get_vault_password.py
#!/usr/bin/python

print("centos")
```



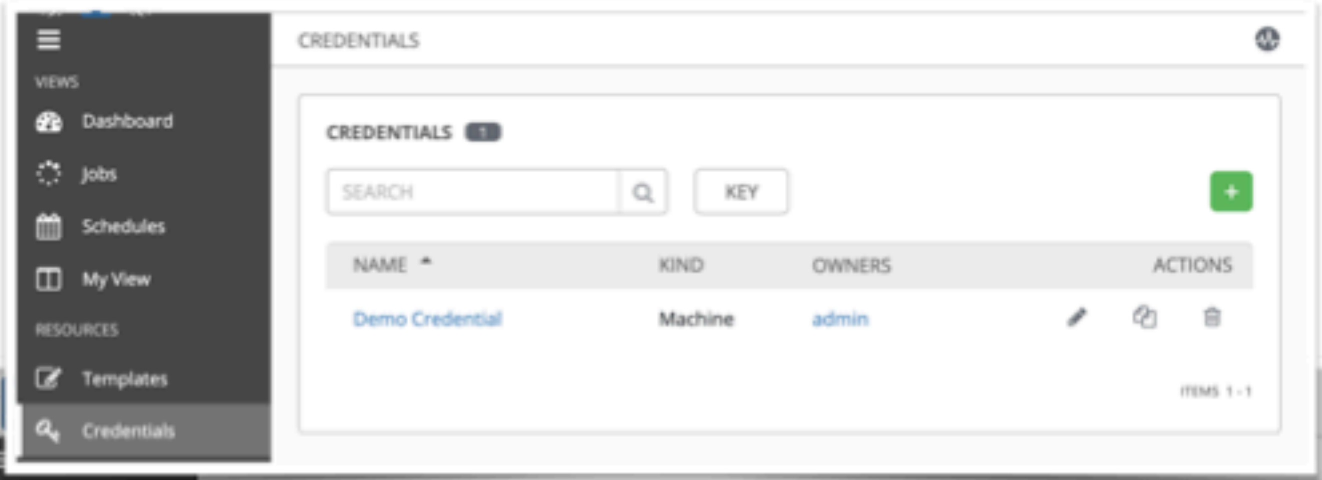
```
drs@TycheMini test2 % ansible-playbook -i myinventory test2.yml --vault-password-file get_vault_password.py

PLAY [all] ********************************************************************************************************

TASK [ping] *******************************************************************************************************
ok: [10.211.55.201]

PLAY RECAP ********************************************************************************************************
10.211.55.201              : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
drs@TycheMini test3 % ./dynamic_inventory --list
 {
    group: {
        hosts: [
            10.211.55.201
        ],
        vars: {
            ansible_password: centos,
        }
    }
}
```



```
drs@TycheMini test3 % ansible-playbook -i ./dynamic_inventory test3.yml

PLAY [all] ********************************************************************************************************

TASK [ping] *******************************************************************************************************
ok: [10.211.55.201]

PLAY RECAP ********************************************************************************************************
10.211.55.201              : ok=1    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

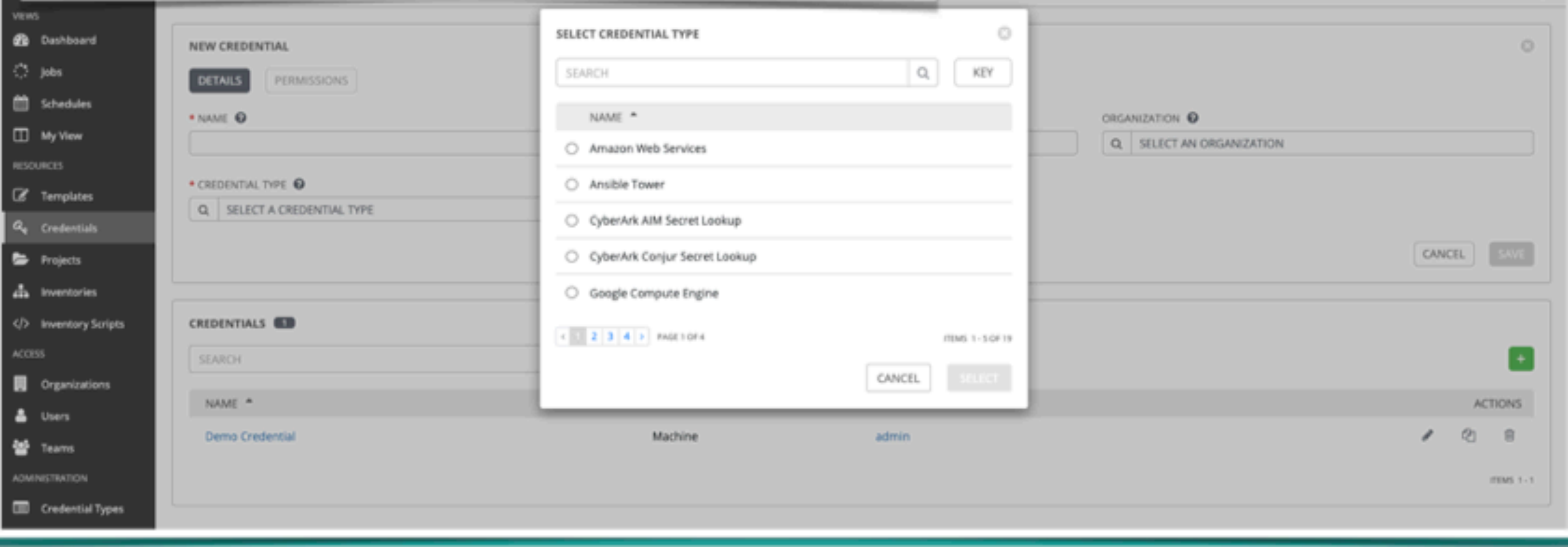
AWX :: Credentials



CREDENTIALS 1

NAME	KIND	OWNERS	ACTIONS
Demo Credential	Machine	admin	

ITEMS 1 - 1



NEW CREDENTIAL

* NAME

* CREDENTIAL TYPE: SELECT A CREDENTIAL TYPE

ORGANIZATION: SELECT AN ORGANIZATION

SELECT CREDENTIAL TYPE

NAME

- ○ Amazon Web Services
- ○ Ansible Tower
- ○ CyberArk AIM Secret Lookup
- ○ CyberArk Conjur Secret Lookup
- ○ Google Compute Engine

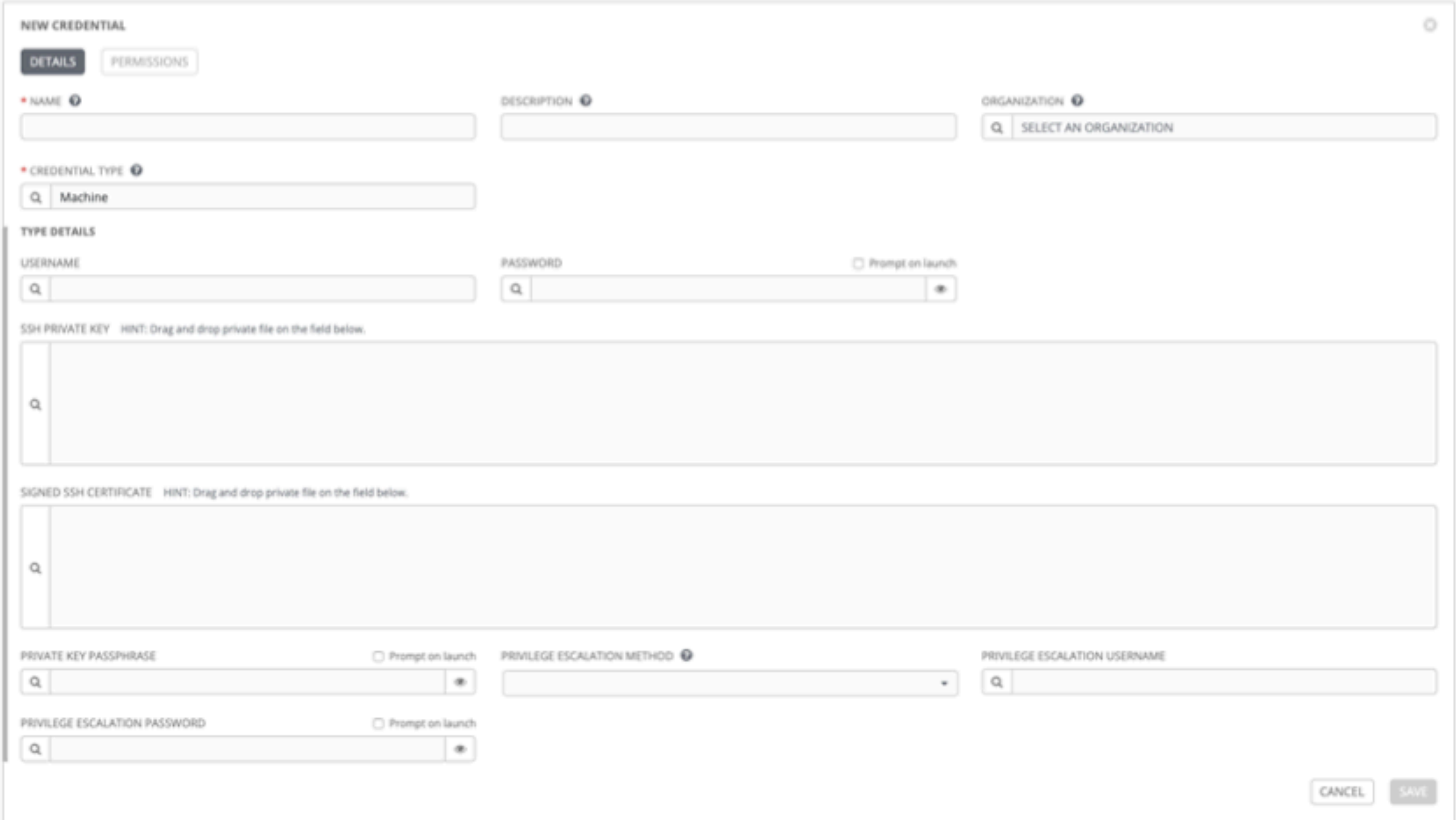
1 2 3 4 PAGE 1 OF 4

ITEMS 1 - 5 OF 19

CREDENTIALS 1

NAME			ACTIONS
Demo Credential	Machine	admin	

ITEMS 1 - 1



NEW CREDENTIAL

* CREDENTIAL TYPE: Machine

TYPE DETAILS

SSH PRIVATE KEY HINT: Drag and drop private file on the field below.

SIGNED SSH CERTIFICATE HINT: Drag and drop private file on the field below.

Hardening and Patch

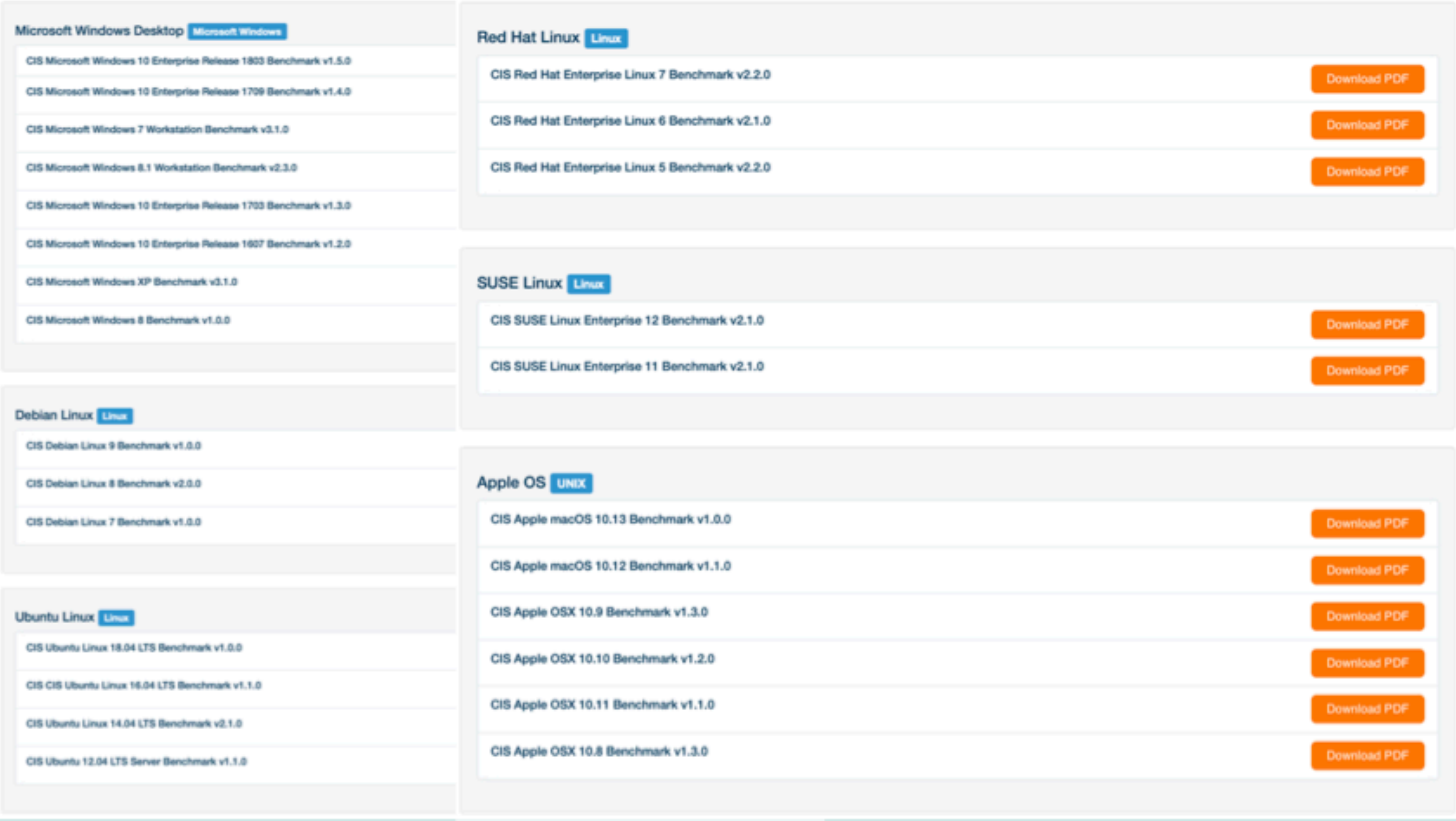
hardening is usually **the process of securing a system by reducing its surface of vulnerability,** which is larger when a system performs more functions; in principle a single-function system is more secure than a multipurpose one. Reducing available ways of attack typically includes changing default passwords, the removal of unnecessary software, unnecessary usernames or logins, and the disabling or removal of unnecessary services.

A **patch** is a set of changes to a computer program or its supporting data designed to update, fix, or improve it. This includes fixing security vulnerabilities and other bugs, with such patches usually being called **bugfixes** or **bug fixes**, and improving the usability or performance. Although meant to fix problems, poorly designed patches can sometimes introduce new problems

CIS® Center for Internet Security®

What are CIS Benchmarks?

CIS Benchmarks are best practices for the secure configuration of a target system. Available for more than 140 technologies, CIS Benchmarks are developed through a unique consensus-based process comprised of cybersecurity professionals and subject matter experts around the world. CIS Benchmarks are the only consensus-based, best-practice security configuration guides both developed and accepted by government, business, industry, and academia.



Microsoft Windows Desktop Microsoft Windows

- CIS Microsoft Windows 10 Enterprise Release 1803 Benchmark v1.5.0
- CIS Microsoft Windows 10 Enterprise Release 1709 Benchmark v1.4.0
- CIS Microsoft Windows 7 Workstation Benchmark v3.1.0
- CIS Microsoft Windows 8.1 Workstation Benchmark v2.3.0
- CIS Microsoft Windows 10 Enterprise Release 1703 Benchmark v1.3.0
- CIS Microsoft Windows 10 Enterprise Release 1607 Benchmark v1.2.0
- CIS Microsoft Windows XP Benchmark v3.1.0
- CIS Microsoft Windows 8 Benchmark v1.0.0

Debian Linux Linux

- CIS Debian Linux 9 Benchmark v1.0.0
- CIS Debian Linux 8 Benchmark v2.0.0
- CIS Debian Linux 7 Benchmark v1.0.0

Ubuntu Linux Linux

- CIS Ubuntu Linux 18.04 LTS Benchmark v1.0.0
- CIS CIS Ubuntu Linux 16.04 LTS Benchmark v1.1.0
- CIS Ubuntu Linux 14.04 LTS Benchmark v2.1.0
- CIS Ubuntu 12.04 LTS Server Benchmark v1.1.0

Red Hat Linux Linux

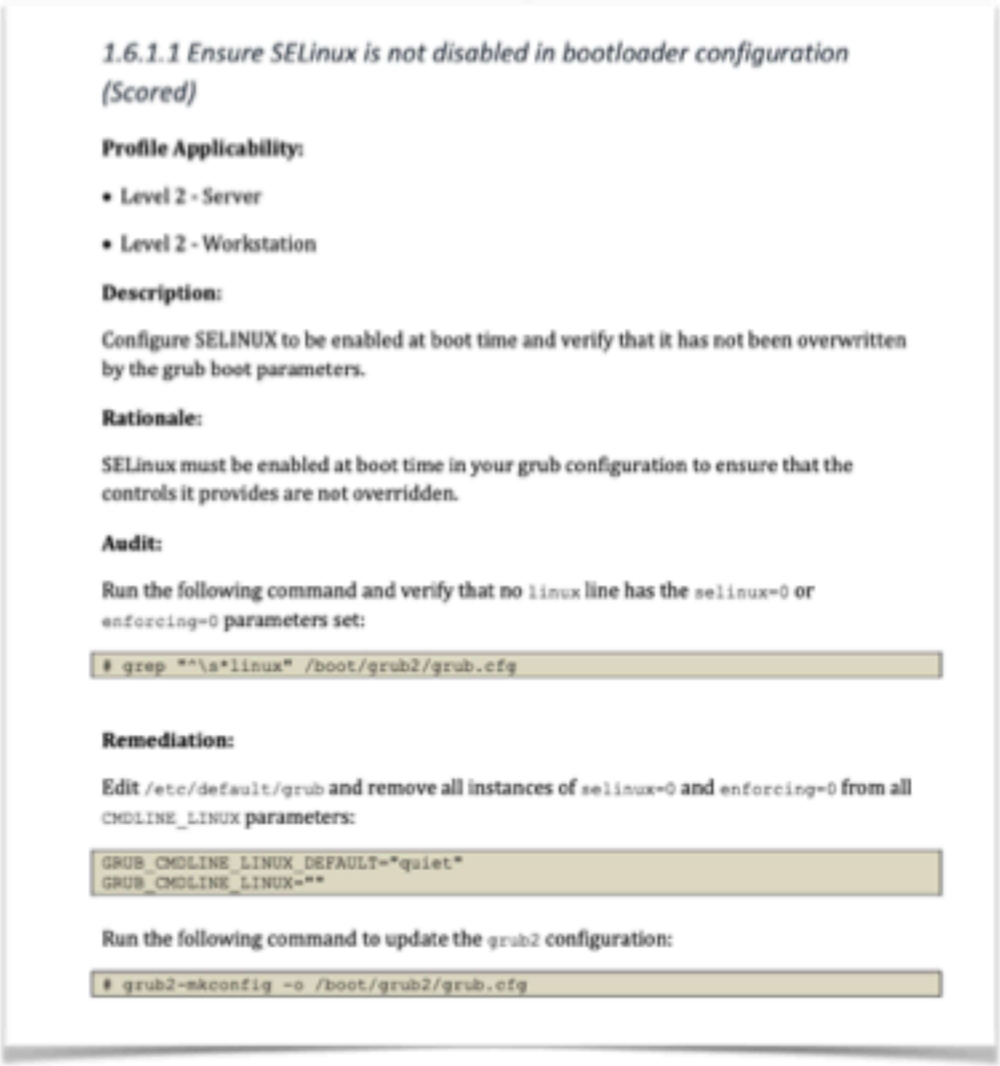
- CIS Red Hat Enterprise Linux 7 Benchmark v2.2.0
- CIS Red Hat Enterprise Linux 6 Benchmark v2.1.0
- CIS Red Hat Enterprise Linux 5 Benchmark v2.2.0

SUSE Linux Linux

- CIS SUSE Linux Enterprise 12 Benchmark v2.1.0
- CIS SUSE Linux Enterprise 11 Benchmark v2.1.0

Apple OS UNIX

- CIS Apple macOS 10.13 Benchmark v1.0.0
- CIS Apple macOS 10.12 Benchmark v1.1.0
- CIS Apple OSX 10.9 Benchmark v1.3.0
- CIS Apple OSX 10.10 Benchmark v1.2.0
- CIS Apple OSX 10.11 Benchmark v1.1.0
- CIS Apple OSX 10.8 Benchmark v1.3.0



1.6.1.1 Ensure SELinux is not disabled in bootloader configuration (Scored)

Profile Applicability:

- Level 2 - Server
- Level 2 - Workstation

Description:

Configure SELINUX to be enabled at boot time and verify that it has not been overwritten by the grub boot parameters.

Rationale:

SELinux must be enabled at boot time in your grub configuration to ensure that the controls it provides are not overridden.

Audit:

Run the following command and verify that no `linux` line has the `selinux=0` or `enforcing=0` parameters set:

```
# grep "^\s*linux" /boot/grub2/grub.cfg
```

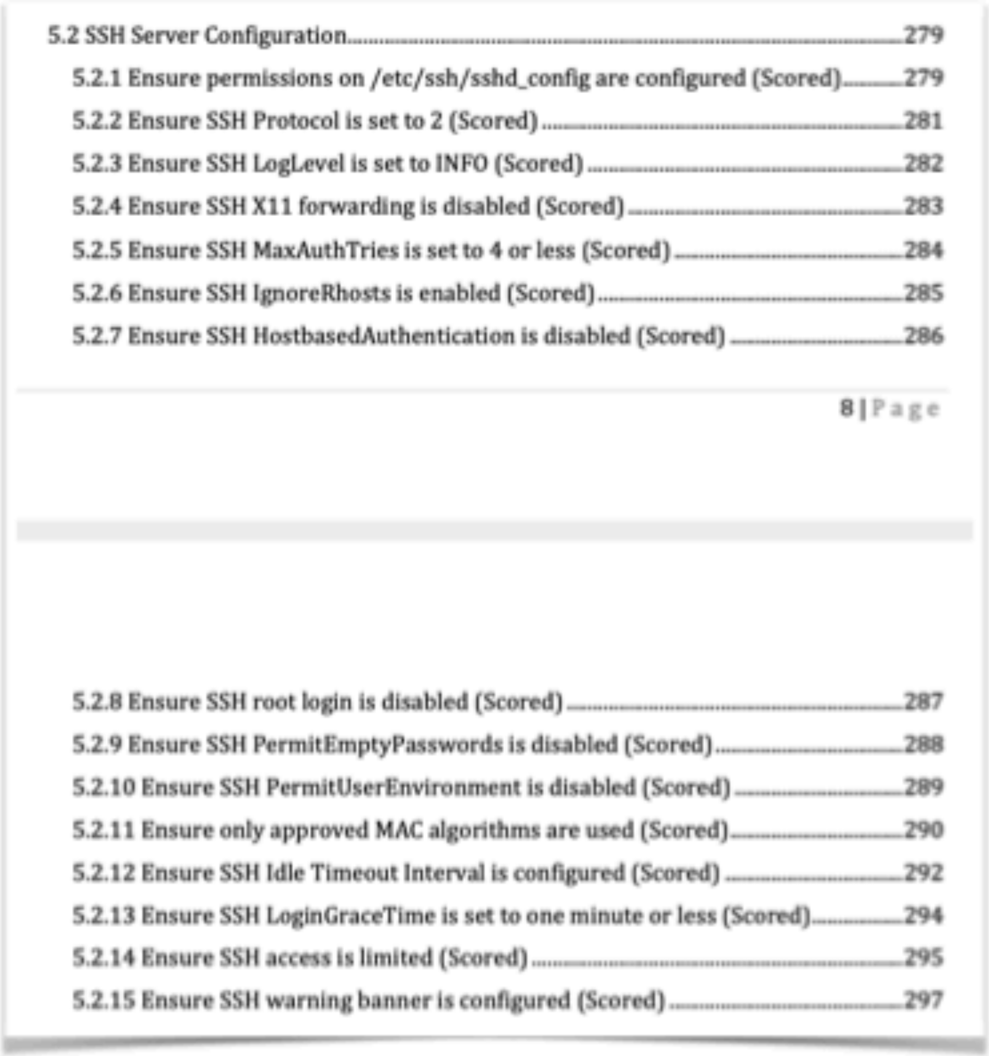
Remediation:

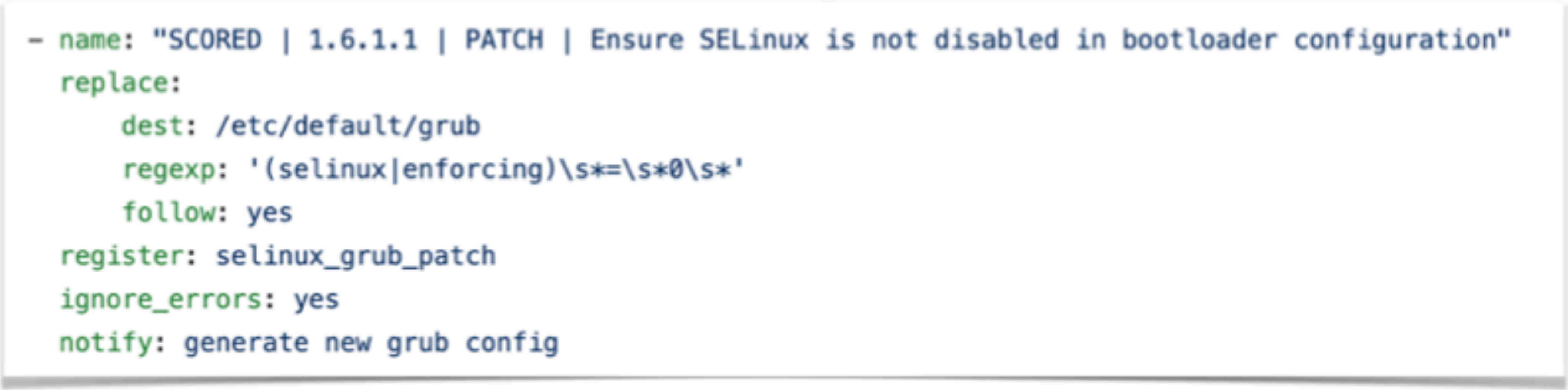
Edit `/etc/default/grub` and remove all instances of `selinux=0` and `enforcing=0` from all `CMDLINE_LINUX` parameters:

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet"
GRUB_CMDLINE_LINUX=""
```

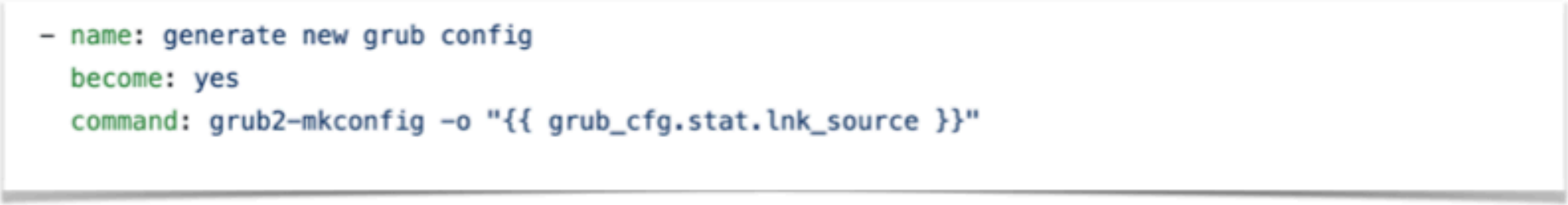
Run the following command to update the `grub2` configuration:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

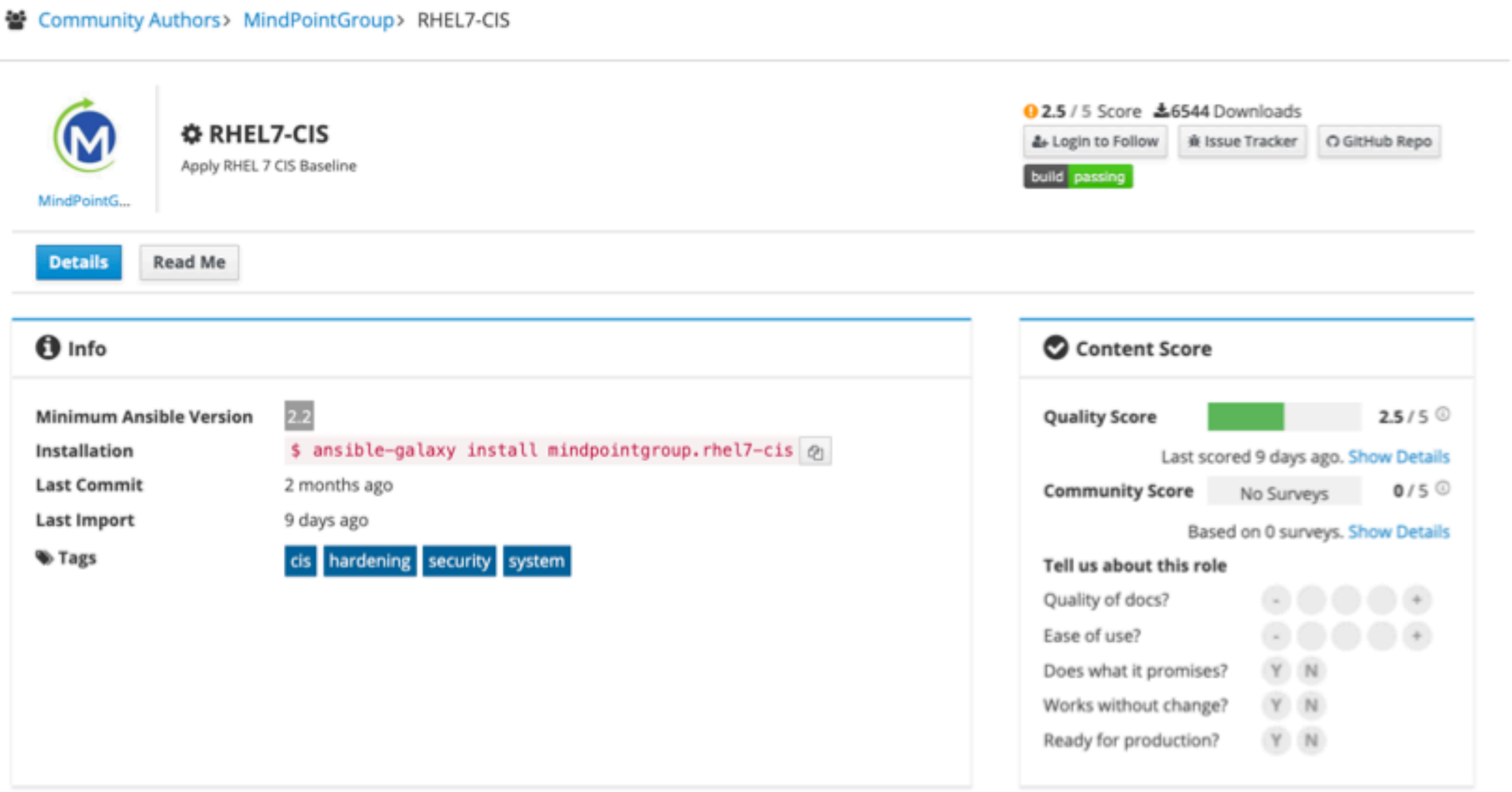




```
- name: "SCORED | 1.6.1.1 | PATCH | Ensure SELinux is not disabled in bootloader configuration"
  replace:
      dest: /etc/default/grub
      regexp: '(selinux|enforcing)\s*=\s*0\s*'
      follow: yes
  register: selinux_grub_patch
  ignore_errors: yes
  notify: generate new grub config
```



```
- name: generate new grub config
  become: yes
  command: grub2-mkconfig -o "{{ grub_cfg.stat.lnk_source }}"
```



Community Authors > MindPointGroup > RHEL7-CIS

MindPointG...

RHEL7-CIS

Apply RHEL 7 CIS Baseline

2.5 / 5 Score 6544 Downloads

Info

Minimum Ansible Version	2.2
Installation	`$ ansible-galaxy install mindpointgroup.rhel7-cis`
Last Commit	2 months ago
Last Import	9 days ago
Tags	cis hardening security system

Content Score

Quality Score 2.5 / 5

Last scored 9 days ago. Show Details

Community Score No Surveys 0 / 5

Based on 0 surveys. Show Details

Tell us about this role

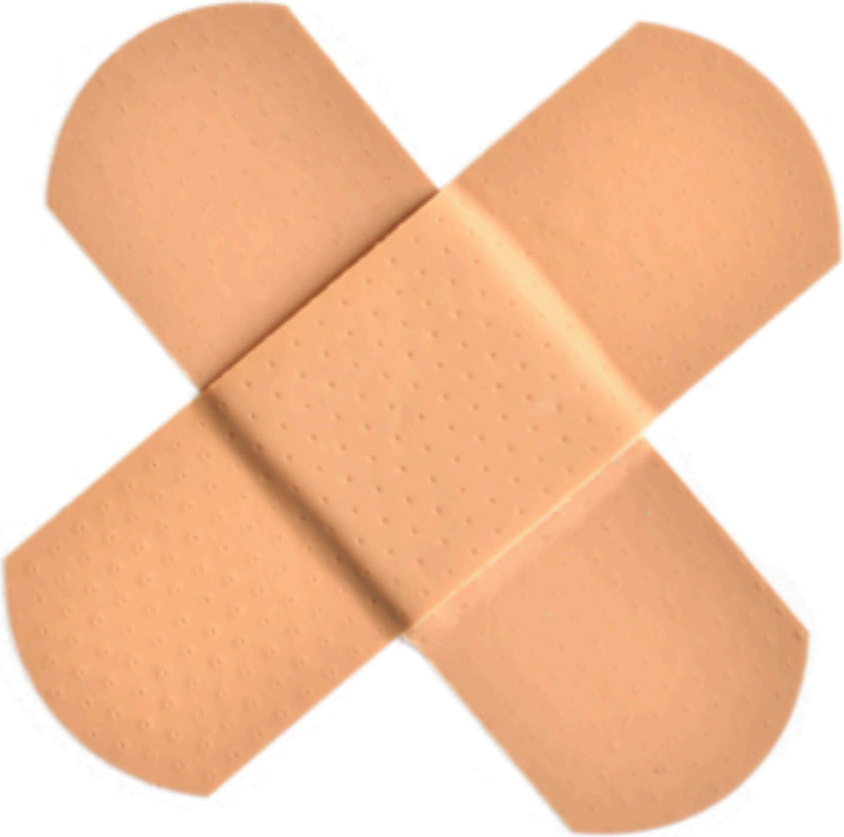
Quality of docs? - ○ ○ ○ +

Ease of use? - ○ ○ ○ +

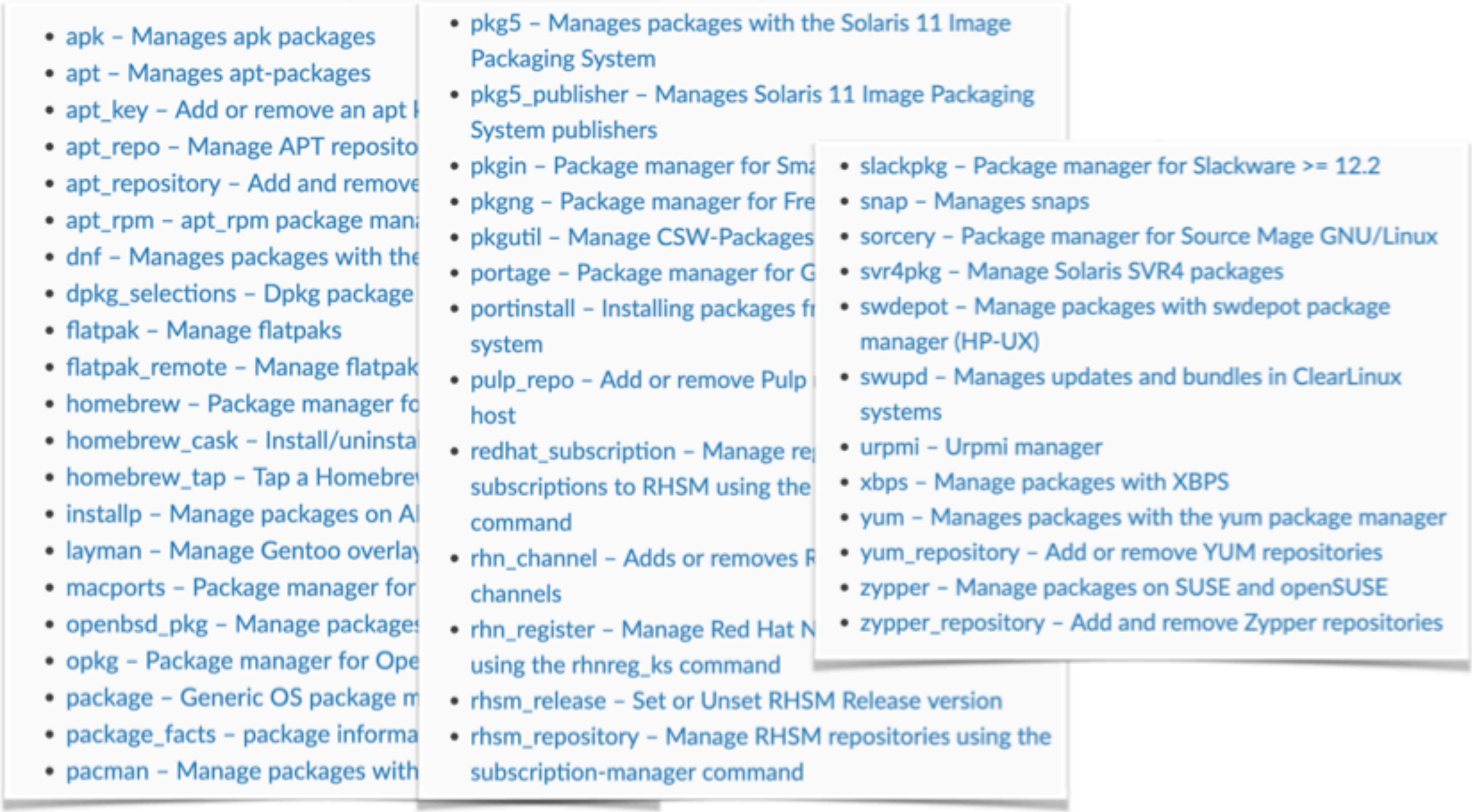
Does what it promises? Y N

Works without change? Y N

Ready for production? Y N



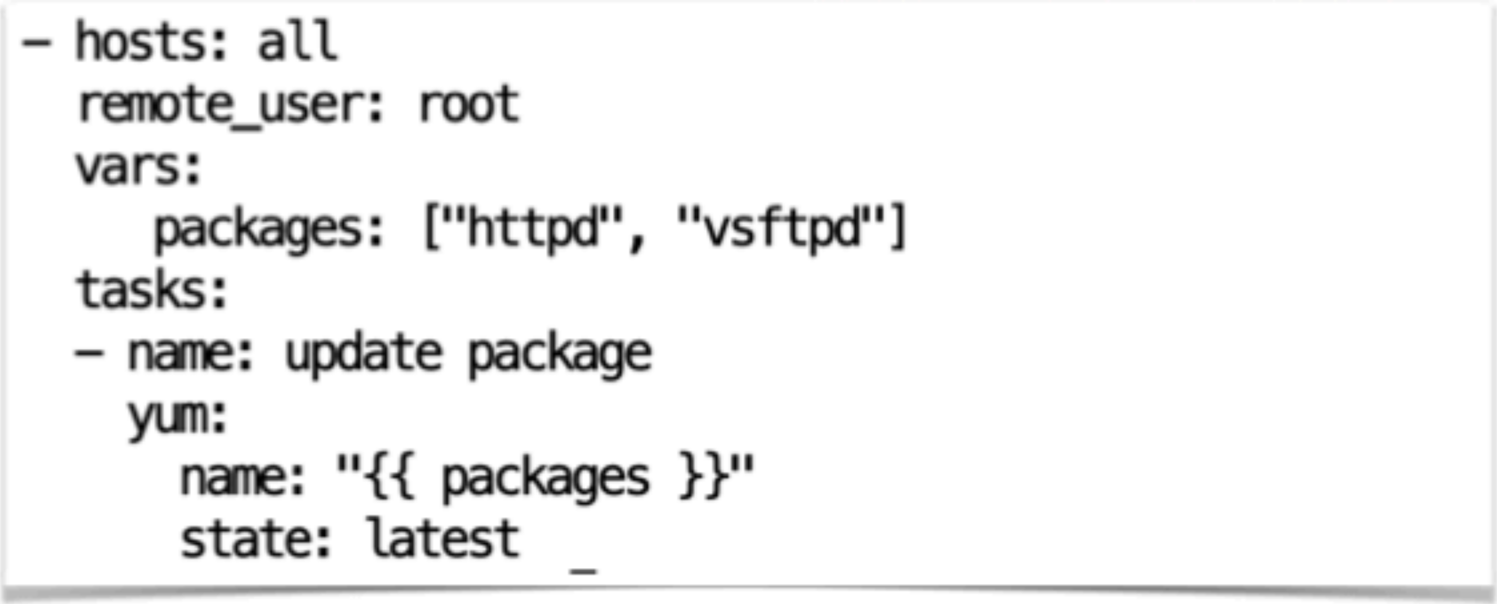
PATCH



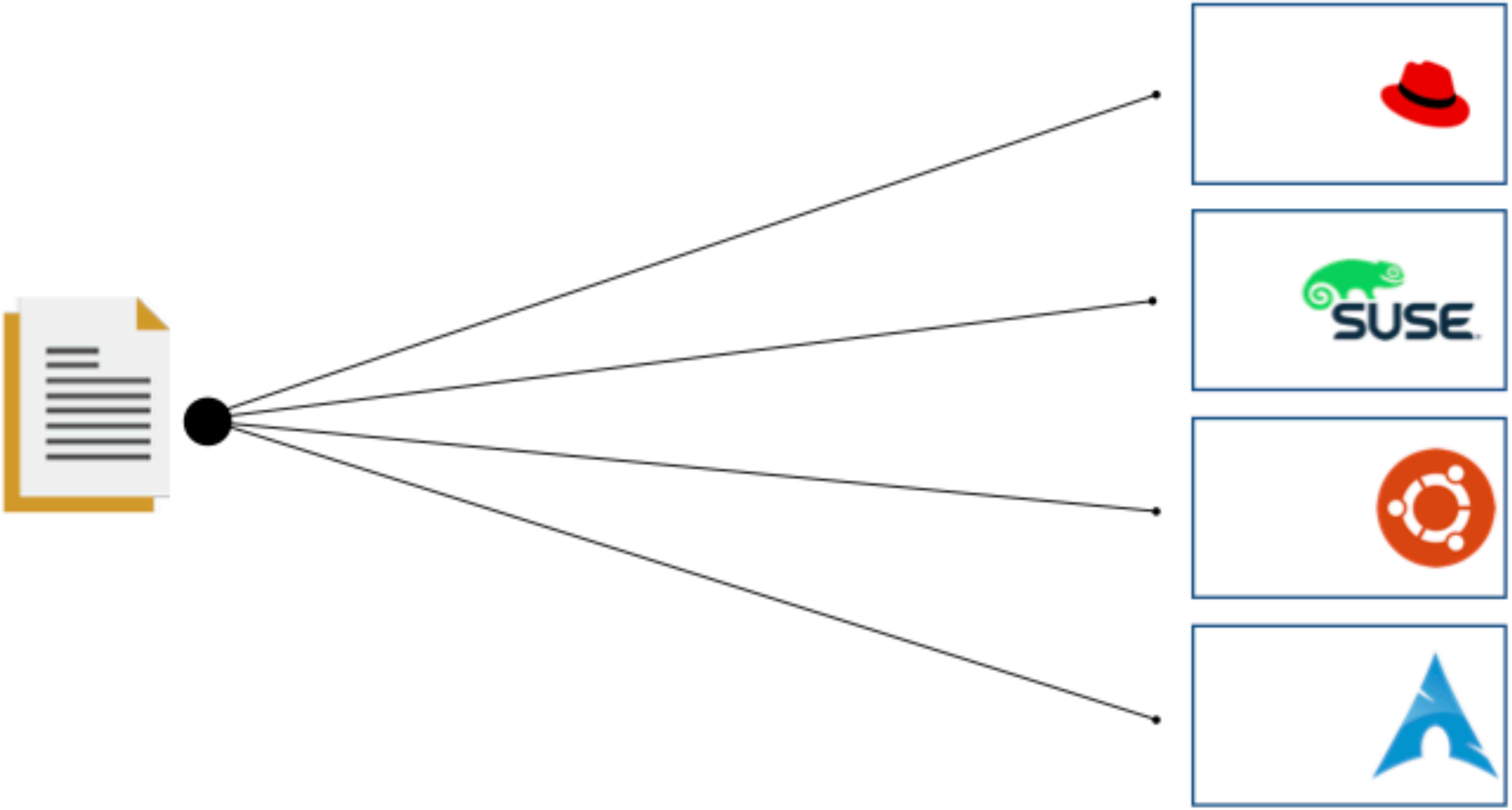
- apk – Manages apk packages
- apt – Manages apt-packages
- apt_key – Add or remove an apt k
- apt_repo – Manage APT reposito
- apt_repository – Add and remove
- apt_rpm – apt_rpm package man
- dnf – Manages packages with the
- dpkg_selections – Dpkg package
- flatpak – Manage flatpaks
- flatpak_remote – Manage flatpak
- homebrew – Package manager fo
- homebrew_cask – Install/uninsta
- homebrew_tap – Tap a Homebrew
- installp – Manage packages on AI
- layman – Manage Gentoo overlay
- macports – Package manager for
- openbsd_pkg – Manage packages
- opkg – Package manager for Ope
- package – Generic OS package m
- package_facts – package informa
- pacman – Manage packages with

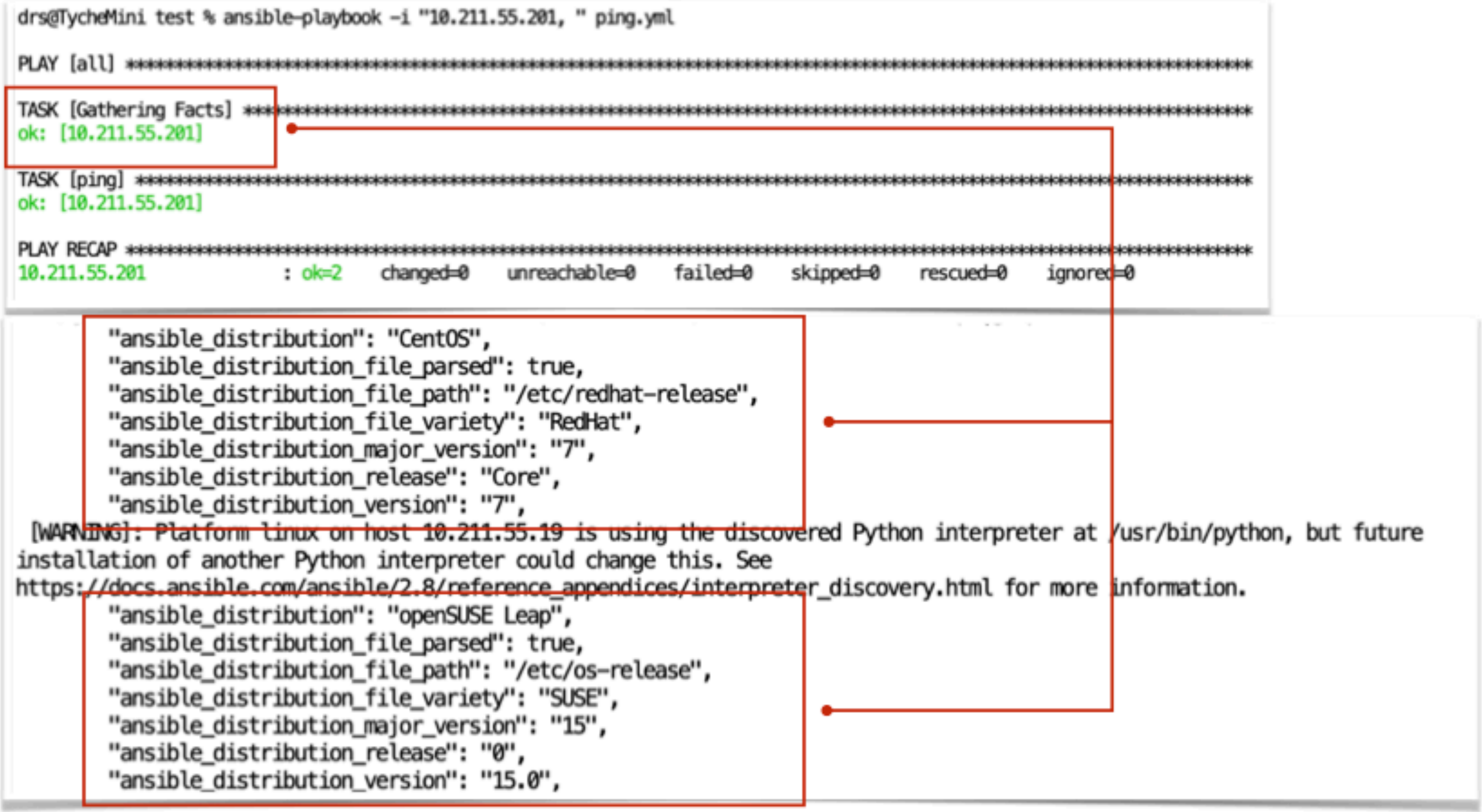
- pkg5 – Manages packages with the Solaris 11 Image Packaging System
- pkg5_publisher – Manages Solaris 11 Image Packaging System publishers
- pkgin – Package manager for Sma
- pkgng – Package manager for Fre
- pkgutil – Manage CSW-Packages
- portage – Package manager for G
- portinstall – Installing packages fr system
- pulp_repo – Add or remove Pulp host
- redhat_subscription – Manage re subscriptions to RHSM using the command
- rhn_channel – Adds or removes R channels
- rhn_register – Manage Red Hat N using the rhnreg_ks command
- rhsm_release – Set or Unset RHSM Release version
- rhsm_repository – Manage RHSM repositories using the subscription-manager command

- slackpkg – Package manager for Slackware >= 12.2
- snap – Manages snaps
- sorcery – Package manager for Source Mage GNU/Linux
- svr4pkg – Manage Solaris SVR4 packages
- swdepot – Manage packages with swdepot package manager (HP-UX)
- swupd – Manages updates and bundles in ClearLinux systems
- urpmi – Urpmi manager
- xbps – Manage packages with XBPS
- yum – Manages packages with the yum package manager
- yum_repository – Add or remove YUM repositories
- zypper – Manage packages on SUSE and openSUSE
- zypper_repository – Add and remove Zypper repositories



```
- hosts: all
  remote_user: root
  vars:
     packages: ["httpd", "vsftpd"]
  tasks:
  - name: update package
    yum:
      name: "{{ packages }}"
      state: latest  _
```

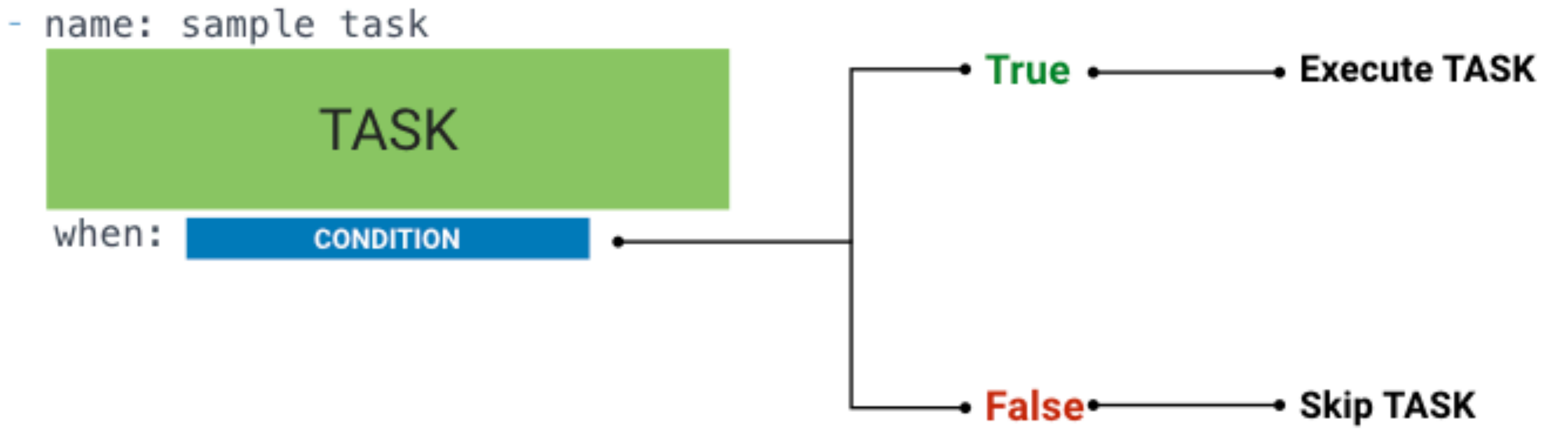

SUSE



```
drs@TycheMini test % ansible-playbook -i "10.211.55.201, " ping.yml

PLAY [all] ***************************************************************************************

TASK [Gathering Facts] ***************************************************************************
ok: [10.211.55.201]

TASK [ping] **************************************************************************************
ok: [10.211.55.201]

PLAY RECAP ***************************************************************************************
10.211.55.201              : ok=2    changed=0    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```

```
        "ansible_distribution": "CentOS",
        "ansible_distribution_file_parsed": true,
        "ansible_distribution_file_path": "/etc/redhat-release",
        "ansible_distribution_file_variety": "RedHat",
        "ansible_distribution_major_version": "7",
        "ansible_distribution_release": "Core",
        "ansible_distribution_version": "7",
 [WARNING]: Platform linux on host 10.211.55.19 is using the discovered Python interpreter at /usr/bin/python, but future
installation of another Python interpreter could change this. See
https://docs.ansible.com/ansible/2.8/reference_appendices/interpreter_discovery.html for more information.
        "ansible_distribution": "openSUSE Leap",
        "ansible_distribution_file_parsed": true,
        "ansible_distribution_file_path": "/etc/os-release",
        "ansible_distribution_file_variety": "SUSE",
        "ansible_distribution_major_version": "15",
        "ansible_distribution_release": "0",
        "ansible_distribution_version": "15.0",
```

When Statement



- name: sample task

TASK

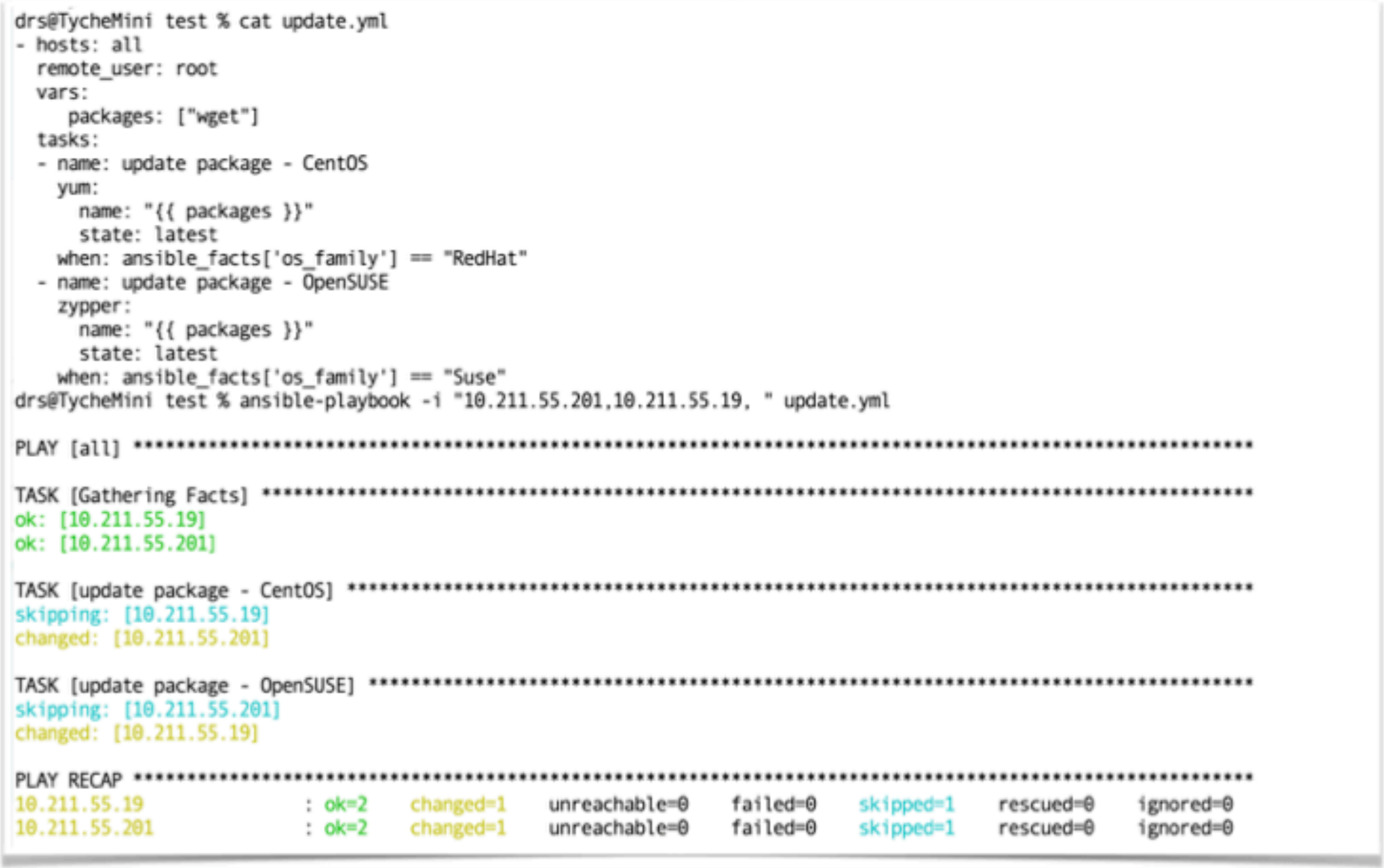
when: CONDITION

True → Execute TASK

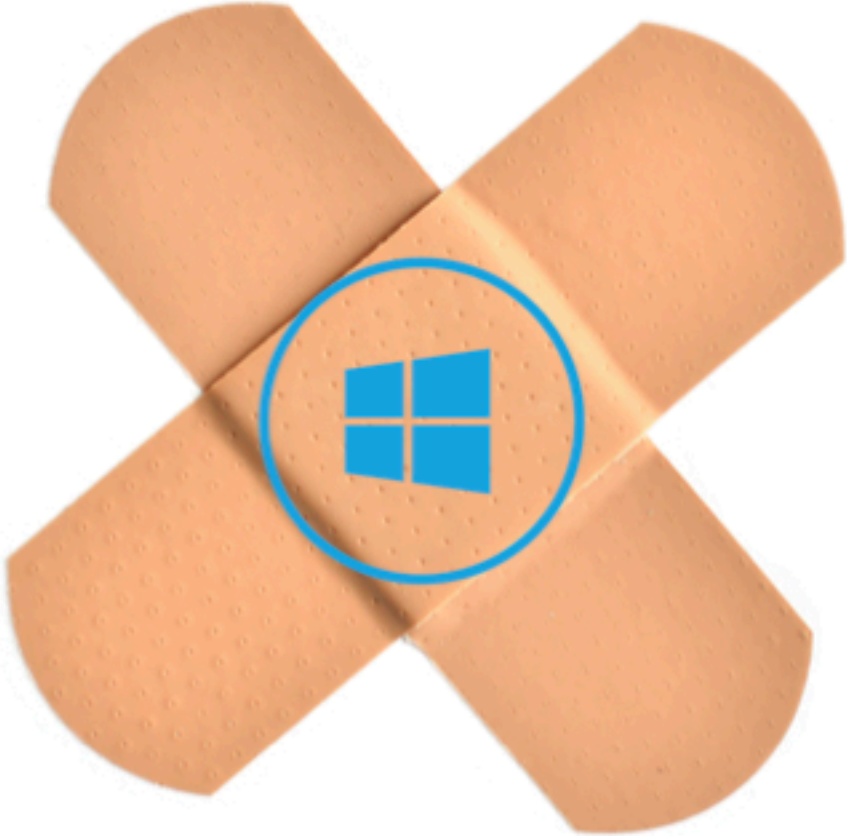
False → Skip TASK

```
tasks:
  - name: “Debian shutdown only”
    command: /sbin/shutdown -t now
    when: ansible_facts['os_family'] == "Debian"
```

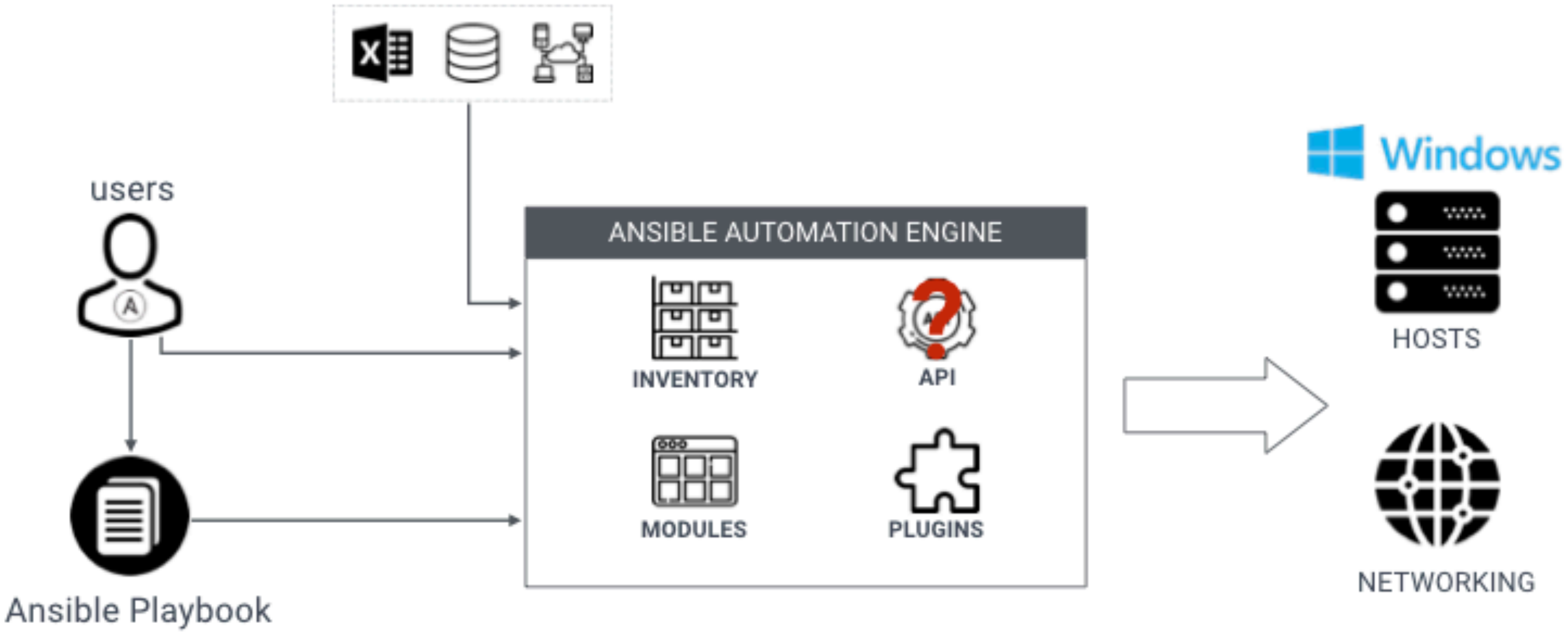
```
tasks:
  - name: "CentOS 7 and Debian 9 shutdown"
    command: /sbin/shutdown -t now
    when: (ansible_facts['distribution'] == "CentOS" and ansible_facts['distribution_major_version']
== "7") or (ansible_facts['distribution'] == "Debian" and
ansible_facts['distribution_major_version'] == “9")
```



```
drs@TycheMini test % cat update.yml
- hosts: all
  remote_user: root
  vars:
     packages: ["wget"]
  tasks:
  - name: update package - CentOS
    yum:
      name: "{{ packages }}"
      state: latest
    when: ansible_facts['os_family'] == "RedHat"
  - name: update package - OpenSUSE
    zypper:
      name: "{{ packages }}"
      state: latest
    when: ansible_facts['os_family'] == "Suse"
drs@TycheMini test % ansible-playbook -i "10.211.55.201,10.211.55.19, " update.yml

PLAY [all] *********************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [10.211.55.19]
ok: [10.211.55.201]

TASK [update package - CentOS] *************************************************************************************
skipping: [10.211.55.19]
changed: [10.211.55.201]

TASK [update package - OpenSUSE] ***********************************************************************************
skipping: [10.211.55.201]
changed: [10.211.55.19]

PLAY RECAP *********************************************************************************************************
10.211.55.19               : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
10.211.55.201              : ok=2    changed=1    unreachable=0    failed=0    skipped=1    rescued=0    ignored=0
```



ANSIBLE ARCHITECTURE



users

ANSIBLE AUTOMATION ENGINE

INVENTORY

API

MODULES

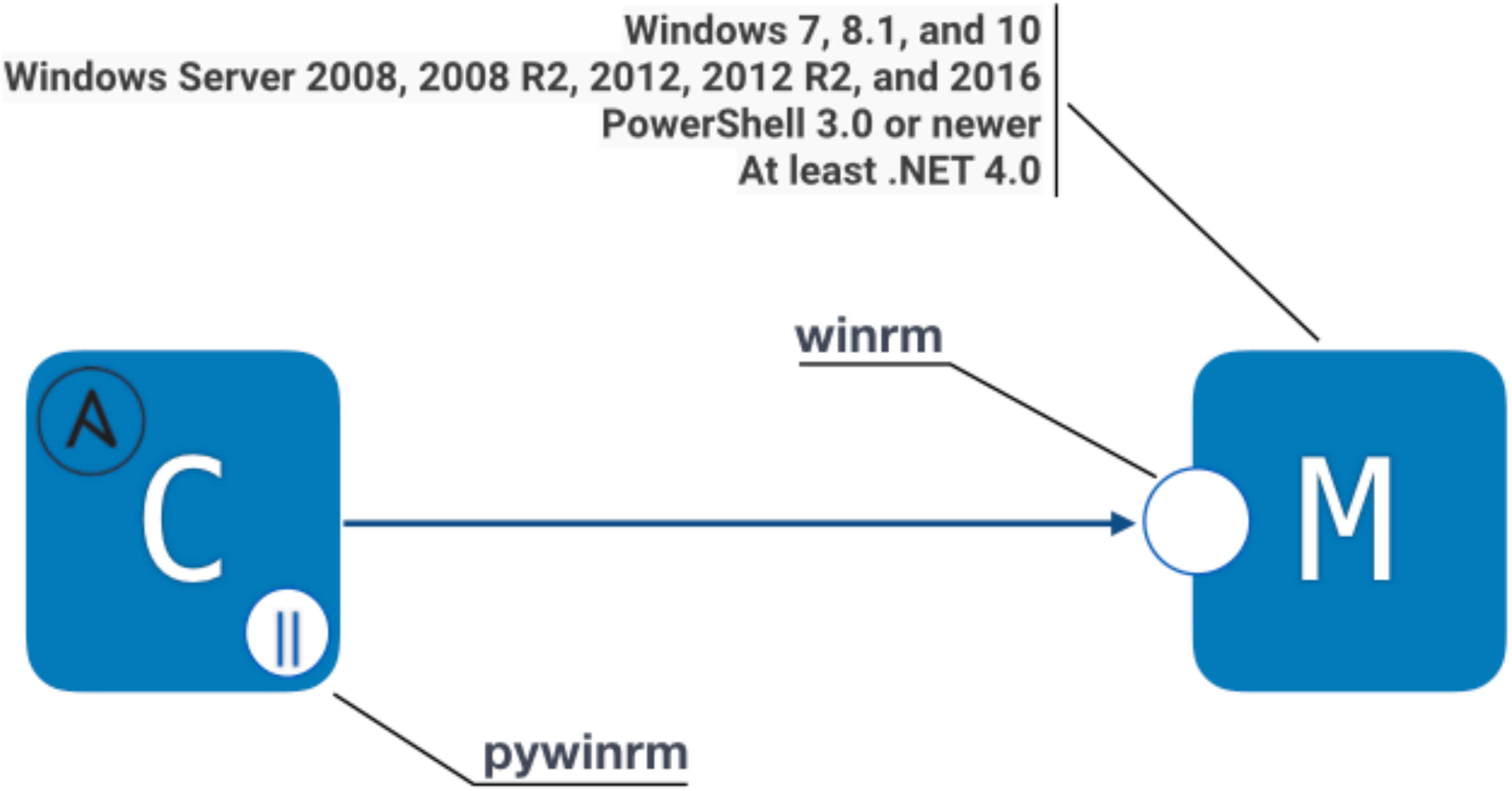
PLUGINS

Ansible Playbook

Windows

HOSTS

NETWORKING

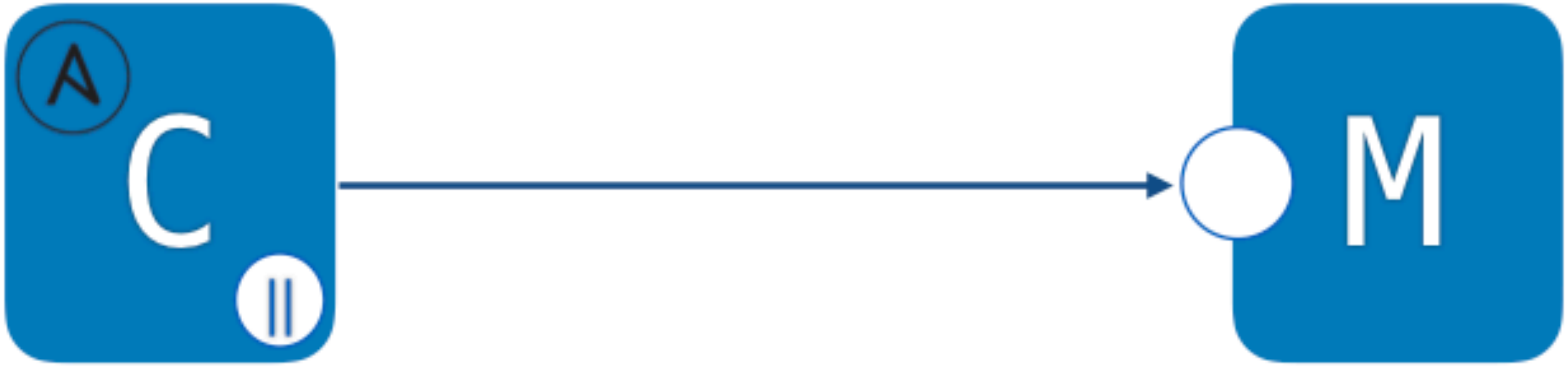


Windows 7, 8.1, and 10
Windows Server 2008, 2008 R2, 2012, 2012 R2, and 2016
PowerShell 3.0 or newer
At least .NET 4.0

winrm

pywinrm

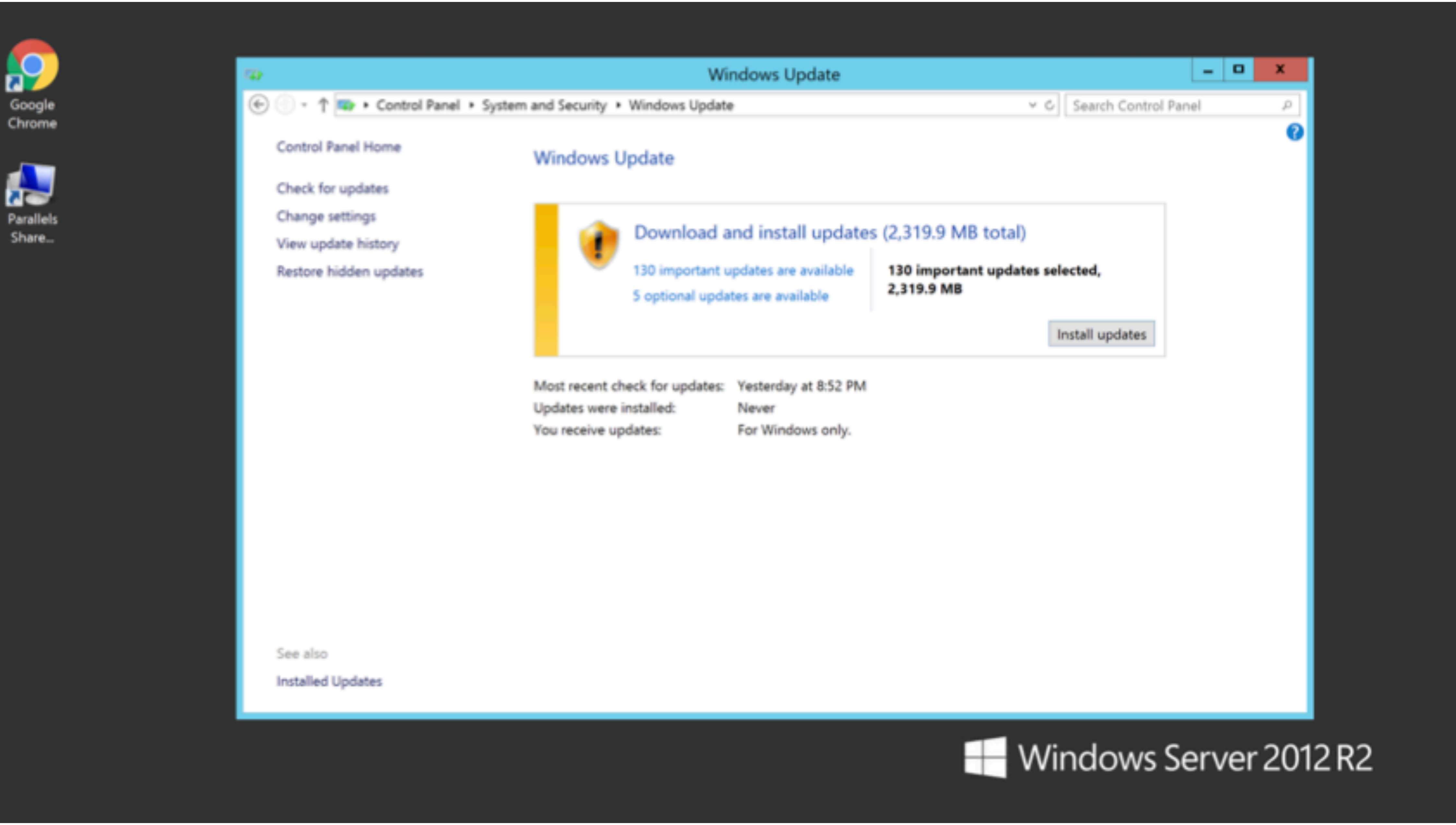
Controller & Host Requirements



inventory

```
[win]
10.211.55.253 ansible_host=myserver.example.com
10.211.55.252 ansible_host=myserver2.example.com
```

```
[win:vars]
ansible_user: administrator
ansible_password: password
ansible_port: 5986
ansible_connection: winrm
ansible_winrm_server_cert_validation: ignore
```



Windows Update

Download and install updates (2,319.9 MB total)

130 important updates are available

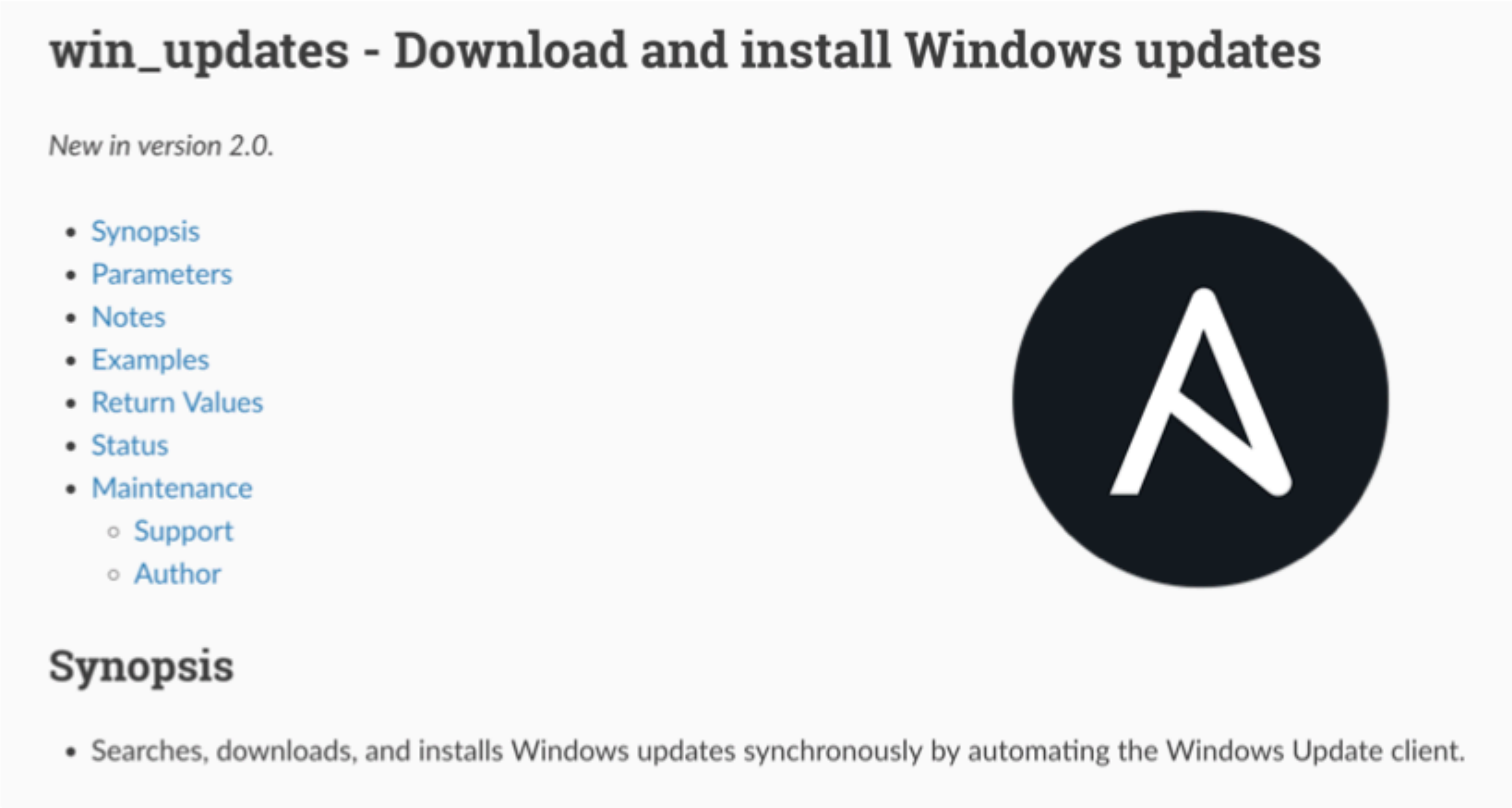
5 optional updates are available

130 important updates selected, 2,319.9 MB

Most recent check for updates: Yesterday at 8:52 PM

Updates were installed: Never

You receive updates: For Windows only.



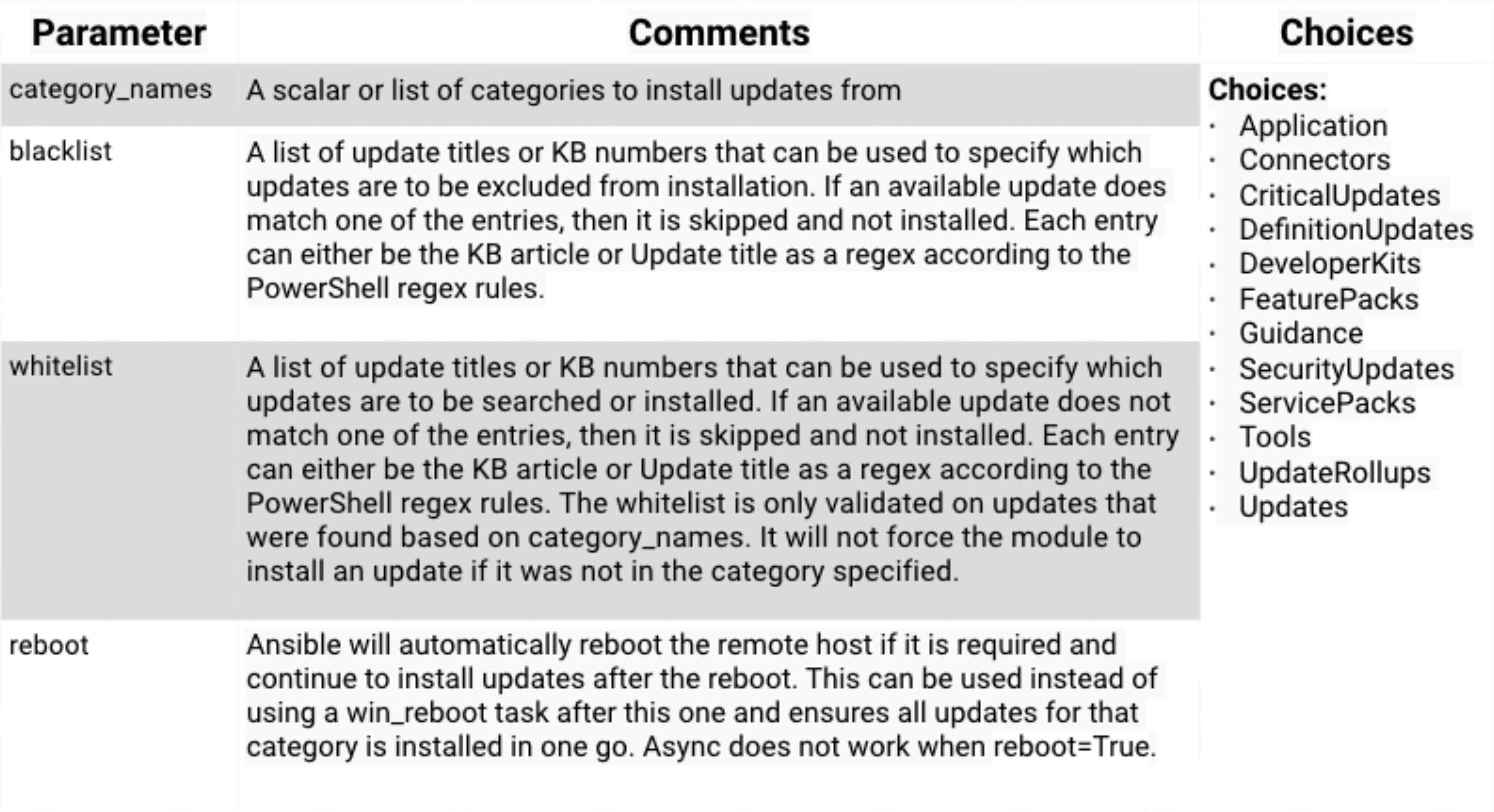
win_updates - Download and install Windows updates

New in version 2.0.

- Synopsis
- Parameters
- Notes
- Examples
- Return Values
- Status
- Maintenance
  - Support
  - Author

Synopsis

- Searches, downloads, and installs Windows updates synchronously by automating the Windows Update client.

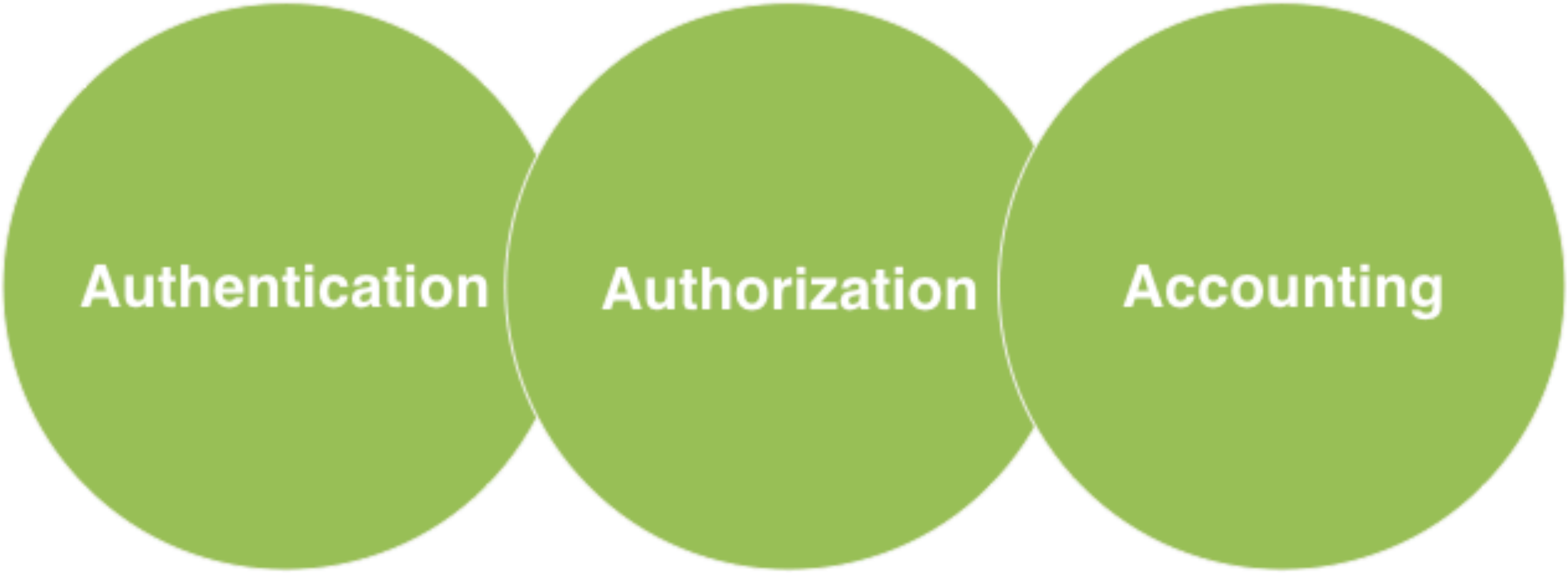


Parameter	Comments	Choices
category_names	A scalar or list of categories to install updates from	**Choices:** · Application · Connectors · CriticalUpdates · DefinitionUpdates · DeveloperKits · FeaturePacks · Guidance · SecurityUpdates · ServicePacks · Tools · UpdateRollups · Updates
blacklist	A list of update titles or KB numbers that can be used to specify which updates are to be excluded from installation. If an available update does match one of the entries, then it is skipped and not installed. Each entry can either be the KB article or Update title as a regex according to the PowerShell regex rules.	
whitelist	A list of update titles or KB numbers that can be used to specify which updates are to be searched or installed. If an available update does not match one of the entries, then it is skipped and not installed. Each entry can either be the KB article or Update title as a regex according to the PowerShell regex rules. The whitelist is only validated on updates that were found based on category_names. It will not force the module to install an update if it was not in the category specified.	
reboot	Ansible will automatically reboot the remote host if it is required and continue to install updates after the reboot. This can be used instead of using a win_reboot task after this one and ensures all updates for that category is installed in one go. Async does not work when reboot=True.	

```yaml
- name: Install all security, critical, and rollup updates without a scheduled task
  win_updates:
    category_names:
      - SecurityUpdates
      - CriticalUpdates
      - UpdateRollups
```

```yaml
- name: Install only particular updates based on the KB numbers
  win_updates:
    category_name:
    - SecurityUpdates
    whitelist:
    - KB4056892
    - KB4073117
```

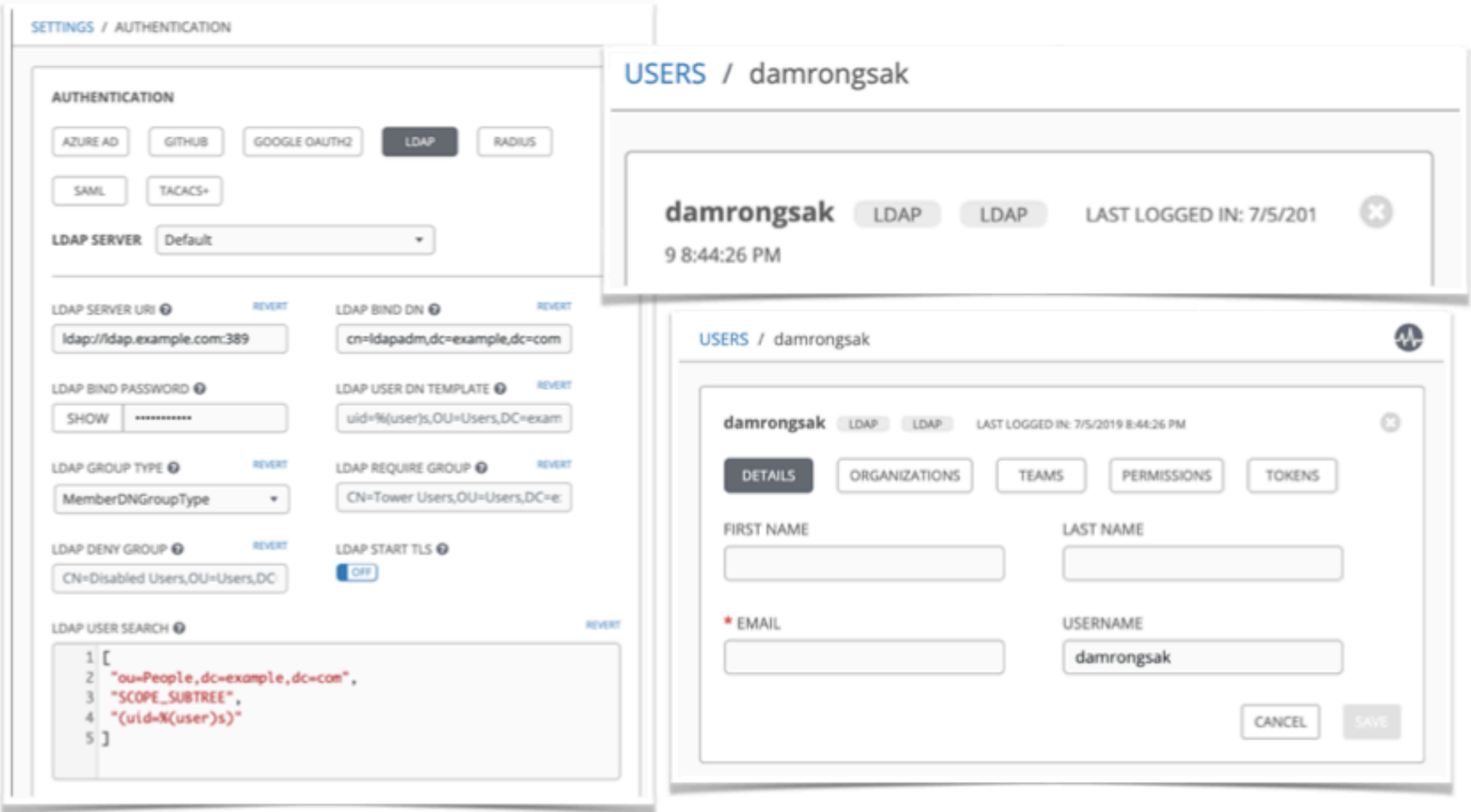
AWX



Authentication

Authorization

Accounting



LDAP SERVER: Default

LDAP SERVER URI: ldap://ldap.example.com:389

LDAP BIND DN: cn=ldapadm,dc=example,dc=com

LDAP BIND PASSWORD: ••••••••••••

LDAP USER DN TEMPLATE: uid=%(user)s,OU=Users,DC=exam

LDAP GROUP TYPE: MemberDNGroupType

LDAP REQUIRE GROUP: CN=Tower Users,OU=Users,DC=e

LDAP DENY GROUP: CN=Disabled Users,OU=Users,DC

LDAP START TLS: OFF

LDAP USER SEARCH

```
[
 "ou=People,dc=example,dc=com",
 "SCOPE_SUBTREE",
 "(uid=%(user)s)"
]
```

USERS / damrongsak

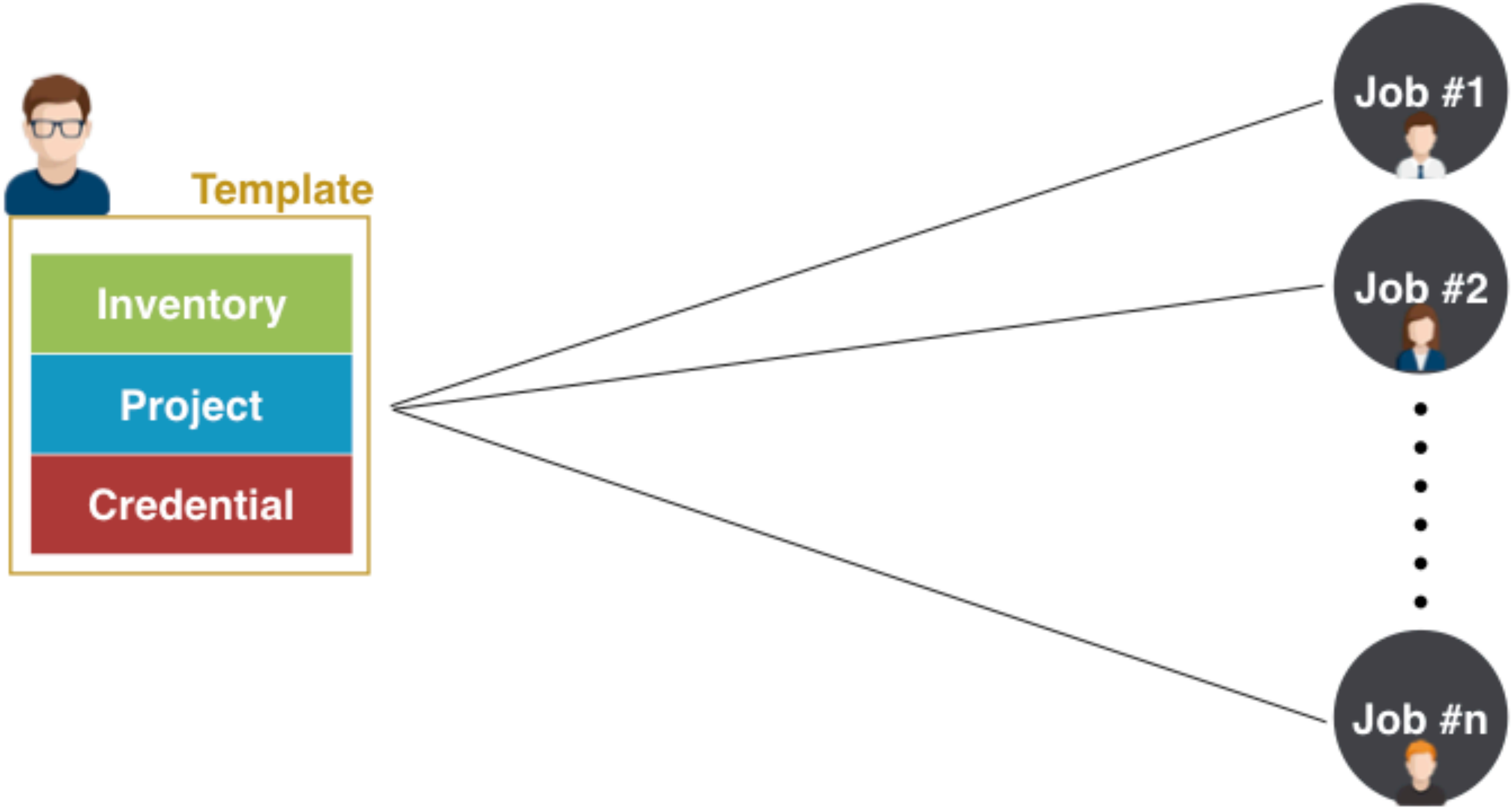
damrongsak LDAP LDAP LAST LOGGED IN: 7/5/2019 8:44:26 PM

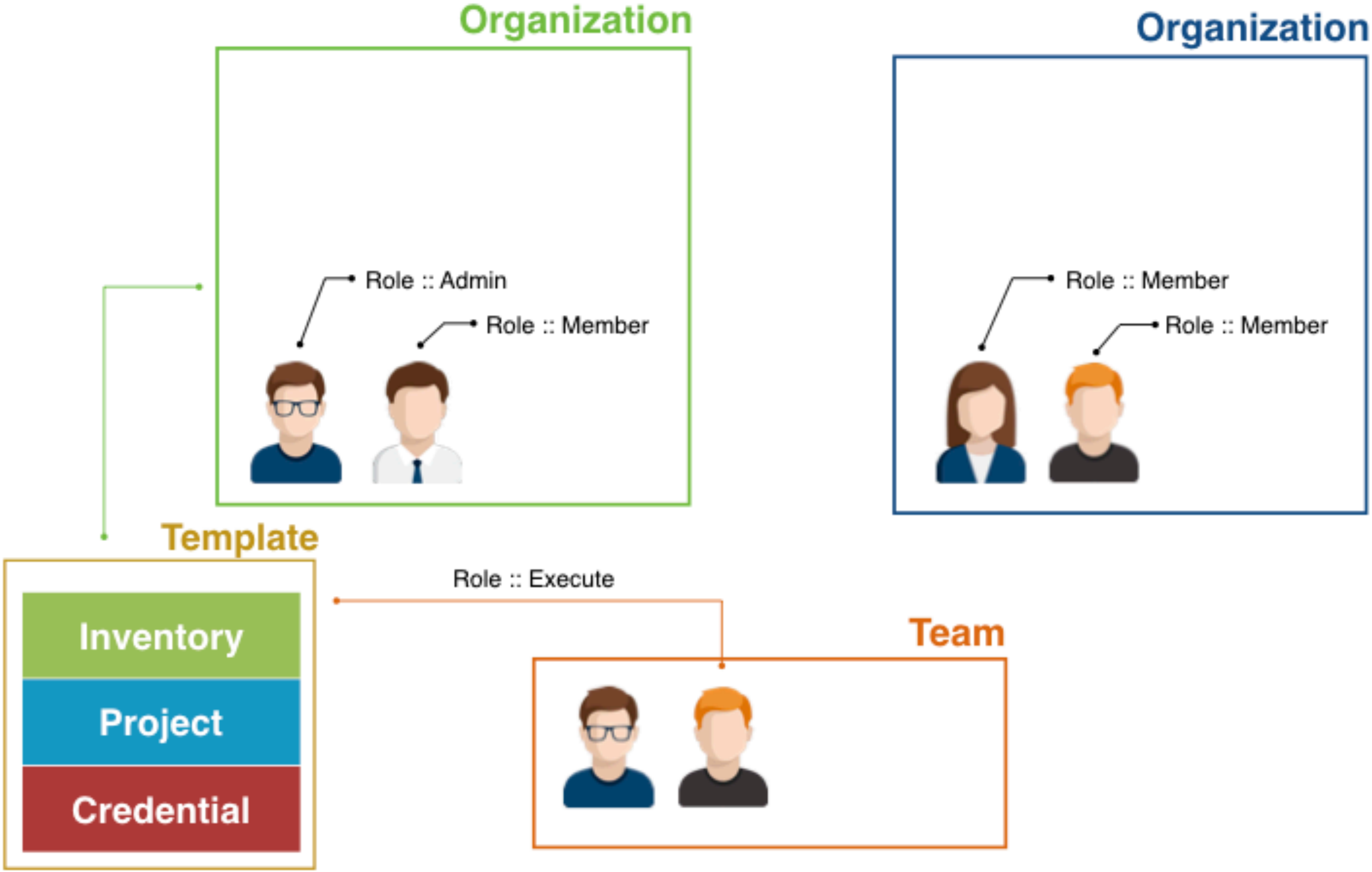
FIRST NAME:

LAST NAME:

* EMAIL:

USERNAME: damrongsak


Template
Inventory
Project
Credential
Job #1
Job #2
Job #n



Organization

Role :: Admin

Role :: Member

Organization

Role :: Member

Role :: Member

Template

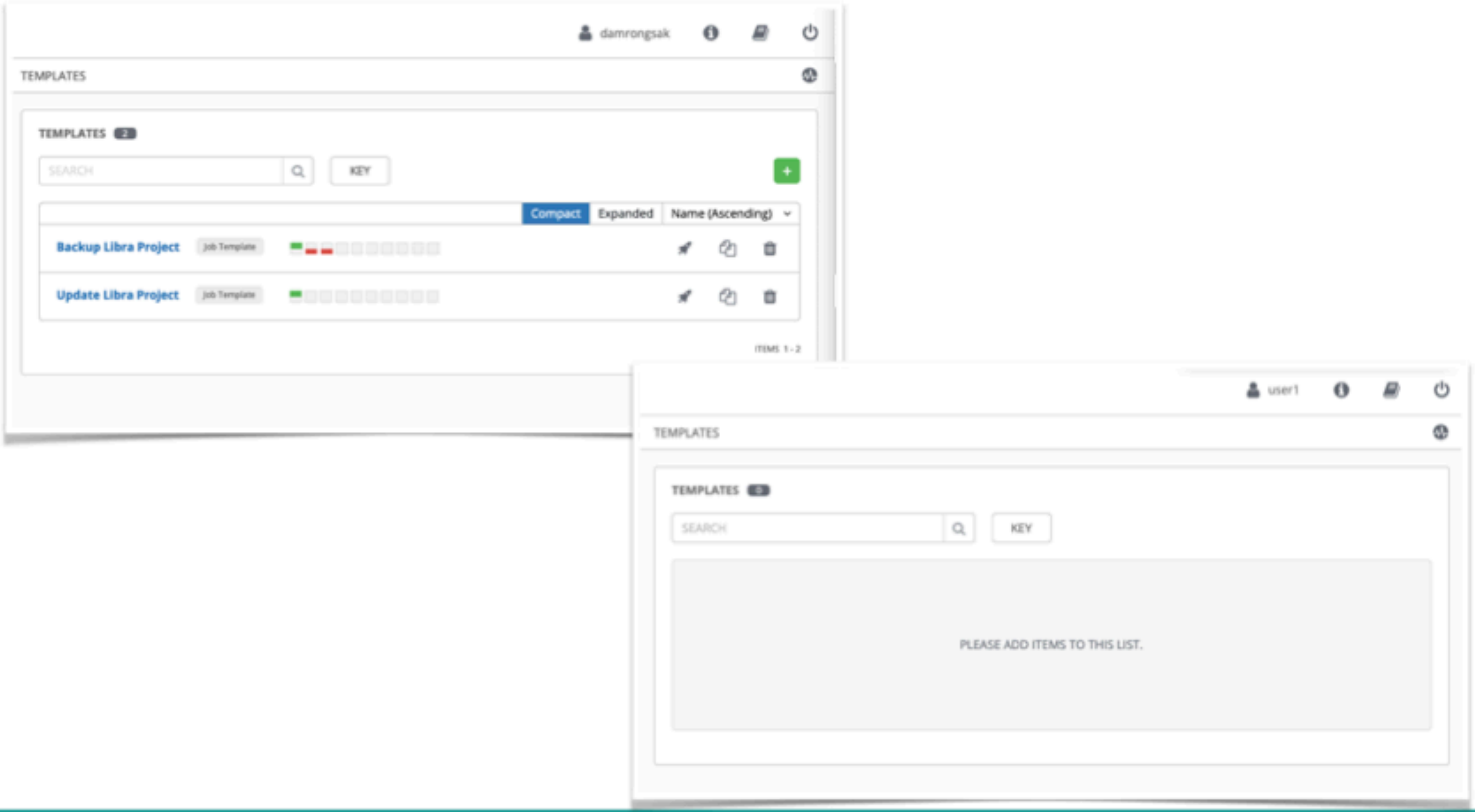
Inventory

Project

Credential

Role :: Execute

Team



damrongsak

TEMPLATES

TEMPLATES 2

SEARCH

KEY

Compact | Expanded | Name (Ascending)

Backup Libra Project | Job Template

Update Libra Project | Job Template

ITEMS 1 - 2

user1

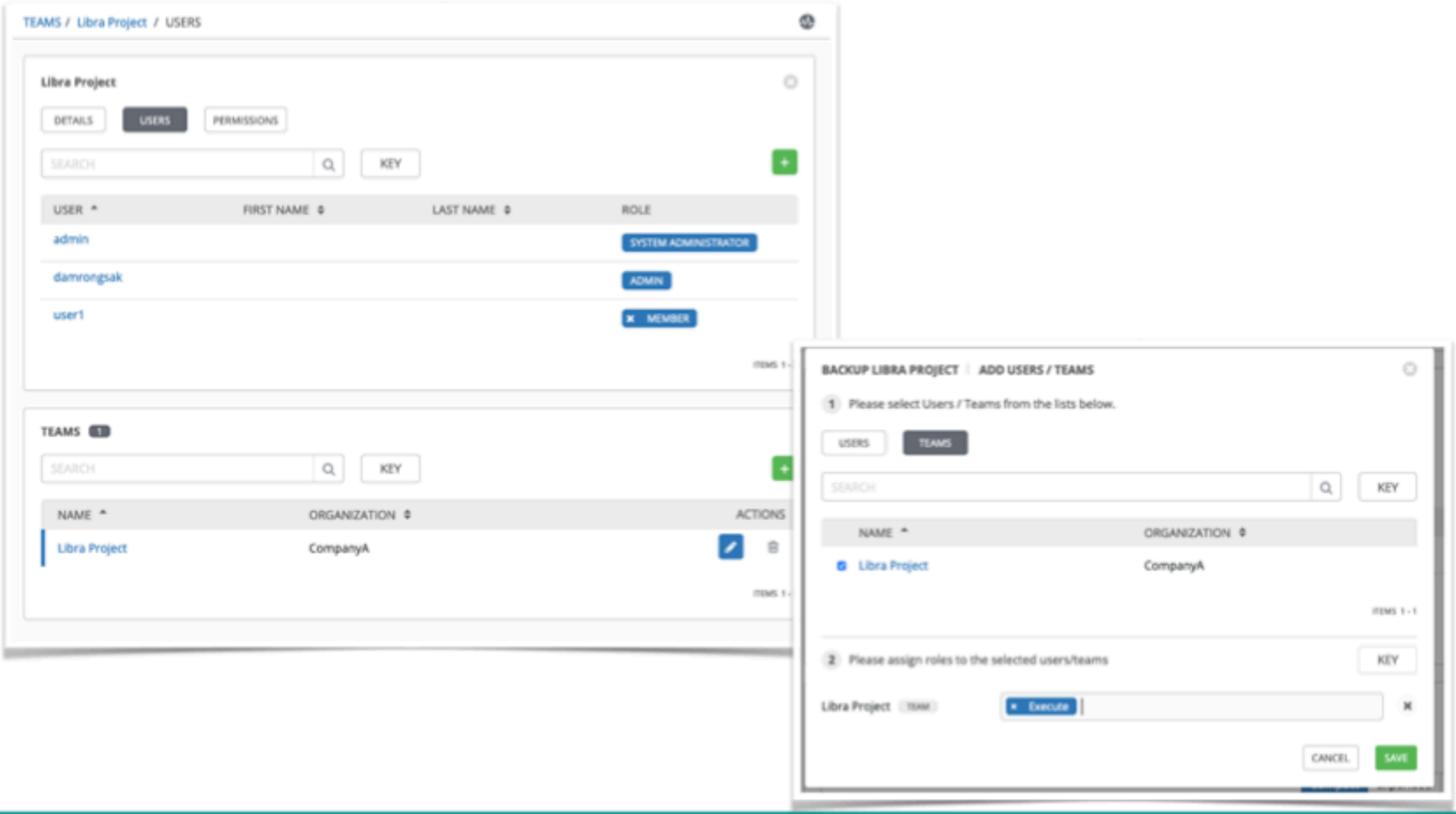
TEMPLATES

TEMPLATES 0

SEARCH

KEY

PLEASE ADD ITEMS TO THIS LIST.



TEAMS / Libra Project / USERS

Libra Project

DETAILS | USERS | PERMISSIONS

SEARCH | KEY | +

USER	FIRST NAME	LAST NAME	ROLE
admin			SYSTEM ADMINISTRATOR
damrongsak			ADMIN
user1			✕ MEMBER

ITEMS 1 -

TEAMS 1

SEARCH | KEY | +

NAME	ORGANIZATION	ACTIONS
Libra Project	CompanyA	

ITEMS 1 -

BACKUP LIBRA PROJECT | ADD USERS / TEAMS

1 Please select Users / Teams from the lists below.

USERS | TEAMS

SEARCH | KEY

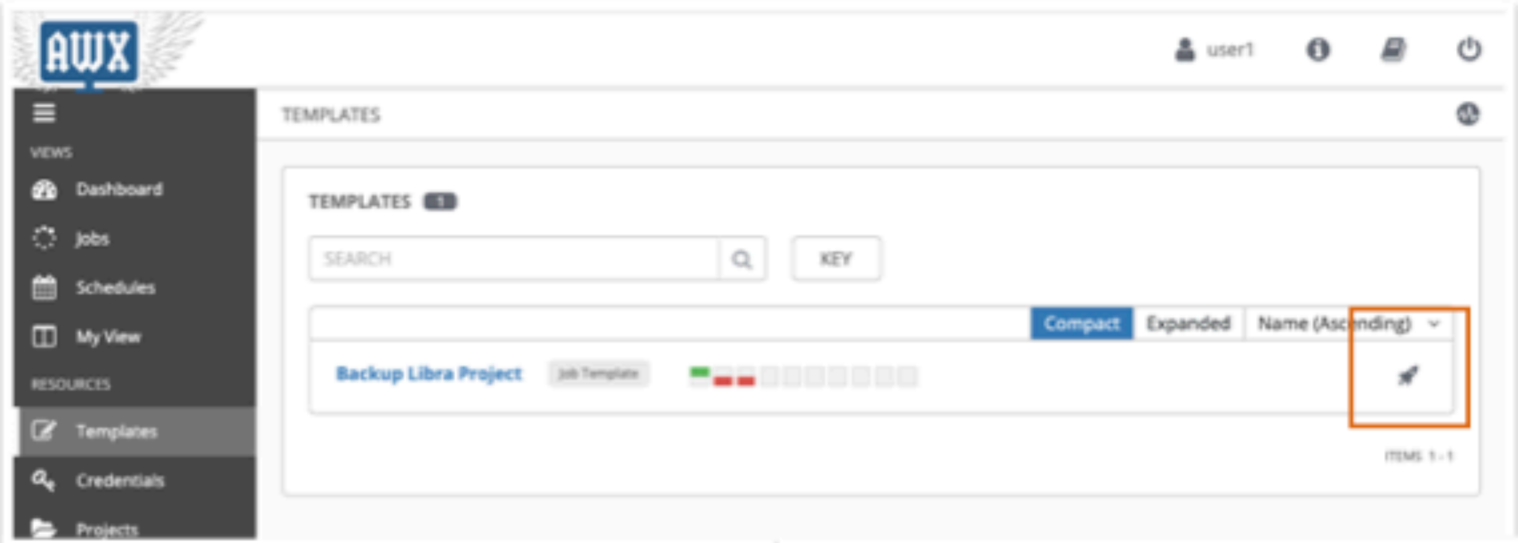
	NAME	ORGANIZATION
☑	Libra Project	CompanyA

ITEMS 1 - 1

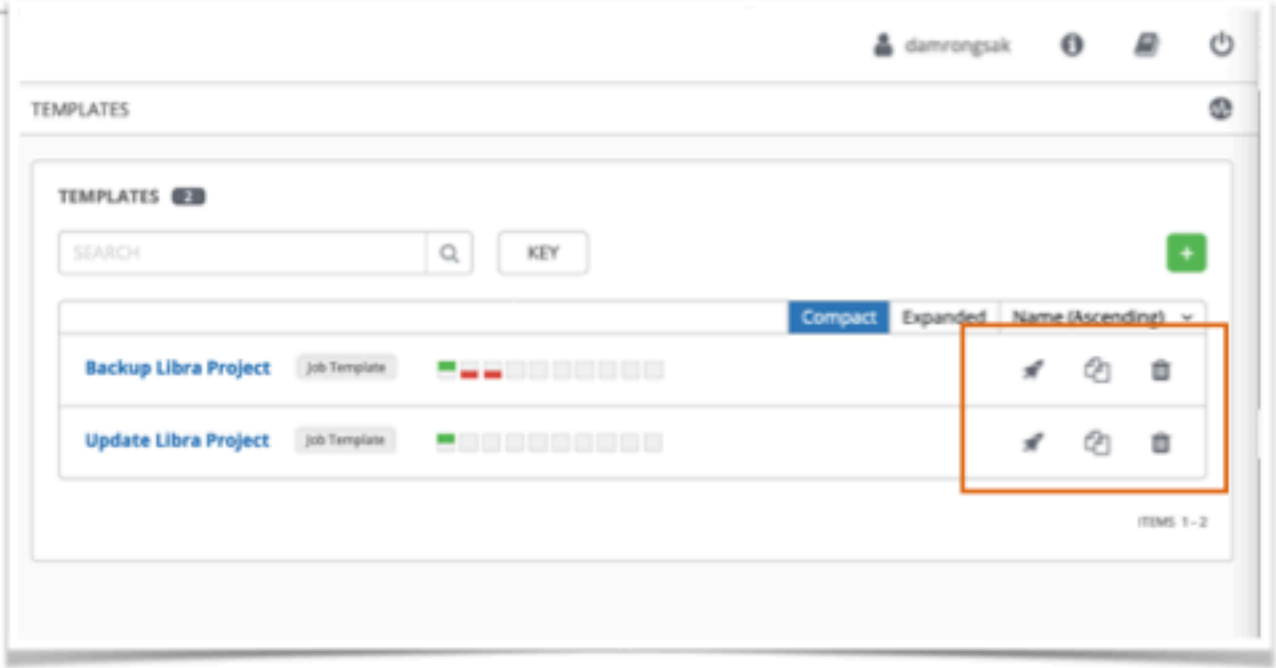
2 Please assign roles to the selected users/teams | KEY

Libra Project TEAM | ✕ Execute

CANCEL | SAVE

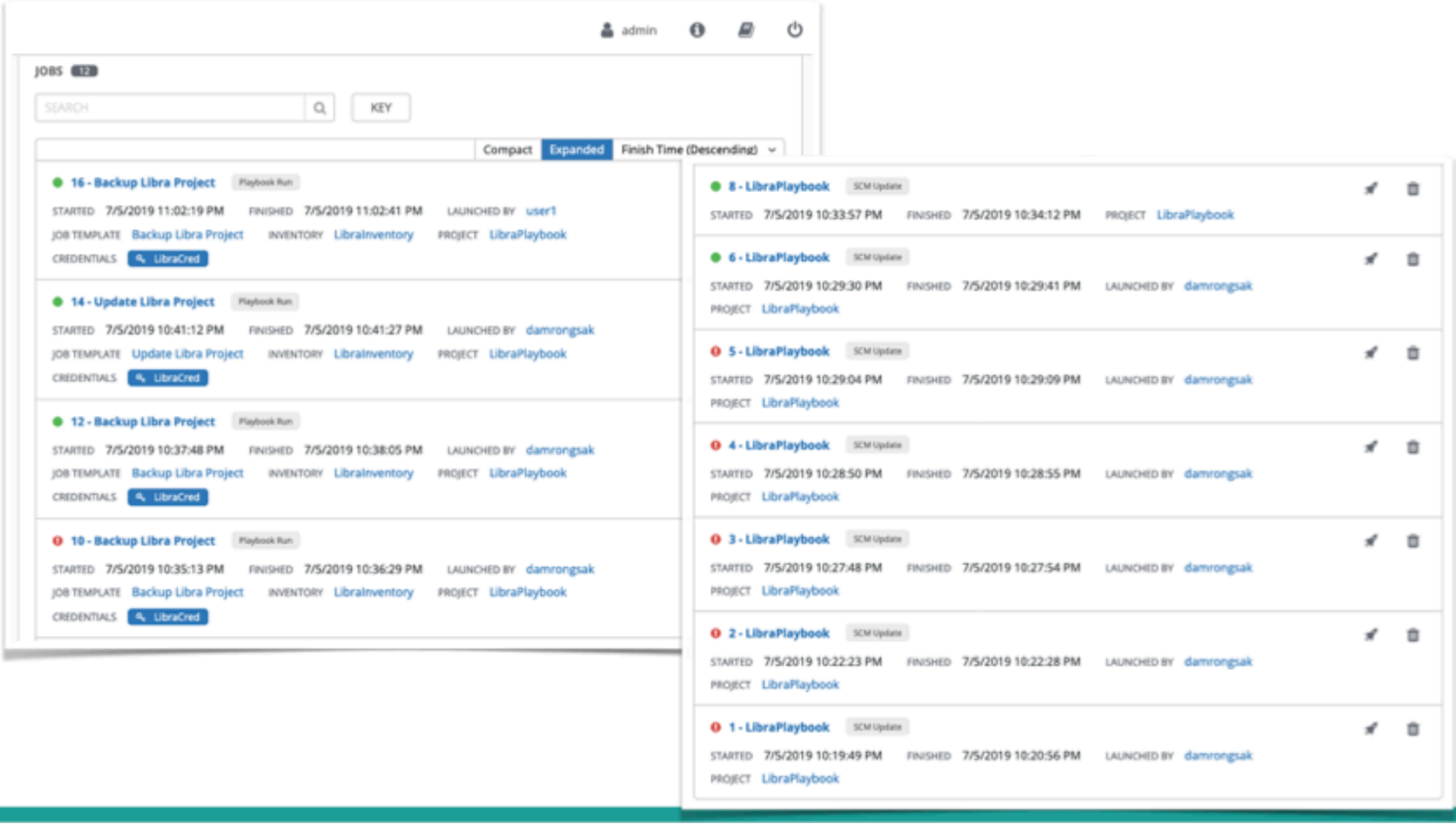


Backup Libra Project — Job Template



Backup Libra Project — Job Template

Update Libra Project — Job Template



admin

JOBS 12

SEARCH

KEY

Compact | Expanded | Finish Time (Descending)

● 16 - Backup Libra Project — Playbook Run
STARTED 7/5/2019 11:02:19 PM FINISHED 7/5/2019 11:02:41 PM LAUNCHED BY user1
JOB TEMPLATE Backup Libra Project INVENTORY LibraInventory PROJECT LibraPlaybook
CREDENTIALS LibraCred

● 14 - Update Libra Project — Playbook Run
STARTED 7/5/2019 10:41:12 PM FINISHED 7/5/2019 10:41:27 PM LAUNCHED BY damrongsak
JOB TEMPLATE Update Libra Project INVENTORY LibraInventory PROJECT LibraPlaybook
CREDENTIALS LibraCred

● 12 - Backup Libra Project — Playbook Run
STARTED 7/5/2019 10:37:48 PM FINISHED 7/5/2019 10:38:05 PM LAUNCHED BY damrongsak
JOB TEMPLATE Backup Libra Project INVENTORY LibraInventory PROJECT LibraPlaybook
CREDENTIALS LibraCred

❗ 10 - Backup Libra Project — Playbook Run
STARTED 7/5/2019 10:35:13 PM FINISHED 7/5/2019 10:36:29 PM LAUNCHED BY damrongsak
JOB TEMPLATE Backup Libra Project INVENTORY LibraInventory PROJECT LibraPlaybook
CREDENTIALS LibraCred

● 8 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:33:57 PM FINISHED 7/5/2019 10:34:12 PM PROJECT LibraPlaybook

● 6 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:29:30 PM FINISHED 7/5/2019 10:29:41 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook

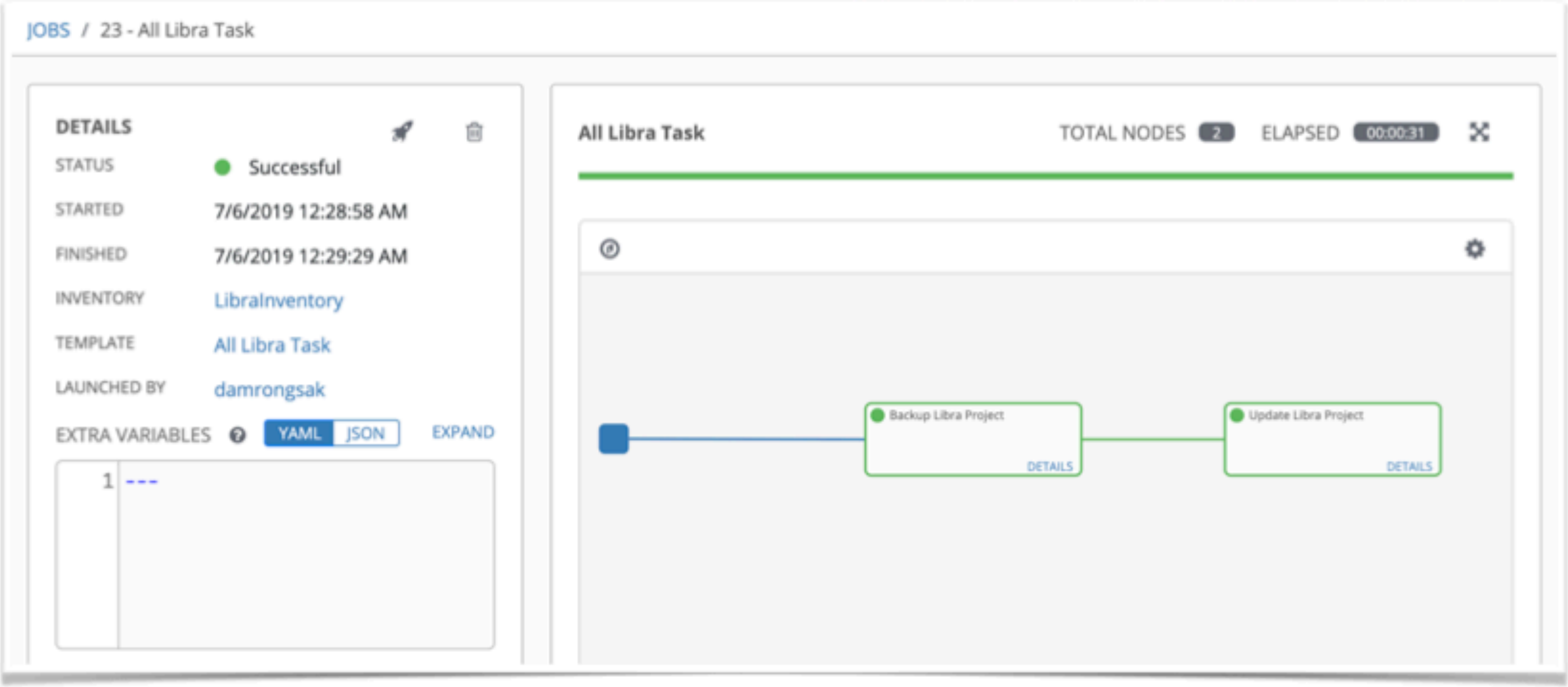
❗ 5 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:29:04 PM FINISHED 7/5/2019 10:29:09 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook

❗ 4 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:28:50 PM FINISHED 7/5/2019 10:28:55 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook

❗ 3 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:27:48 PM FINISHED 7/5/2019 10:27:54 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook

❗ 2 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:22:23 PM FINISHED 7/5/2019 10:22:28 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook

❗ 1 - LibraPlaybook — SCM Update
STARTED 7/5/2019 10:19:49 PM FINISHED 7/5/2019 10:20:56 PM LAUNCHED BY damrongsak
PROJECT LibraPlaybook



JOBS / 23 - All Libra Task

DETAILS

STATUS ● Successful

STARTED 7/6/2019 12:28:58 AM

FINISHED 7/6/2019 12:29:29 AM

INVENTORY LibraInventory

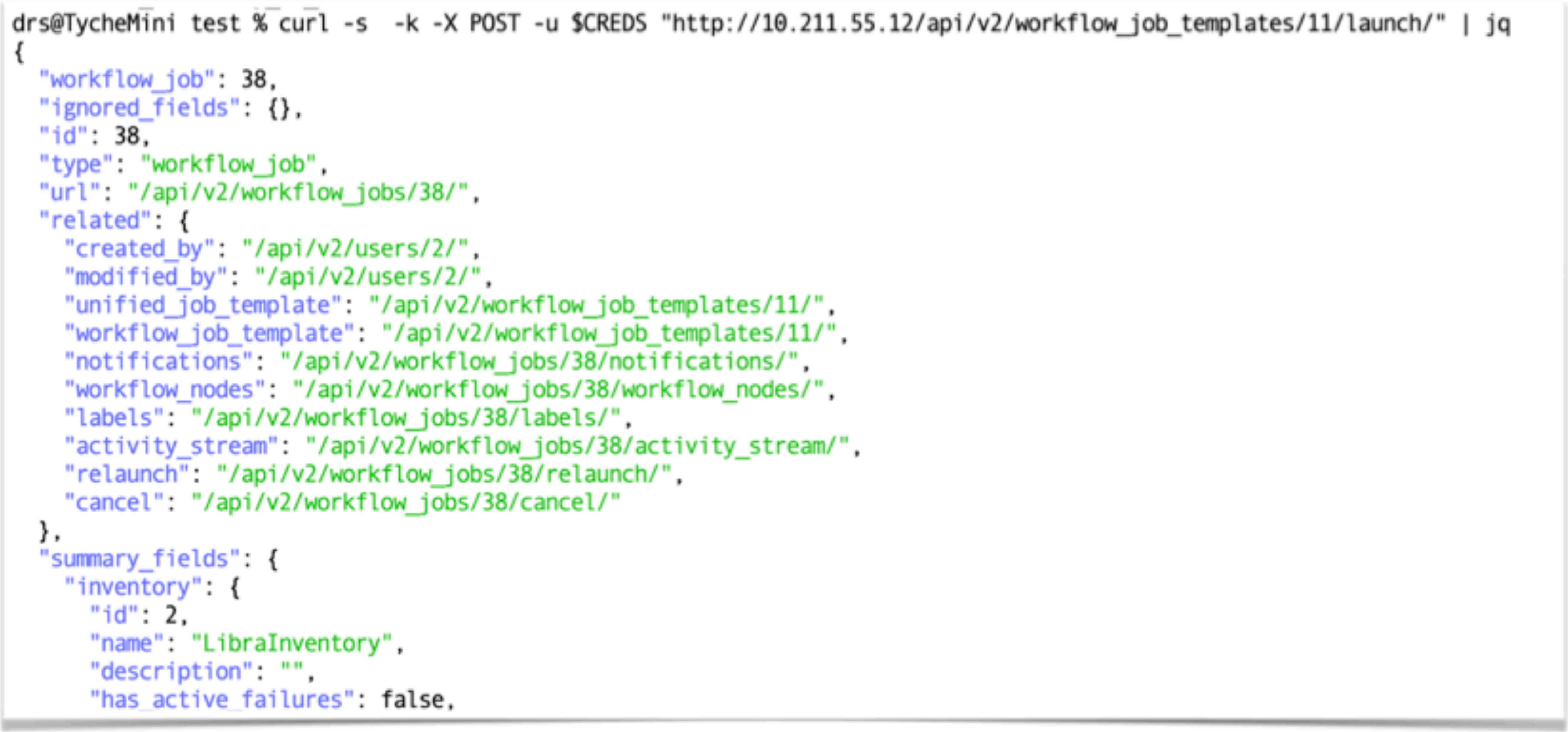
TEMPLATE All Libra Task

LAUNCHED BY damrongsak

EXTRA VARIABLES YAML JSON EXPAND

```
---
```

All Libra Task TOTAL NODES 2 ELAPSED 00:00:31

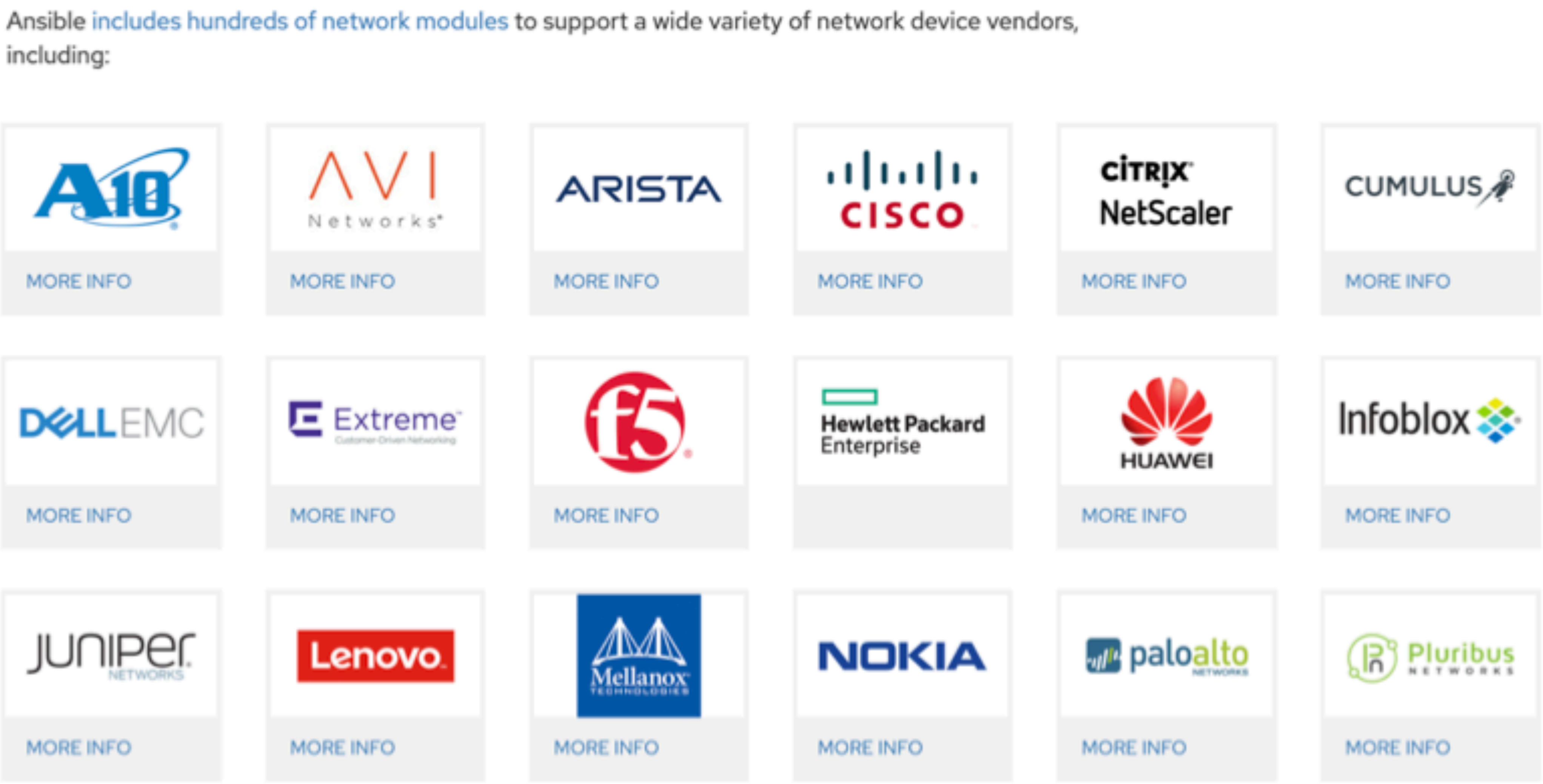


```
drs@TycheMini test % curl -s  -k -X POST -u $CREDS "http://10.211.55.12/api/v2/workflow_job_templates/11/launch/" | jq
{
  "workflow_job": 38,
  "ignored_fields": {},
  "id": 38,
  "type": "workflow_job",
  "url": "/api/v2/workflow_jobs/38/",
  "related": {
    "created_by": "/api/v2/users/2/",
    "modified_by": "/api/v2/users/2/",
    "unified_job_template": "/api/v2/workflow_job_templates/11/",
    "workflow_job_template": "/api/v2/workflow_job_templates/11/",
    "notifications": "/api/v2/workflow_jobs/38/notifications/",
    "workflow_nodes": "/api/v2/workflow_jobs/38/workflow_nodes/",
    "labels": "/api/v2/workflow_jobs/38/labels/",
    "activity_stream": "/api/v2/workflow_jobs/38/activity_stream/",
    "relaunch": "/api/v2/workflow_jobs/38/relaunch/",
    "cancel": "/api/v2/workflow_jobs/38/cancel/"
  },
  "summary_fields": {
    "inventory": {
      "id": 2,
      "name": "LibraInventory",
      "description": "",
      "has_active_failures": false,
```

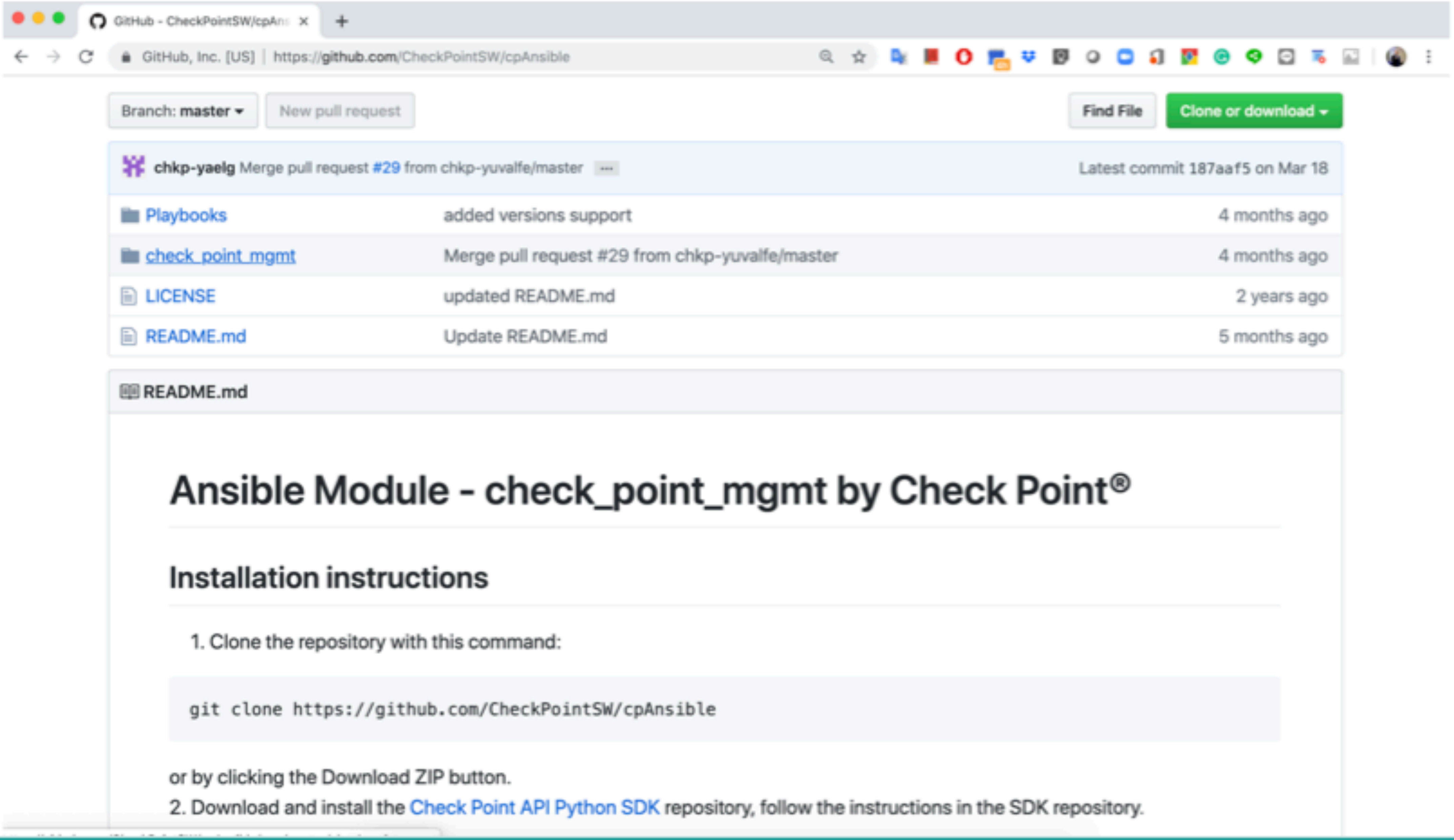
thank
you

Security automation – the use of information technology in place of manual processes for cyber incident response and security event management.

https://www.microsoft.com/security/blog/2017/08/03/top-5-best-practices-to-automate-security-operations/



Ansible includes hundreds of network modules to support a wide variety of network device vendors, including:


GitHub - CheckPointSW/cpAn
GitHub, Inc. [US] | https://github.com/CheckPointSW/cpAnsible
Branch: master
New pull request
Find File
Clone or download

chkp-yaelg Merge pull request #29 from chkp-yuvalfe/master … Latest commit 187aaf5 on Mar 18

Playbooks	added versions support	4 months ago
check_point_mgmt	Merge pull request #29 from chkp-yuvalfe/master	4 months ago
LICENSE	updated README.md	2 years ago
README.md	Update README.md	5 months ago

README.md

Ansible Module - check_point_mgmt by Check Point®

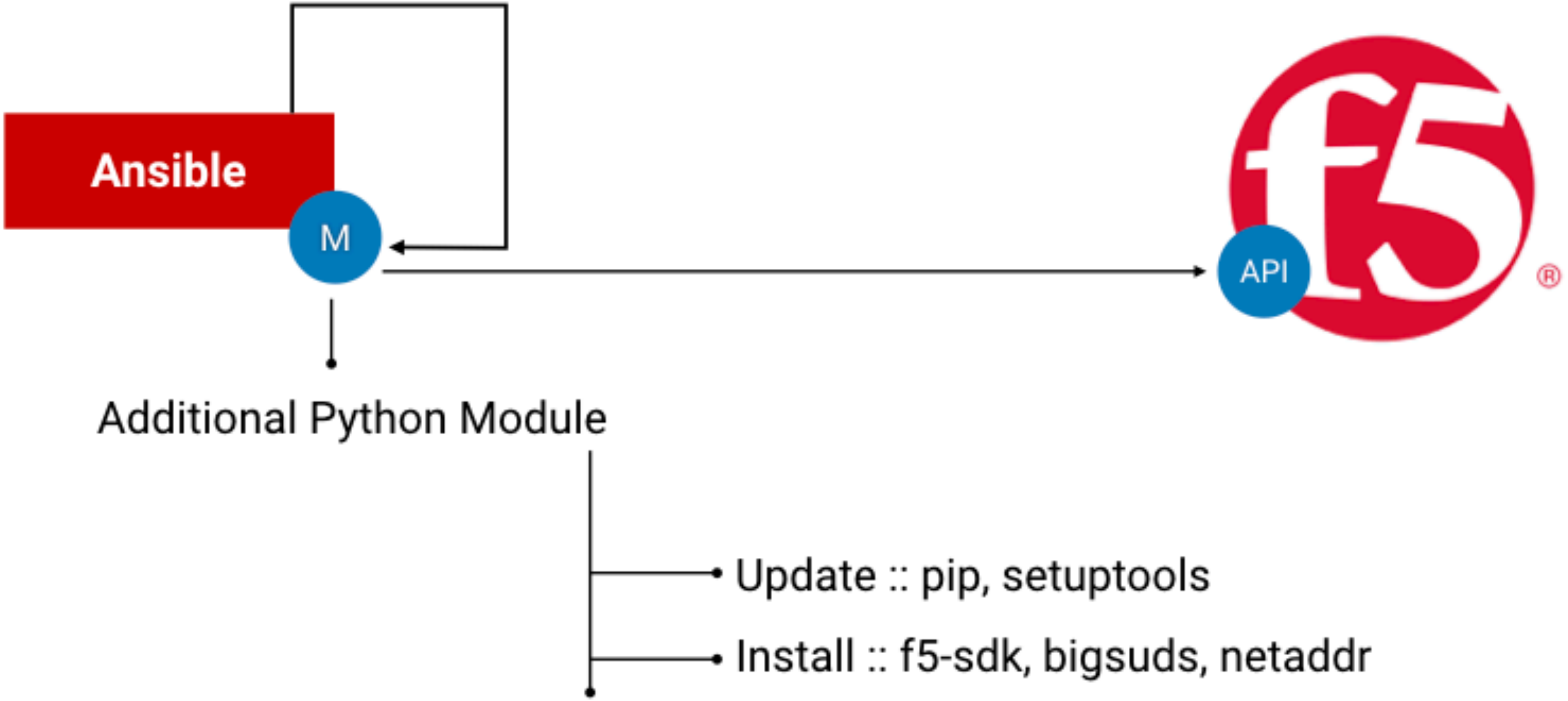
Installation instructions

1. Clone the repository with this command:

```
git clone https://github.com/CheckPointSW/cpAnsible
```

or by clicking the Download ZIP button.

2. Download and install the Check Point API Python SDK repository, follow the instructions in the SDK repository.



Additional Python Module

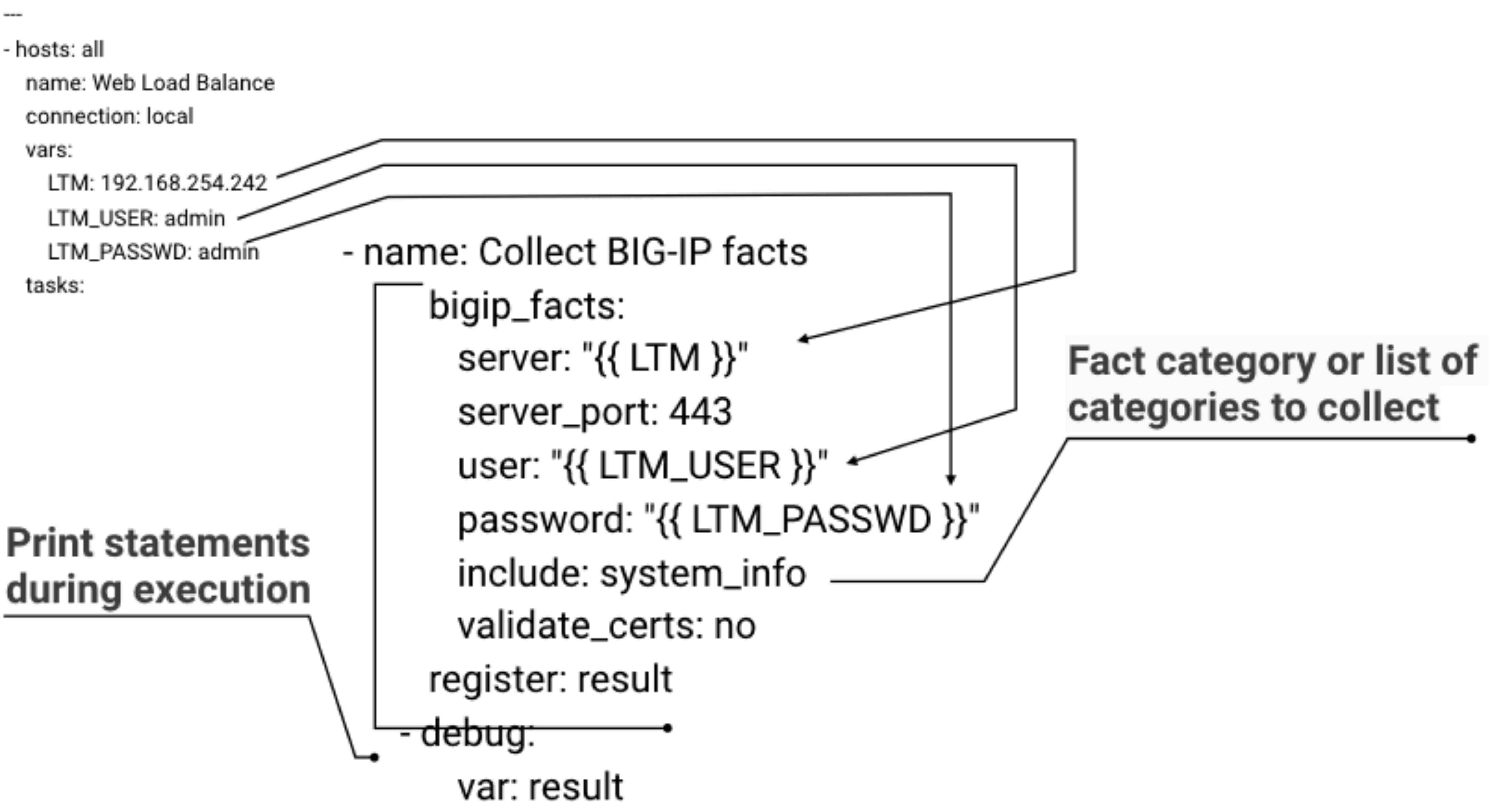
Update :: pip, setuptools

Install :: f5-sdk, bigsuds, netaddr

```
# pip install —upgrade pip setuptools
# pip install f5-sdk big suds netaddr
```

```
---
-  hosts: all
   name: Web Load Balance
   connection: local
   vars:
     LTM: 192.168.254.242
     LTM_USER: admin
     LTM_PASSWD: admin
   tasks:
```

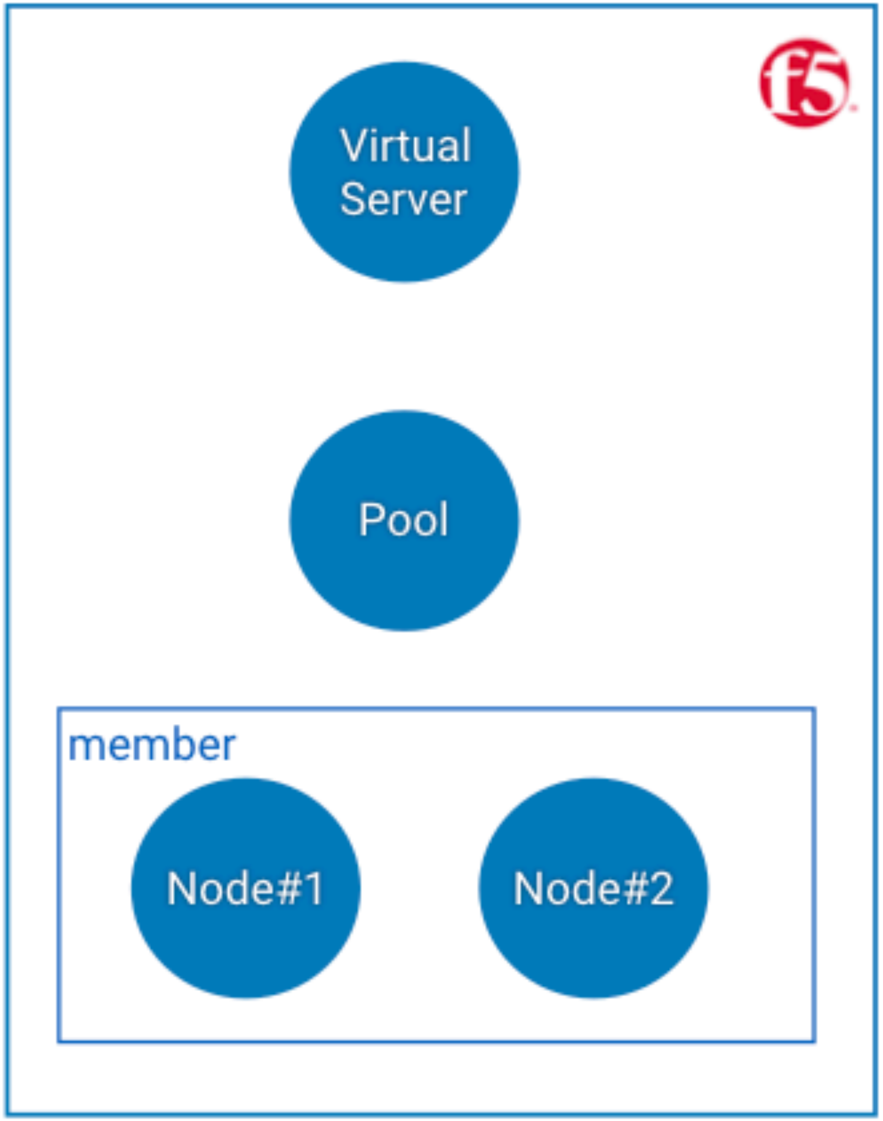
Define Variable



```
---
- hosts: all
  name: Web Load Balance
  connection: local
  vars:
    LTM: 192.168.254.242
    LTM_USER: admin
    LTM_PASSWD: admin
  tasks:
    - name: Collect BIG-IP facts
      bigip_facts:
        server: "{{ LTM }}"
        server_port: 443
        user: "{{ LTM_USER }}"
        password: "{{ LTM_PASSWD }}"
        include: system_info
        validate_certs: no
      register: result
    - debug:
        var: result
```

Fact category or list of categories to collect

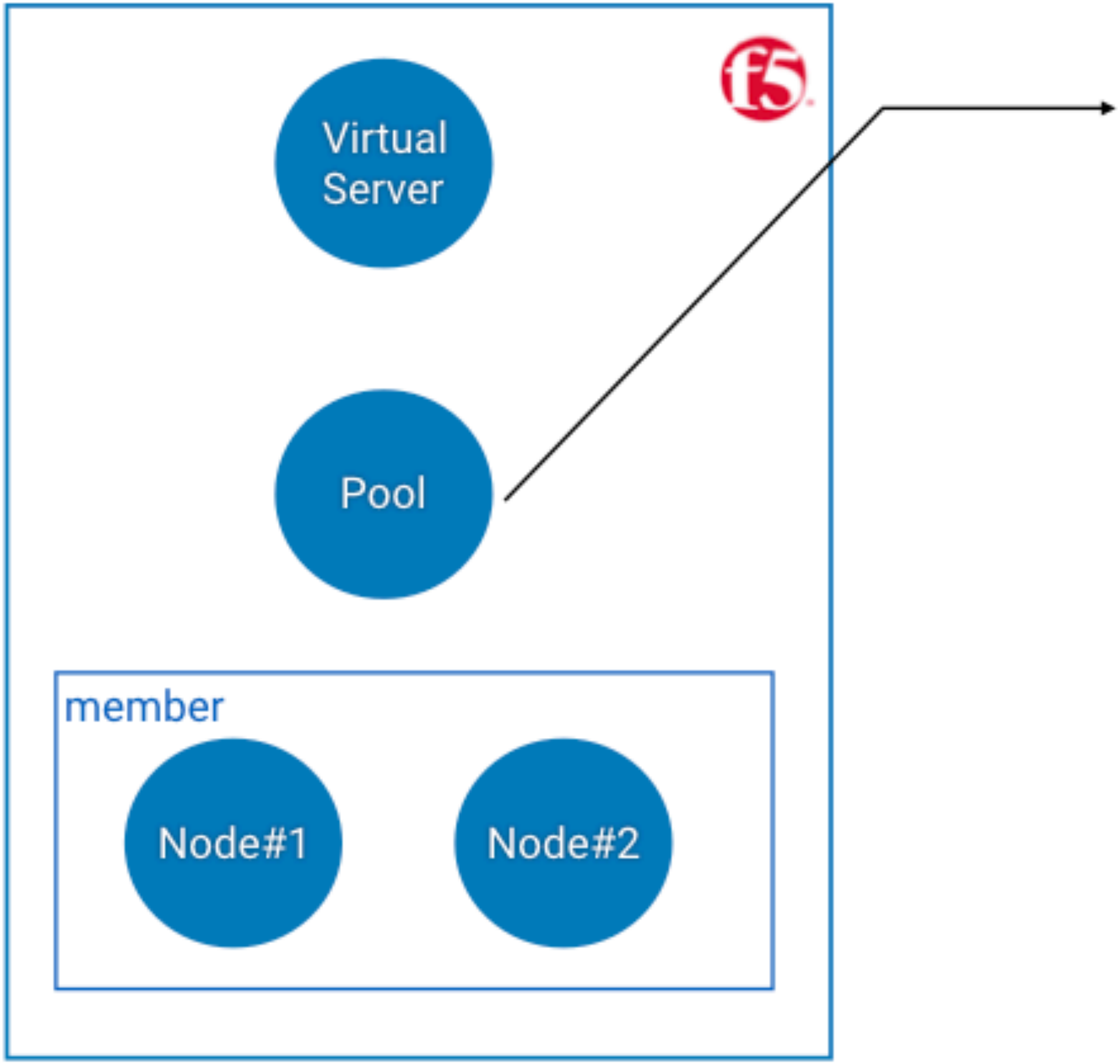
Print statements during execution



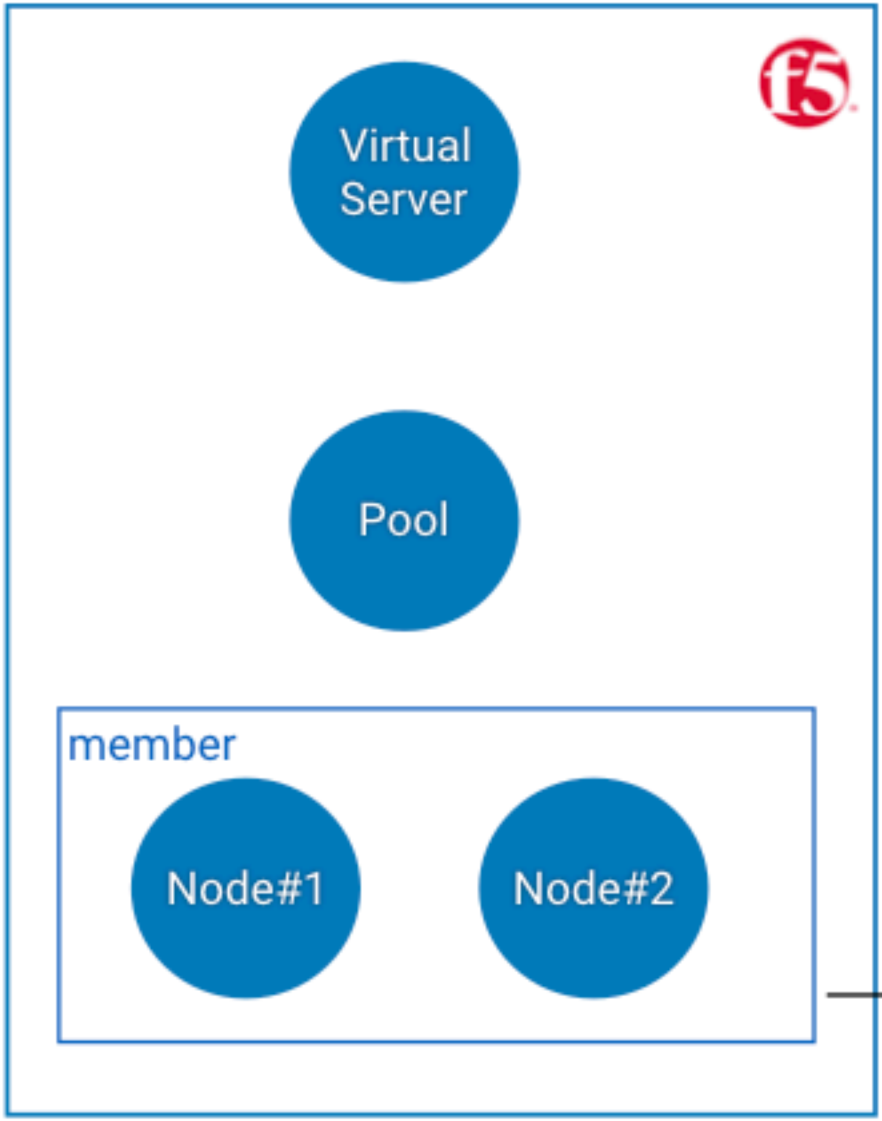
3. Create Virtual Server

1. Creating Pool

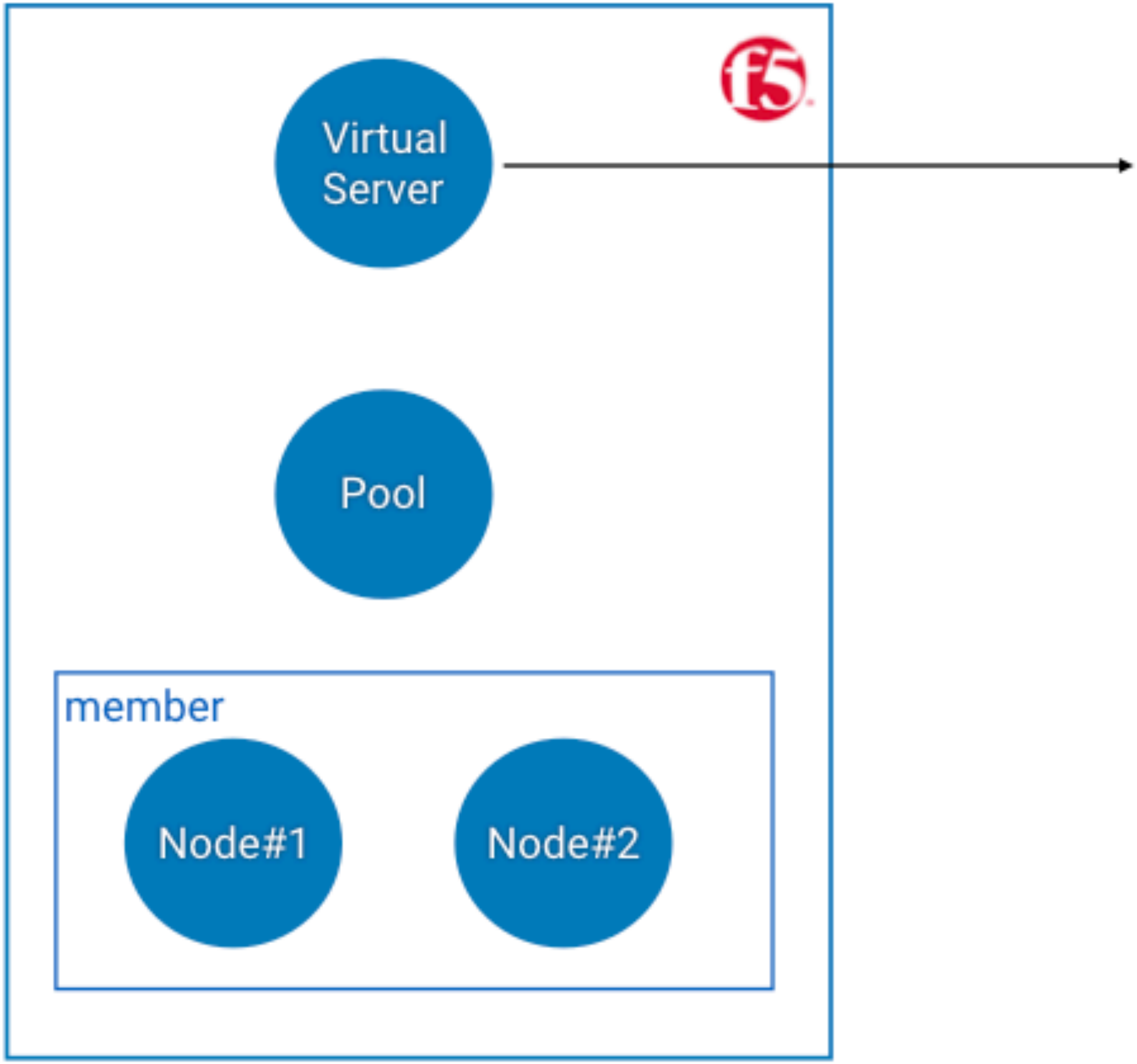
2. Add Node to Pool



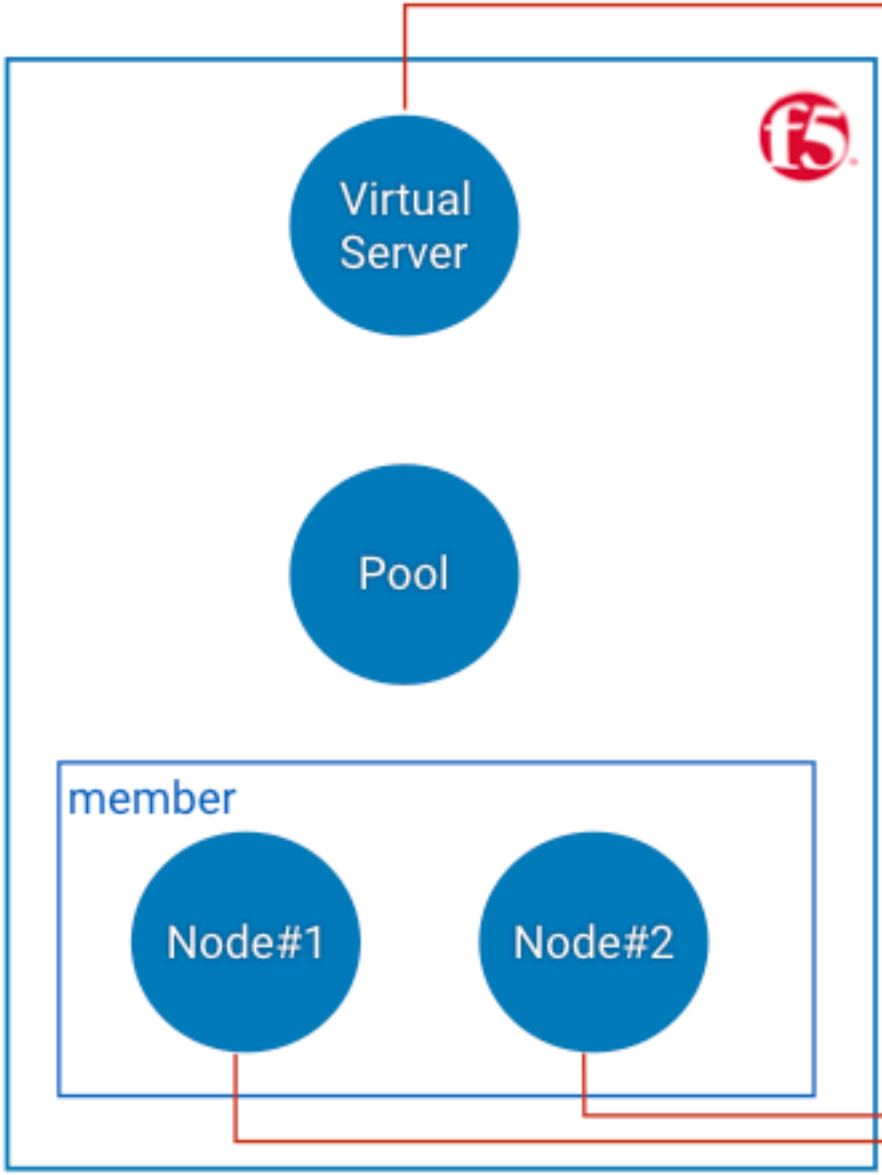
```
bigip_pool:
    server: "{{ LTM }}"
    user: "{{ LTM_USER }}"
    password: "{{ LTM_PASSWD }}"
    validate_certs: no
    state: present
    name: "{{ POOL_NAME }}"
    partition: Common
    lb_method: round-robin
    monitor_type: m_of_n
    quorum: 1
    monitors:
        - http
```

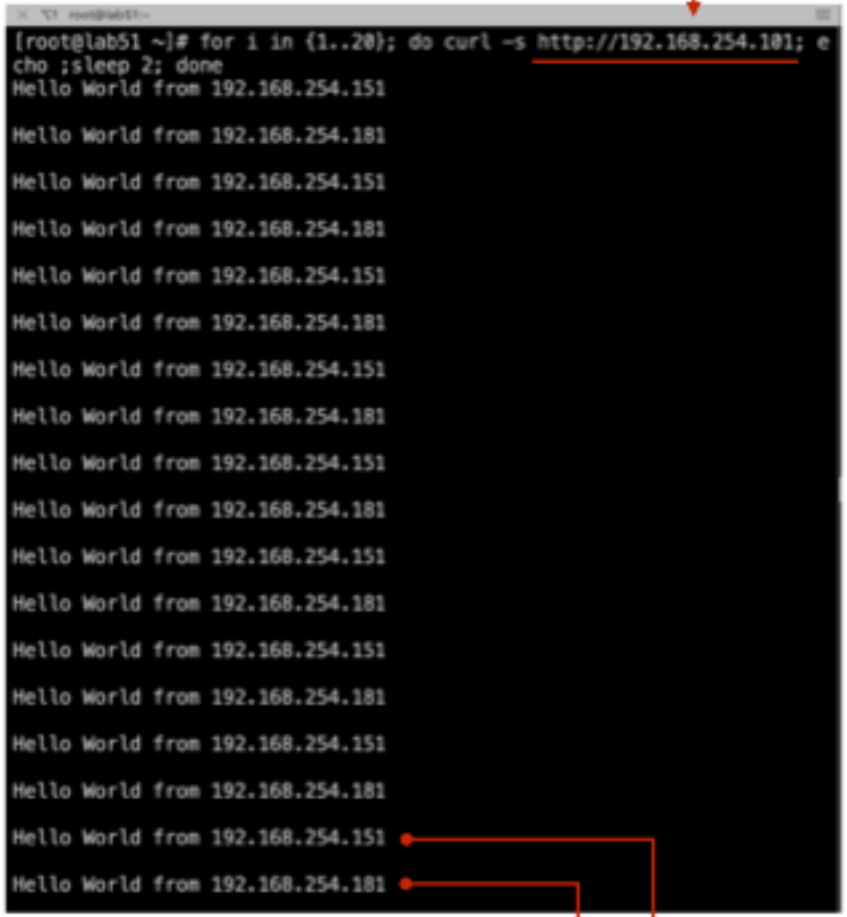


```
- name: Add pool member
  bigip_pool_member:
    server: "{{ LTM }}"
    user: "{{ LTM_USER }}"
    password: "{{ LTM_PASSWD }}"
    state: present
    pool: "{{ POOL_NAME }}"
    partition: Common
    validate_certs: no
    host: "{{ item }}"
    port: 80
  with_items:
    - 192.168.254.<x+100>
    - 192.168.254.<x+130>
```



```
- name: Create Virtual Server
  bigip_virtual_server:
      server: "{{ LTM }}"
      user: "{{ LTM_USER }}”
      password: "{{ LTM_PASSWD }}"
      state: present
      validate_certs: no
      partition: Common
      name: "{{ VIR_SERVER_NAME }}"
      destination: "{{ VIR_IP }}"
      port: 80
      pool: "{{ POOL_NAME }}"
      snat: Automap
```





```
[root@lab51 ~]# for i in {1..20}; do curl -s http://192.168.254.101; e
cho ;sleep 2; done
Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181

Hello World from 192.168.254.151

Hello World from 192.168.254.181
```

thank
you