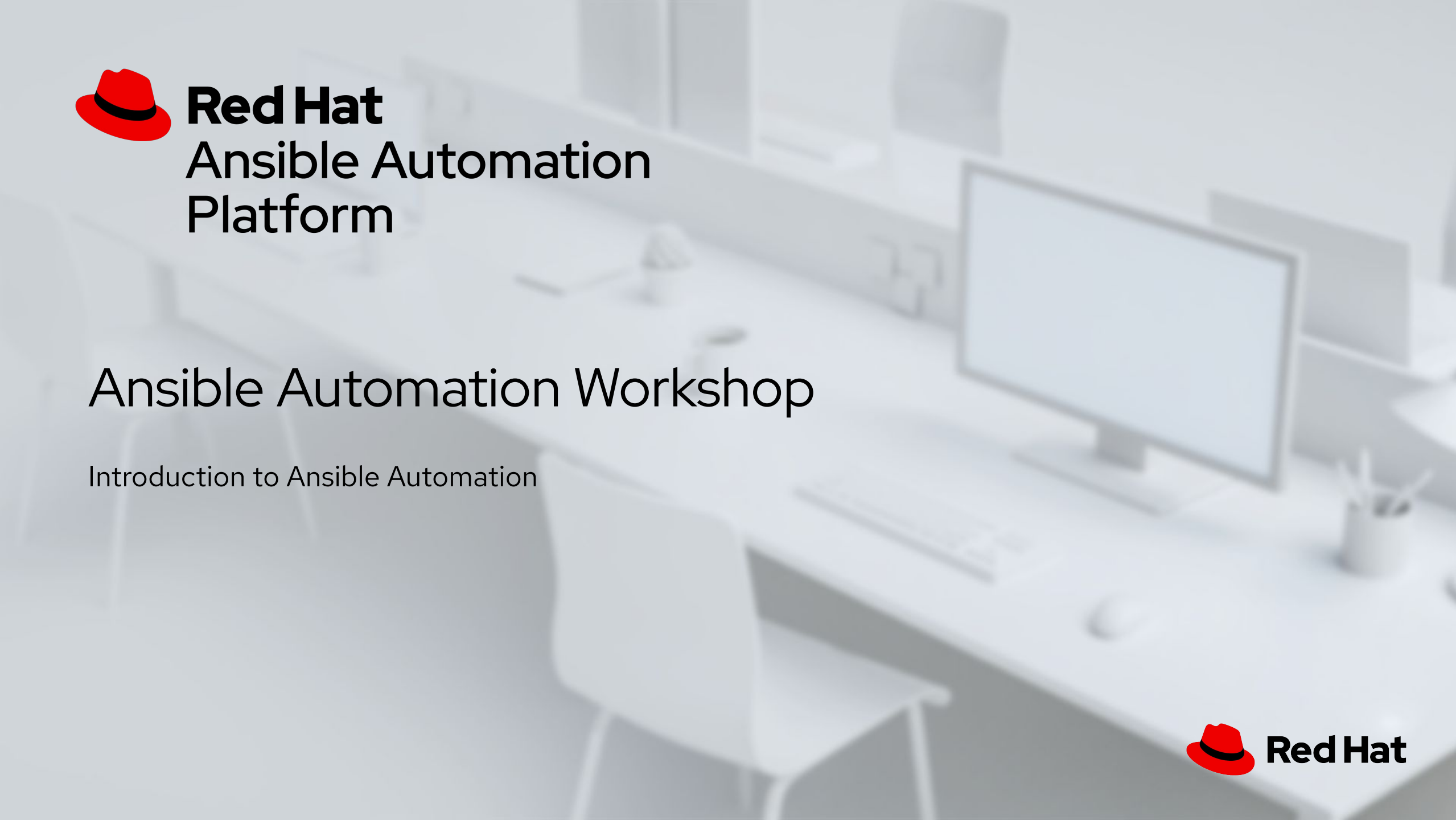


Red Hat
Ansible Automation
Platform

Ansible Automation Workshop

Introduction to Ansible Automation

Red Hat

Agenda

1400-1430	30 mins	**Welcome & Ansible Automation overview presentation**
1430-1440	10 mins	**Lab setup & introducing the lab controls**
1440-1525	45 mins	**Hands-on 1+2** Overview of public cloud provisioning Converting shell commands into Ansible commands
1525-1610	45 mins	**Hands-on 3+4** Retrieving information from hosts Deploying applications at scale
1610-1655	45 mins	**Hands-on 5+6** Self-service IT via surveys Automation Workflows
1655-1700	5 mins	**Wrap-up: hands-on workshop review**

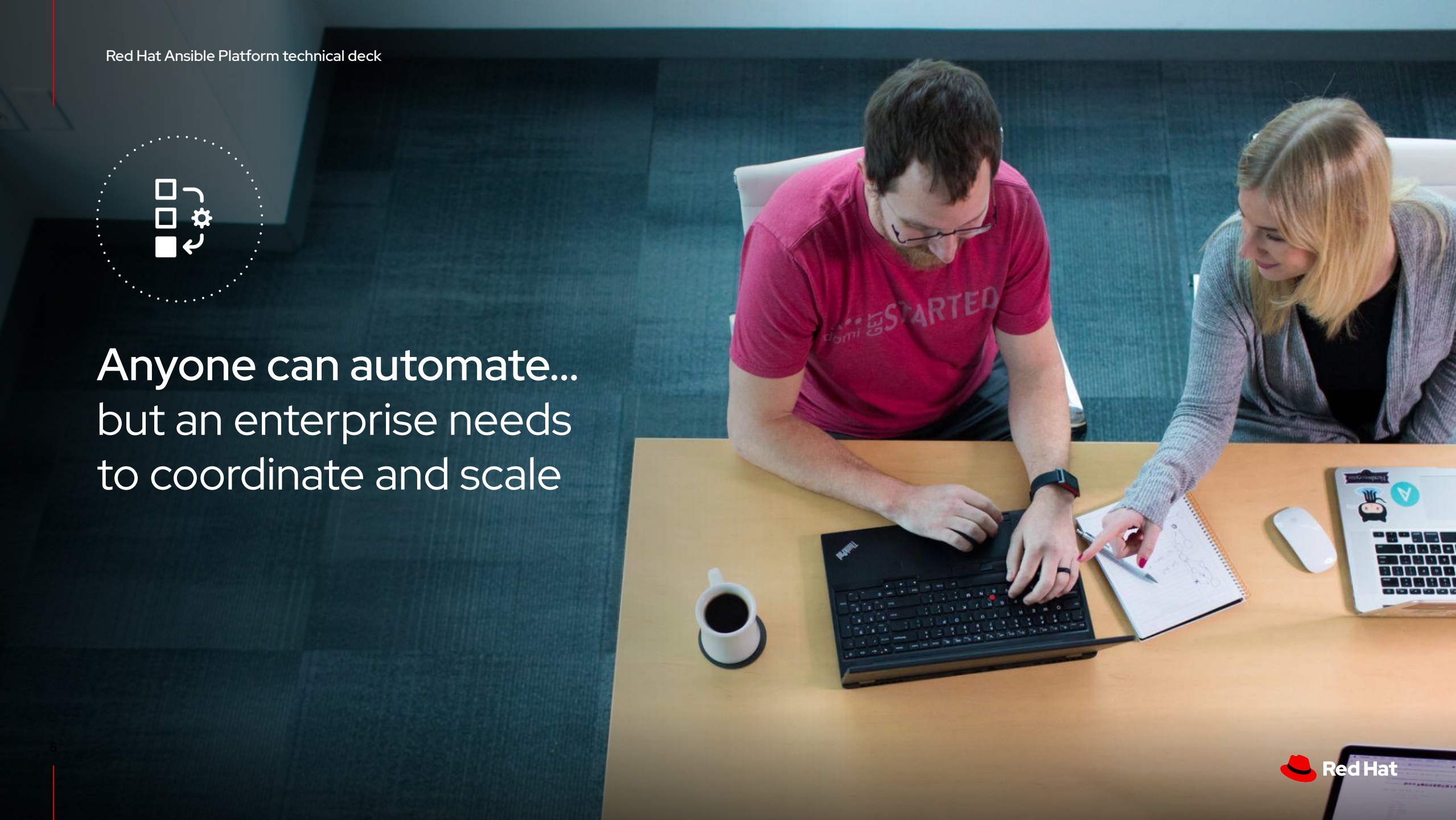
What you will learn

- Overview of public cloud provisioning
- Converting shell commands into Ansible commands
- Retrieving information from hosts
- Deploying applications at scale
- Self-service IT via surveys
- Automation Workflows

Introduction

Topics Covered:

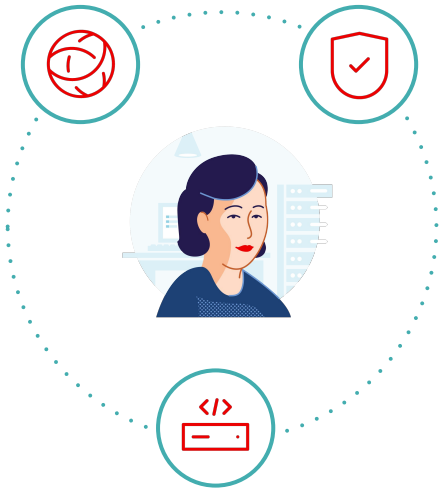
- What is the Ansible Automation Platform?
- What can it do?



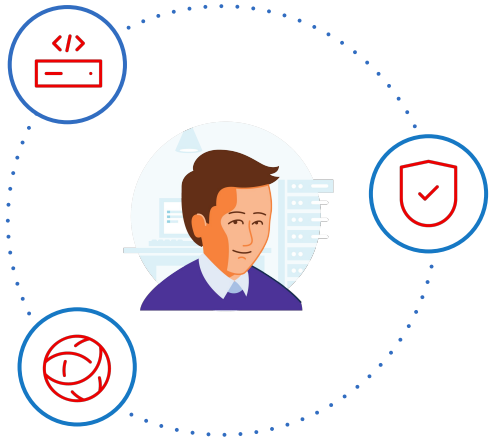
Anyone can automate... but an enterprise needs to coordinate and scale

Many organizations share the same challenge

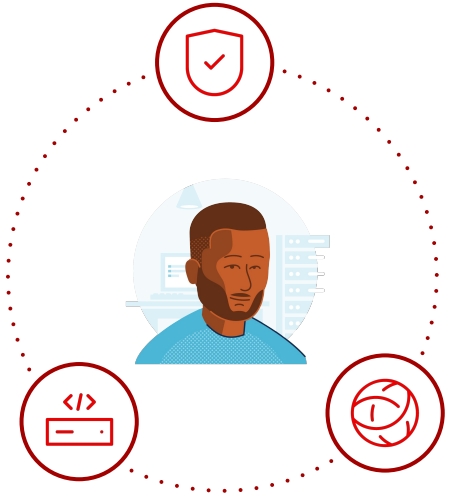
Too many unintegrated, domain-specific tools



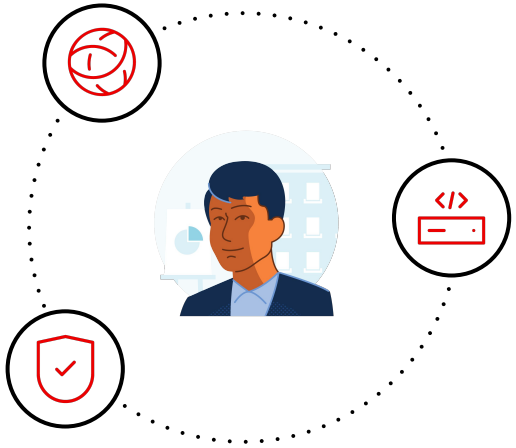
Network ops



SecOps



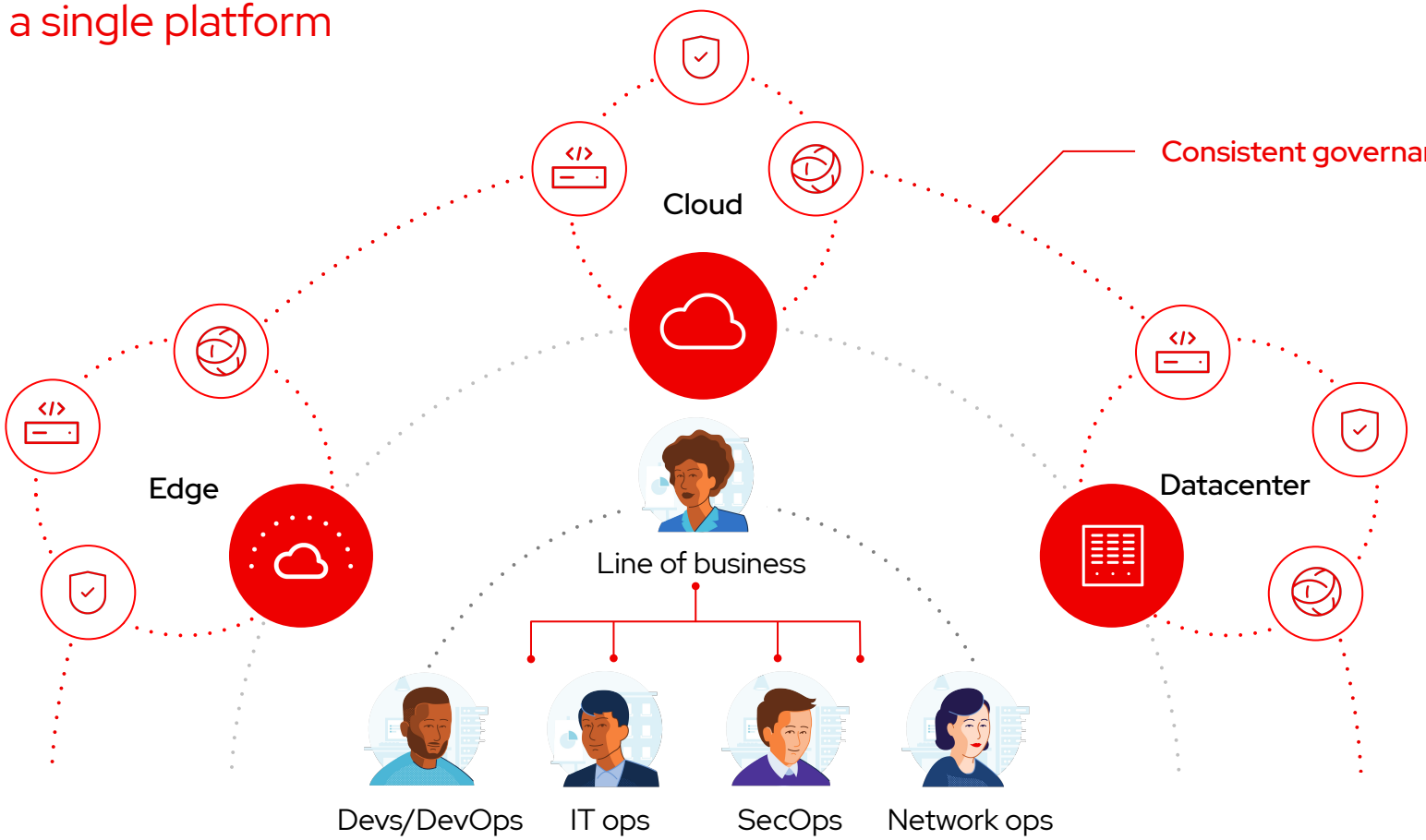
Devs/DevOps



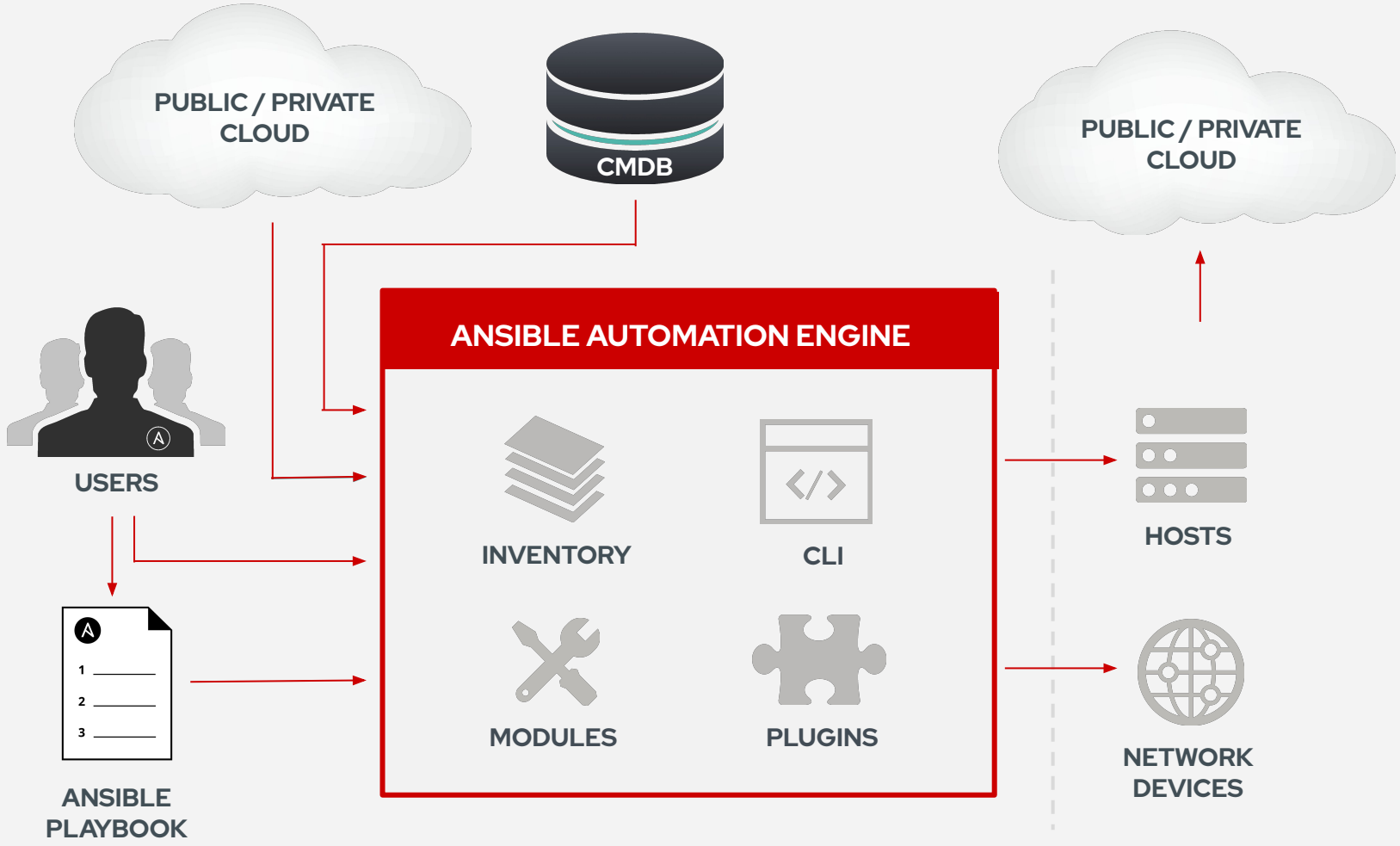
IT ops

Break down silos

Different teams a single platform

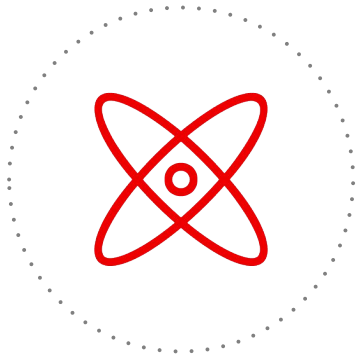


Red Hat


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
ANSIBLE AUTOMATION ENGINE
USERS
INVENTORY
CLI
HOSTS
MODULES
PLUGINS
NETWORK DEVICES
1
2
3
ANSIBLE PLAYBOOK

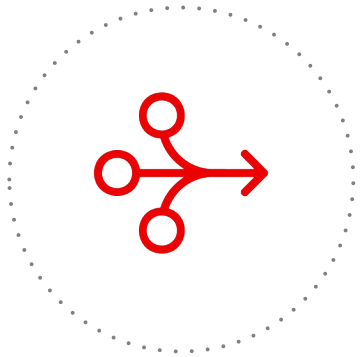
Red Hat

Why the Ansible Automation Platform?



Powerful

Orchestrate complex processes at enterprise scale.

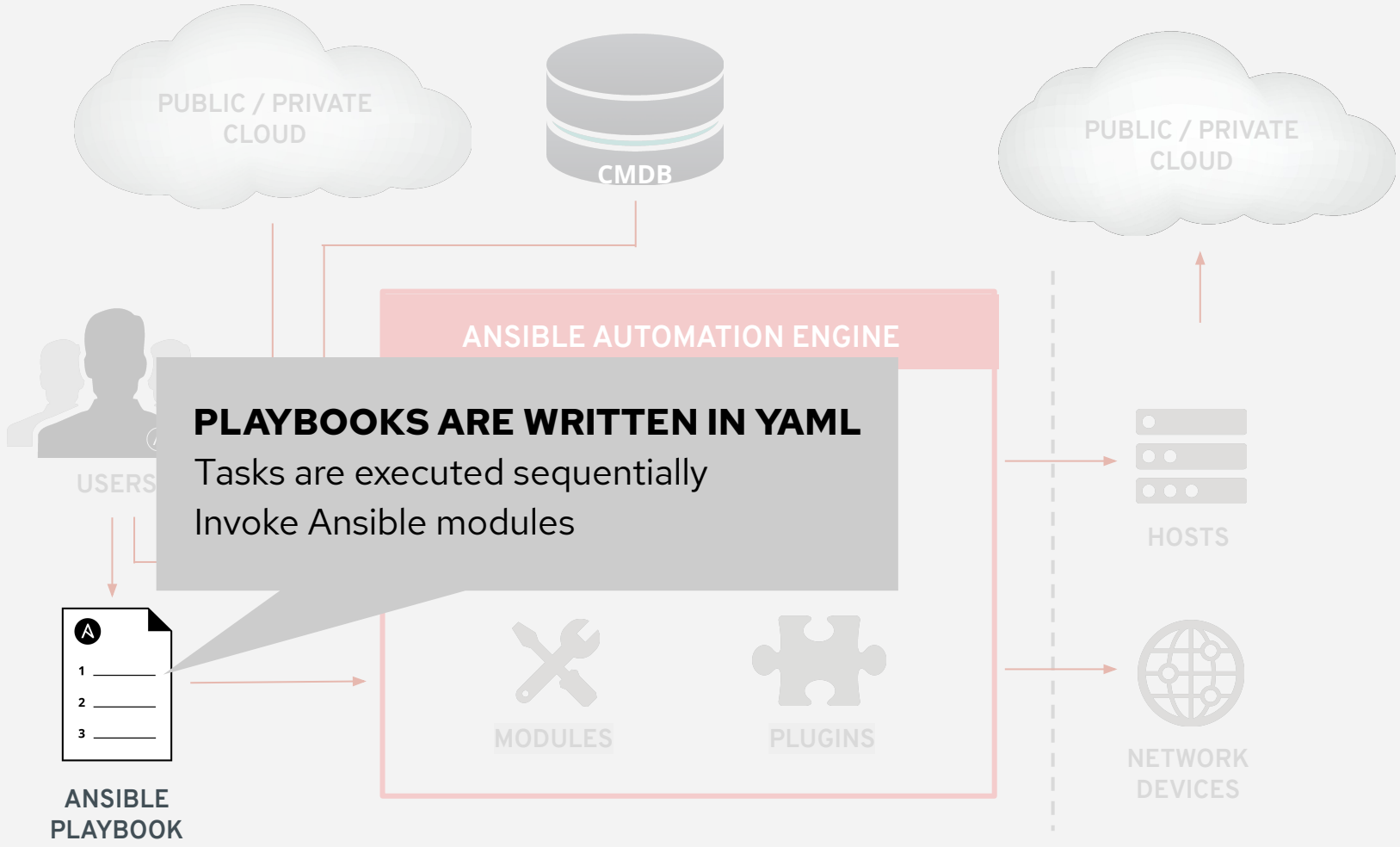


Simple

Simplify automation creation and management across multiple domains.

Agentless

Easily integrate with hybrid environments.


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
ANSIBLE AUTOMATION ENGINE
USERS
PLAYBOOKS ARE WRITTEN IN YAML
Tasks are executed sequentially
Invoke Ansible modules
HOSTS
1
2
3
MODULES
PLUGINS
NETWORK DEVICES
ANSIBLE PLAYBOOK

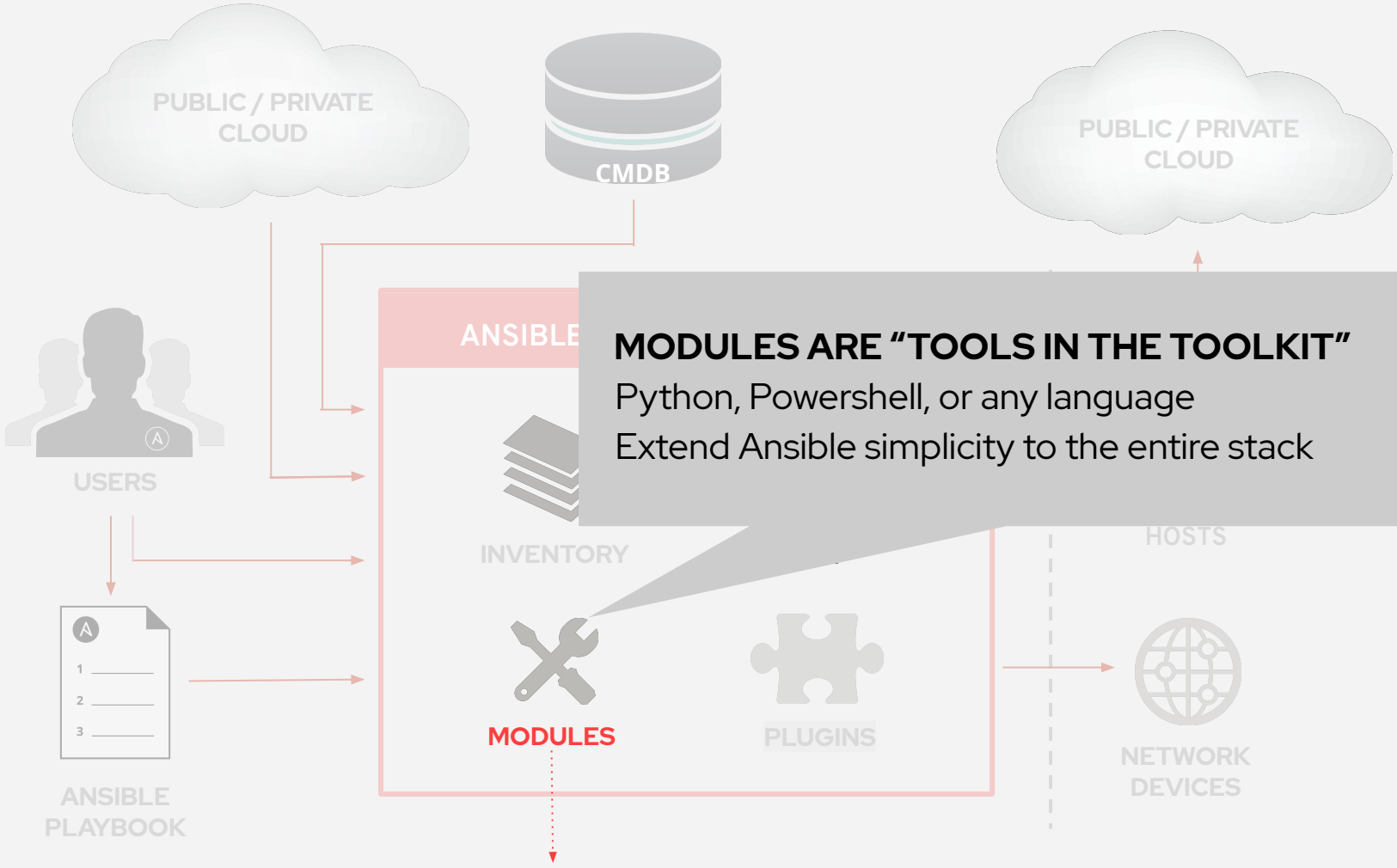
Red Hat

```
---
- name: install and start apache
  hosts: web
  become: yes

  tasks:
    - name: httpd package is present
      yum:
        name: httpd
        state: latest

    - name: latest index.html file is present
      file:
        src: files/index.html
        dest: /var/www/html/

    - name: httpd is started
      service:
        name: httpd
        state: started
```



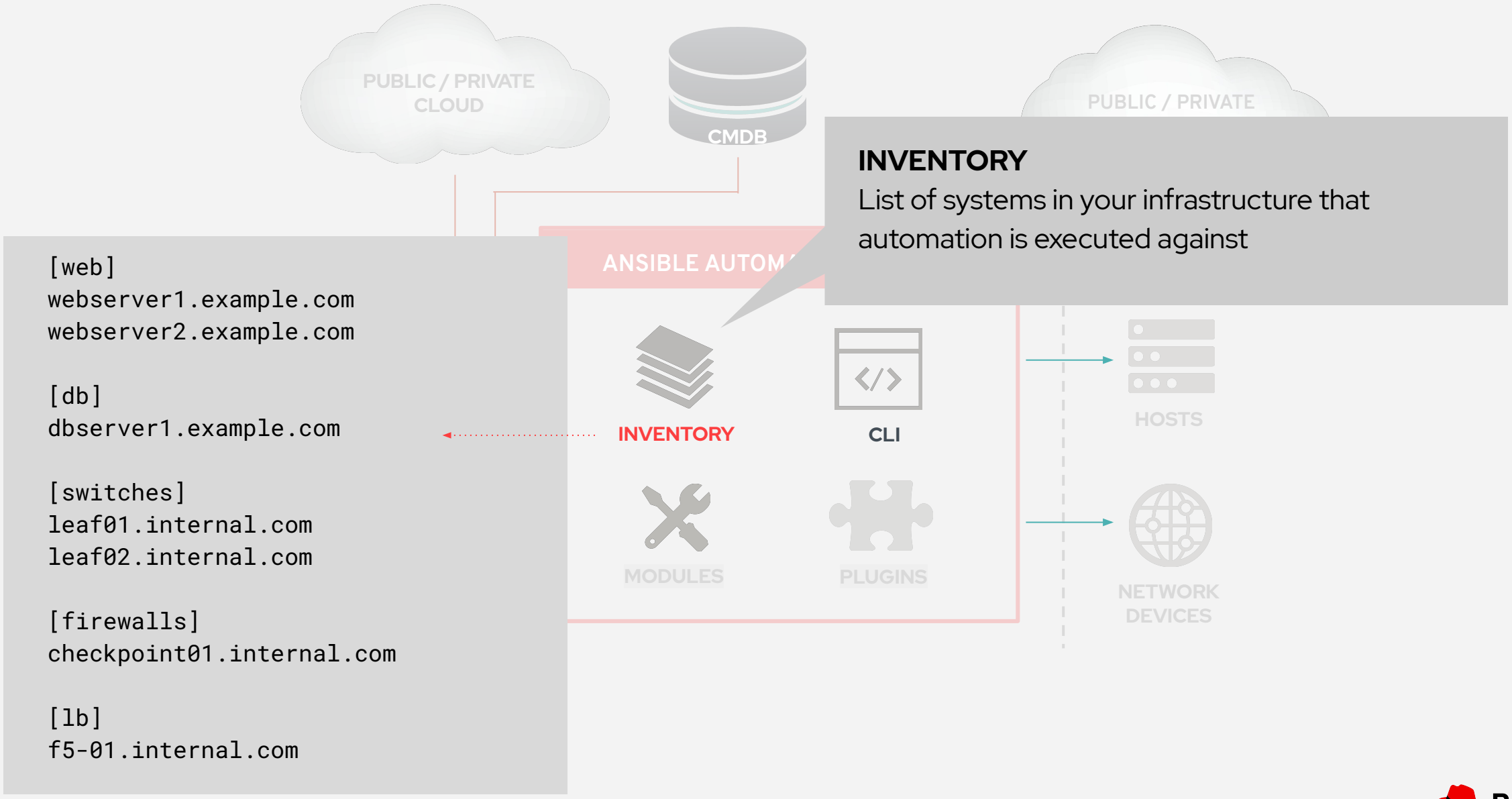
MODULES ARE "TOOLS IN THE TOOLKIT"

Python, Powershell, or any language

Extend Ansible simplicity to the entire stack

```
- name: latest index.html file is present
  template:
    src: files/index.j2
    dest: /var/www/html/
```

Red Hat


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE
INVENTORY
List of systems in your infrastructure that automation is executed against
[web]
webserver1.example.com
webserver2.example.com
[db]
dbserver1.example.com
[switches]
leaf01.internal.com
leaf02.internal.com
[firewalls]
checkpoint01.internal.com
[lb]
f5-01.internal.com
ANSIBLE AUTOM
INVENTORY
CLI
HOSTS
MODULES
PLUGINS
NETWORK DEVICES

Red Hat

Ansible Automates Technologies You Use

Time to Automate is Measured in Minutes

CLOUD

AWS
Azure
Digital Ocean
Google
OpenStack
Rackspace
+more

OPERATING SYSTEMS

RHEL and Linux
UNIX
Windows
+more

VIRT & CONTAINER

Docker
VMware
RHV
OpenStack
OpenShift
+more

STORAGE

NetApp
Red Hat Storage
Infinidat
+more

WINDOWS

ACLs
Files
Packages
IIS
Regedits
Shares
Services
Configs
Users
Domains
+more

NETWORK

Arista
Aruba
A10
Cumulus
Bigswitch
Cisco
Dell
Ericsson
F5
Juniper
OpenSwitch
Ruckus
VyOS
+more

DEVOPS

Jira
GitHub
Vagrant
Jenkins
Bamboo
Atlassian
Subversion
Slack
Hipchat
+more

MONITORING

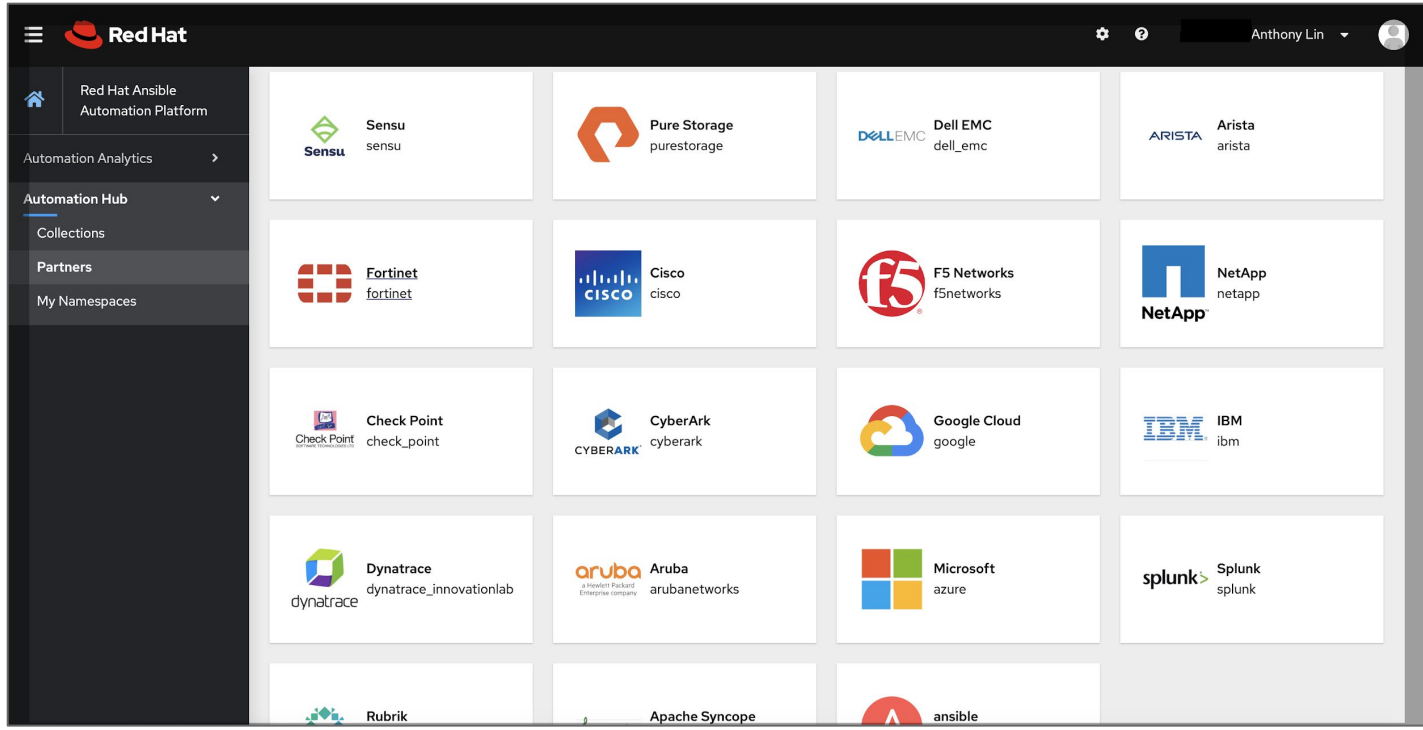
Dynatrace
Airbrake
BigPanda
Datadog
LogicMonitor
Nagios
New Relic
PagerDuty
Sensu
StackDriver
Zabbix
+more

Red Hat

Automation Hub

Discover, publish, and manage Collections

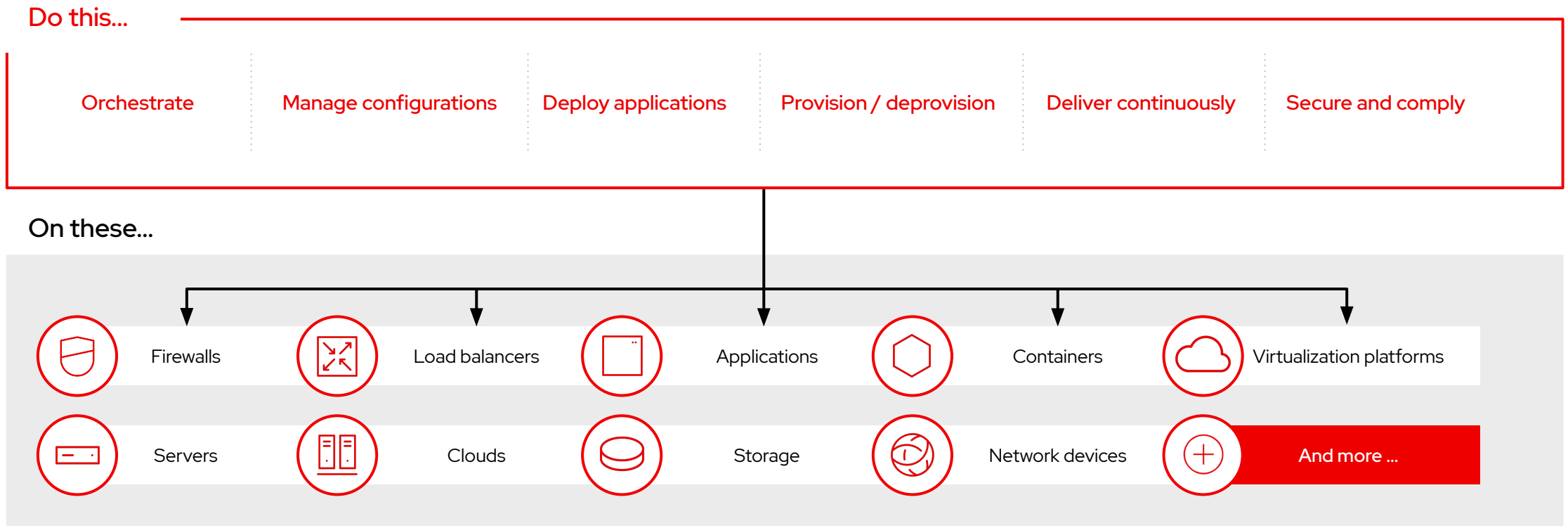
- Quickly discover available Red Hat and certified content through Collections.
- Manage and test your organization's view of available content.*
- Manage your locally available automation via on-premise.*



Red Hat

Automate the deployment and management of automation

Your entire IT footprint



Red Hat

RED HAT®
ANSIBLE®
Tower

ACCESS CONTROL
Role-based access control
& LDAP integration

INVENTORY MANAGEMENT
Graphically manage your internal &
cloud resources

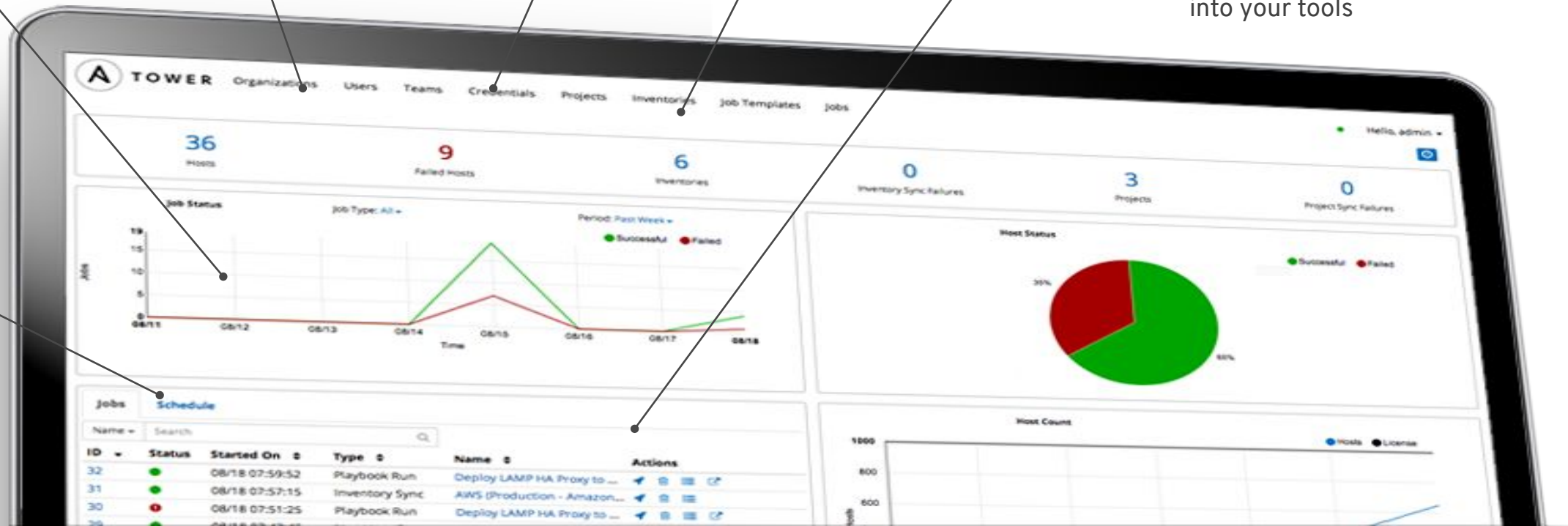
DELEGATION OF CREDENTIALS
Delegate credentials without
giving away secrets

PUSH-BUTTON LAUNCH
Launch automation jobs with a button

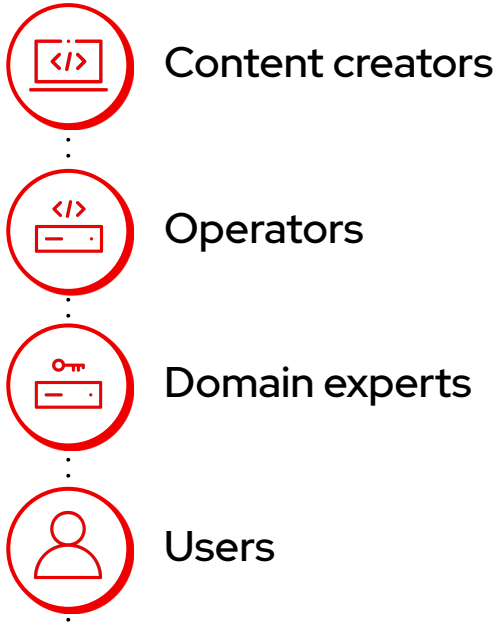
AUDITING
See a full Ansible job
history with drill-in details

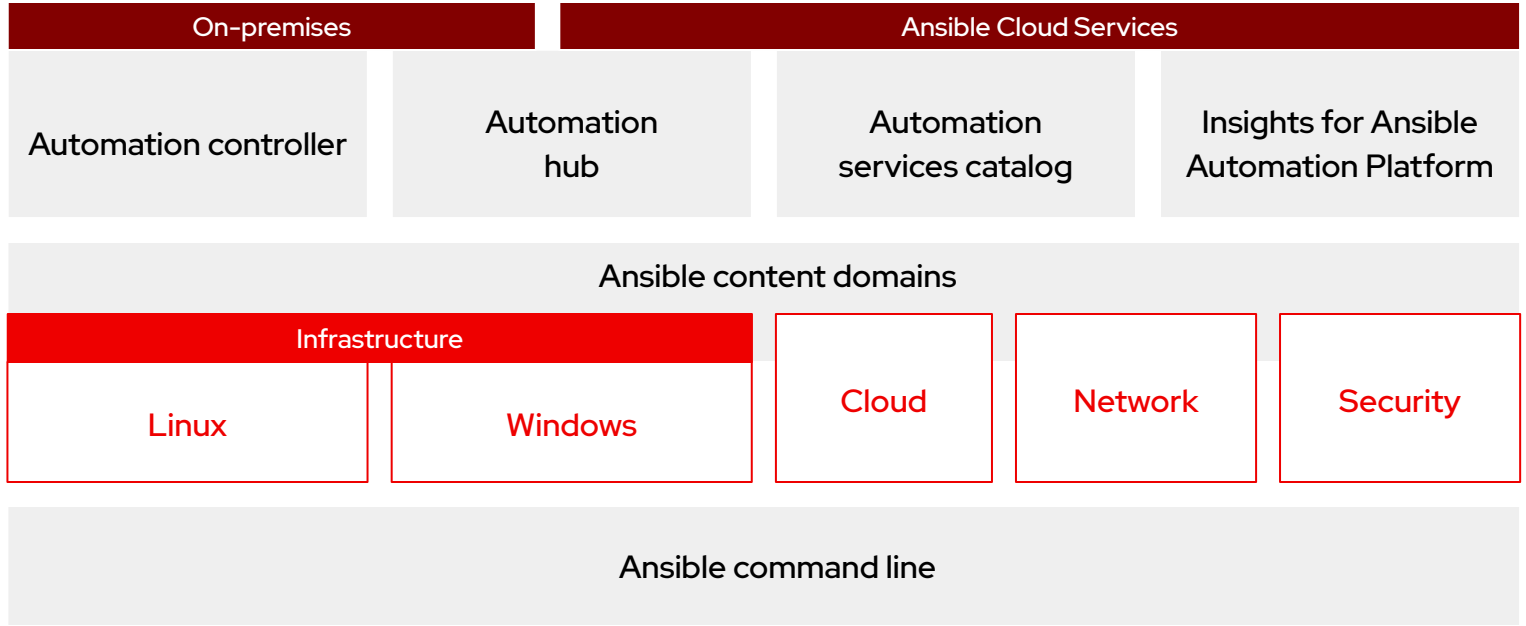
API & CLI
Documented RESTful API and
Tower CLI to integrate Tower
into your tools

SCHEDULING
Schedule
automation jobs
(great for
periodic
remediation)



Red Hat Ansible Automation Platform

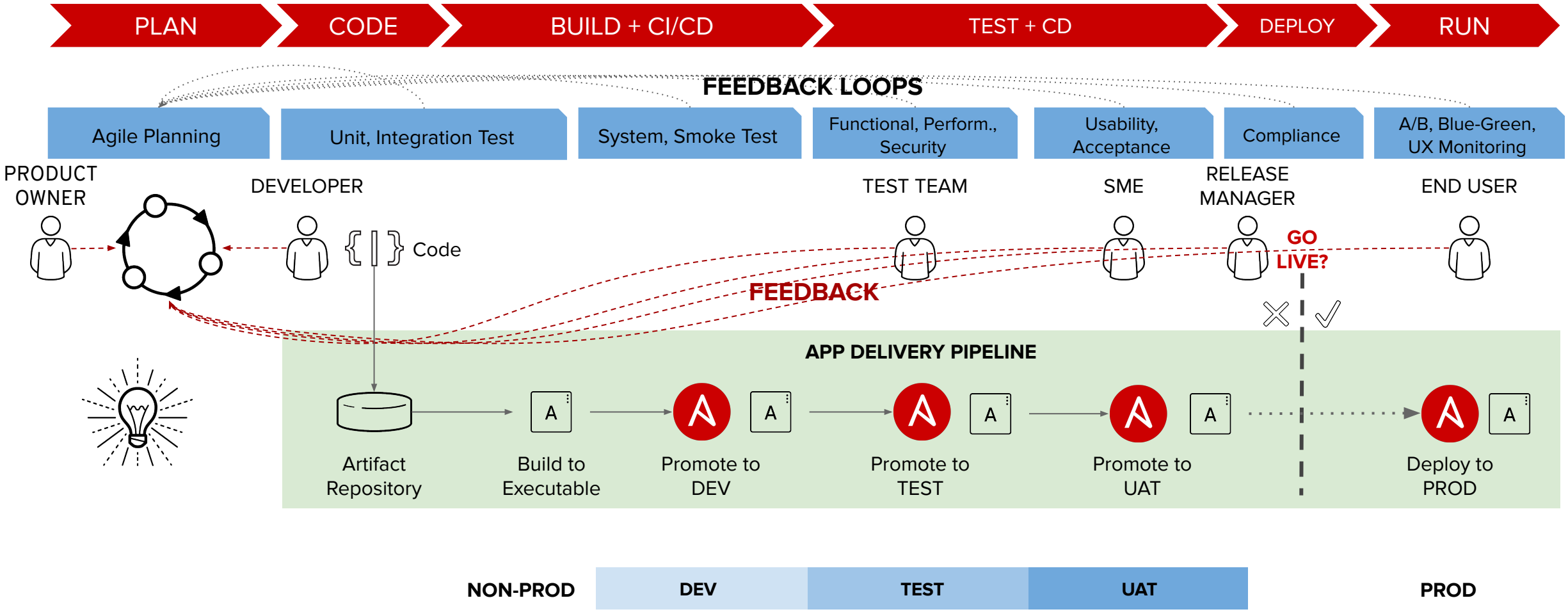




Fueled by an open source community

Red Hat

DevOps Pipeline with Ansible



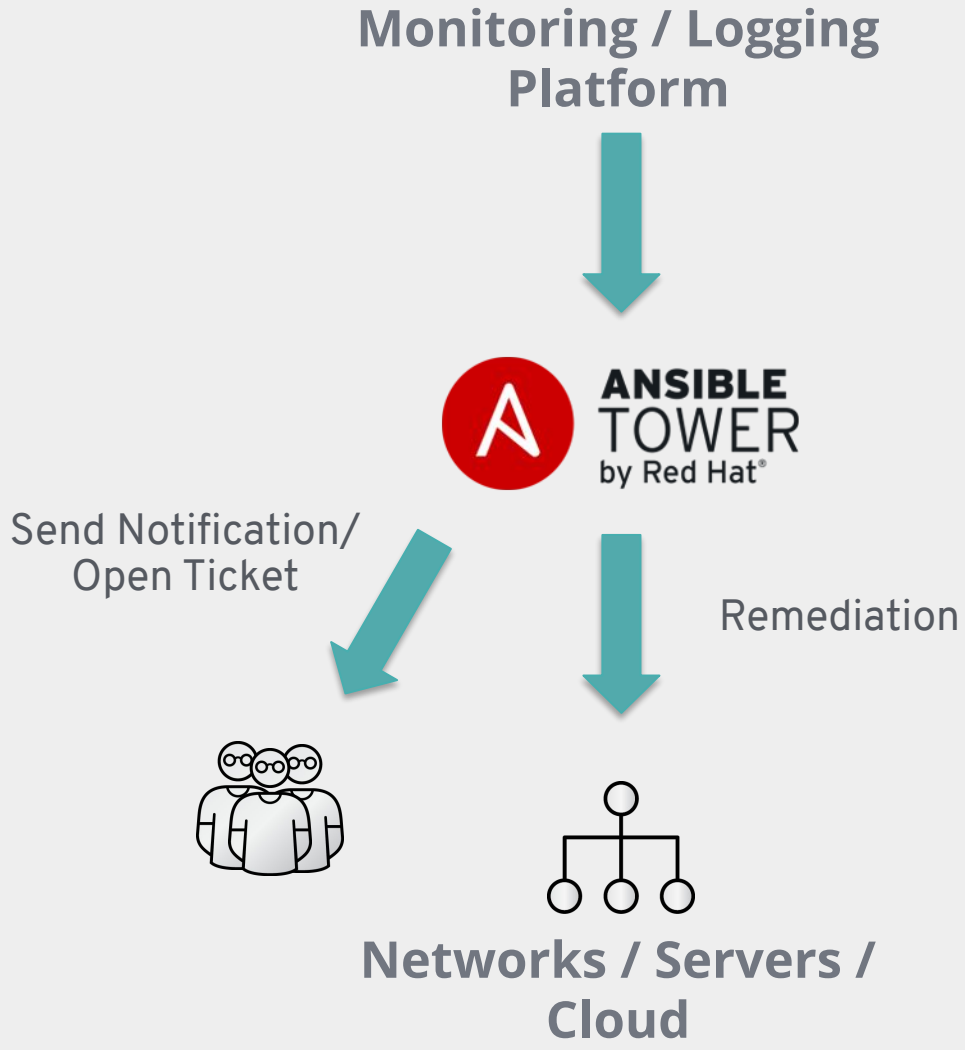
Red Hat

Remediate Automation

1. Monitoring/Logging Platform detects security issues and calls the Ansible Tower API

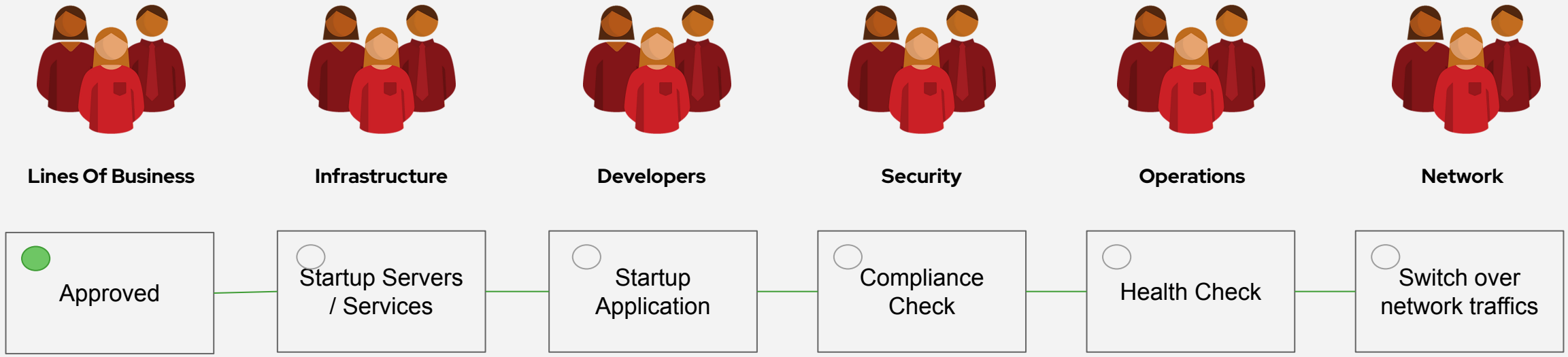
2. Ansible Tower runs a playbook to automate remediation in servers / equipments

3. Ansible Tower runs a playbook to open a support ticket and/or notify security managers / system administrators



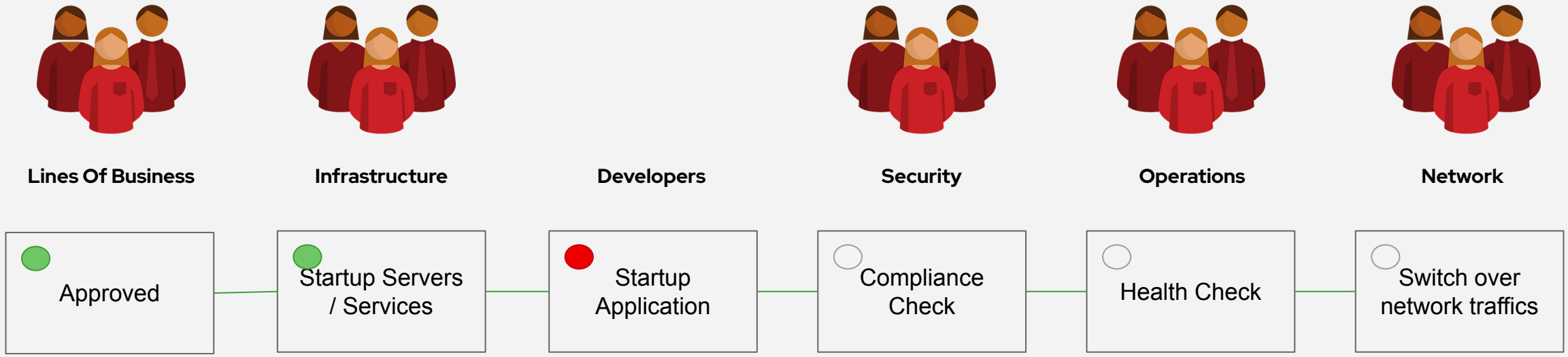
Red Hat

Teams are implementing Disaster Recovery...



Some planning tasks (e.g. Disaster Recovery Drill) usually required different teams to work together.

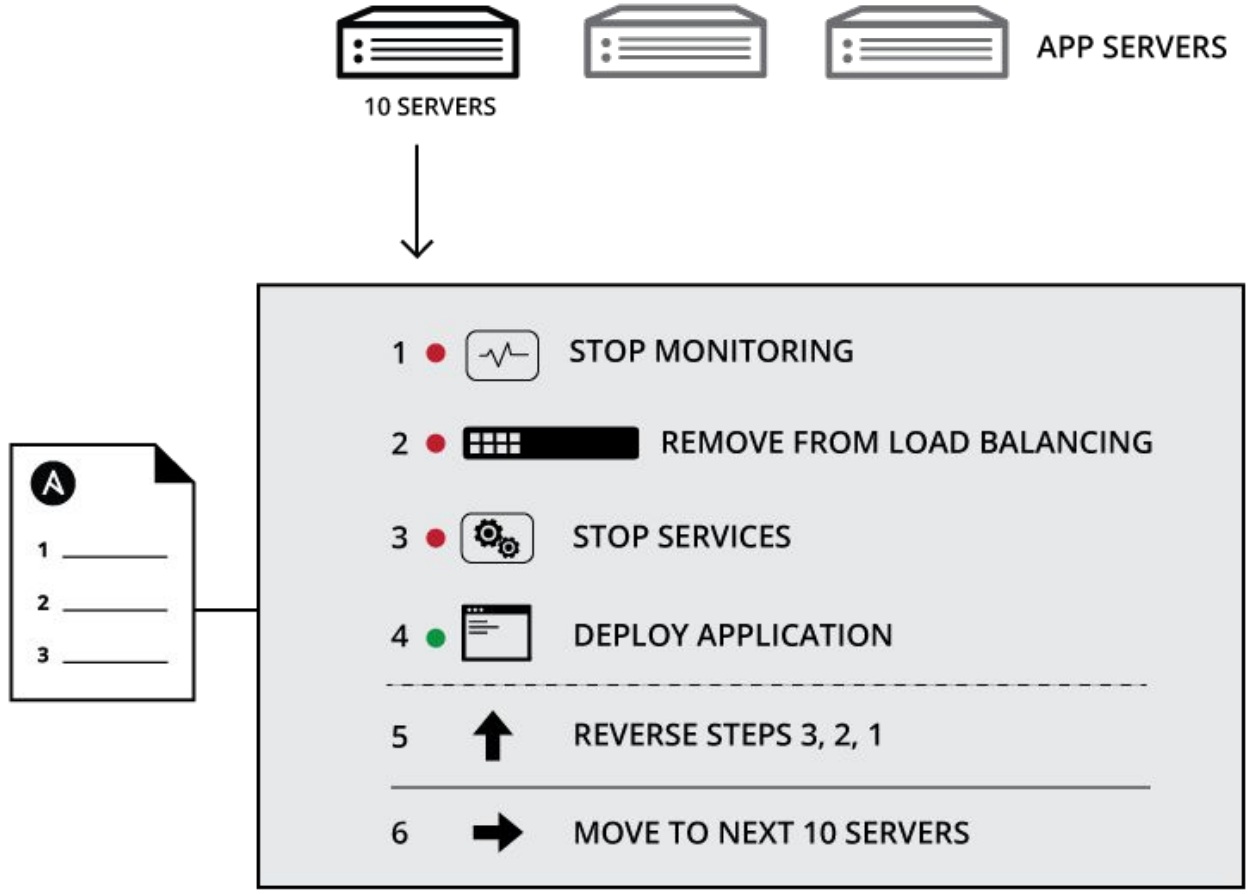
Teams are implementing Disaster Recovery...



Some planning tasks (e.g. Disaster Recovery Drill) can't be performed as it usually required different teams to work together.

Rolling Upgrade / Patching

Your applications and systems **are more than just collections of configurations.** They're a finely tuned and **ordered list** of tasks and processes that result in **your working application.**



Red Hat

Security Compliance – C2S

Ansible remediation role for profile C2S
Profile Title: C2S for Red Hat Enterprise Linux 7

```
- hosts: all
  roles:
    - role: RedHatOfficial.rhel7_c2s
      when:
        - ansible_os_family == 'RedHat'
        - ansible_distribution_major_version | version_compare('7', '=')
```

- Huge manual works for checking / remediation of security compliance settings

After Ansible Automation:

- Shorten the time for manual works

RHEL 7 C2S

https://rhelblog.redhat.com/2018/06/19/automating-security-compliance-with-ease/

Red Hat
Ansible Automation
Platform

Exercise 1

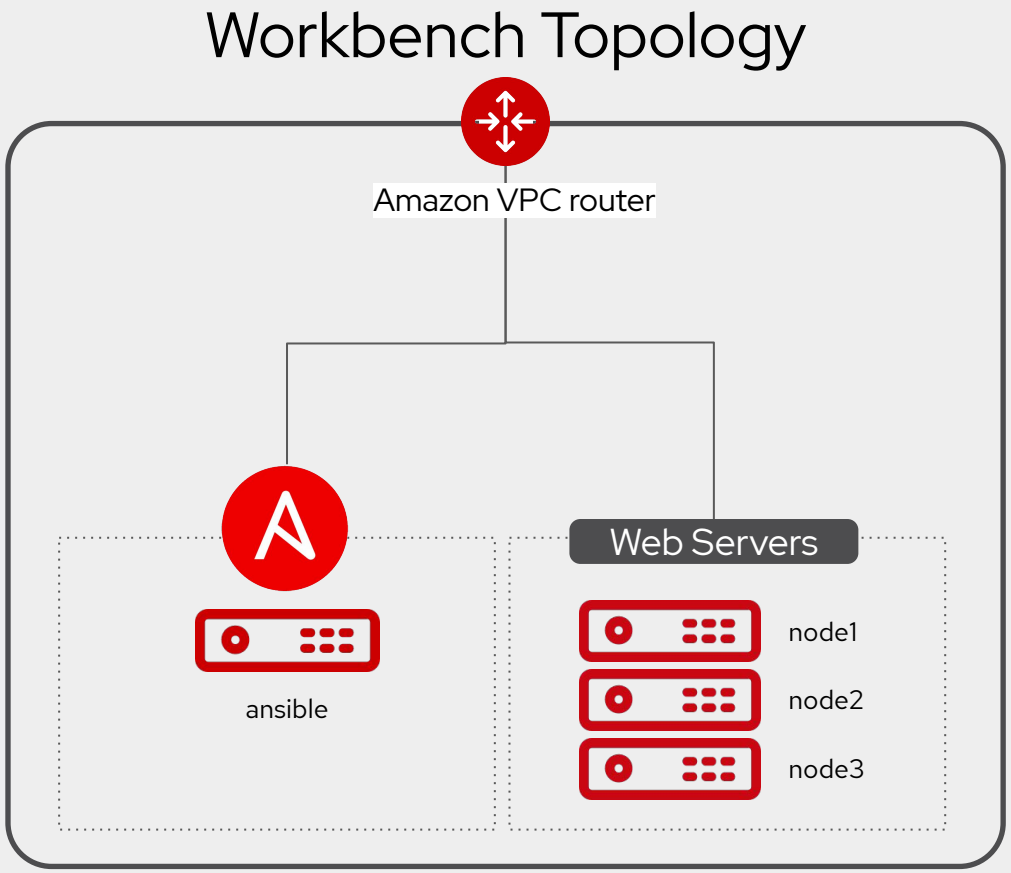
Topics Covered:

- Understanding the Ansible Infrastructure
- Check the prerequisites

Red Hat

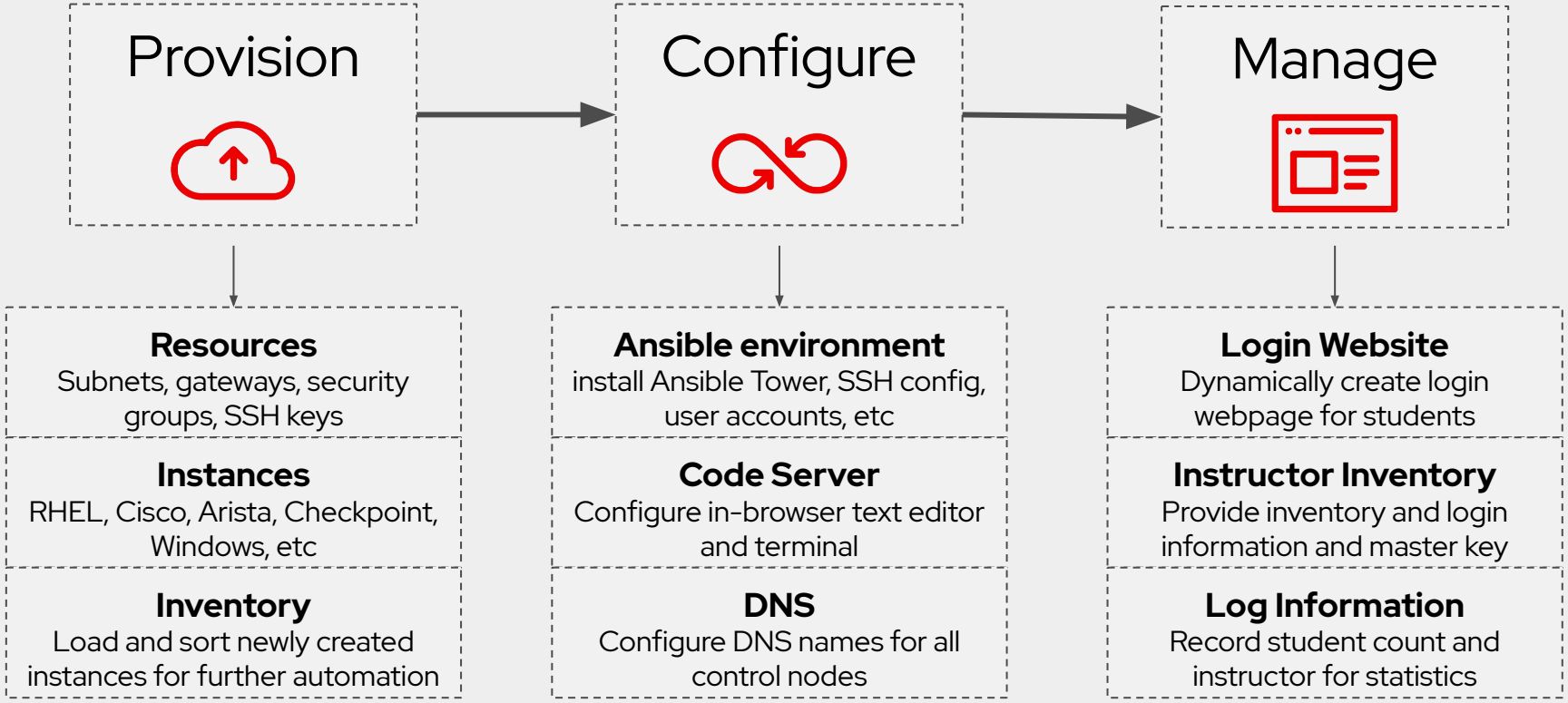
The lab environment today

- **Drink our own champagne**.
  Provisioned by, configured by, and managed by Red Hat Ansible Automation Platform.
  https://github.com/ansible/workshops
- **Learn with the real thing**
  Every student will have their own fully licensed Red Hat Ansible Tower control node. No emulators or simulators here.
- **Red Hat Enterprise Linux**
  All four nodes are enterprise Linux, showcasing real life use-cases to help spark ideas for what you can automate today.

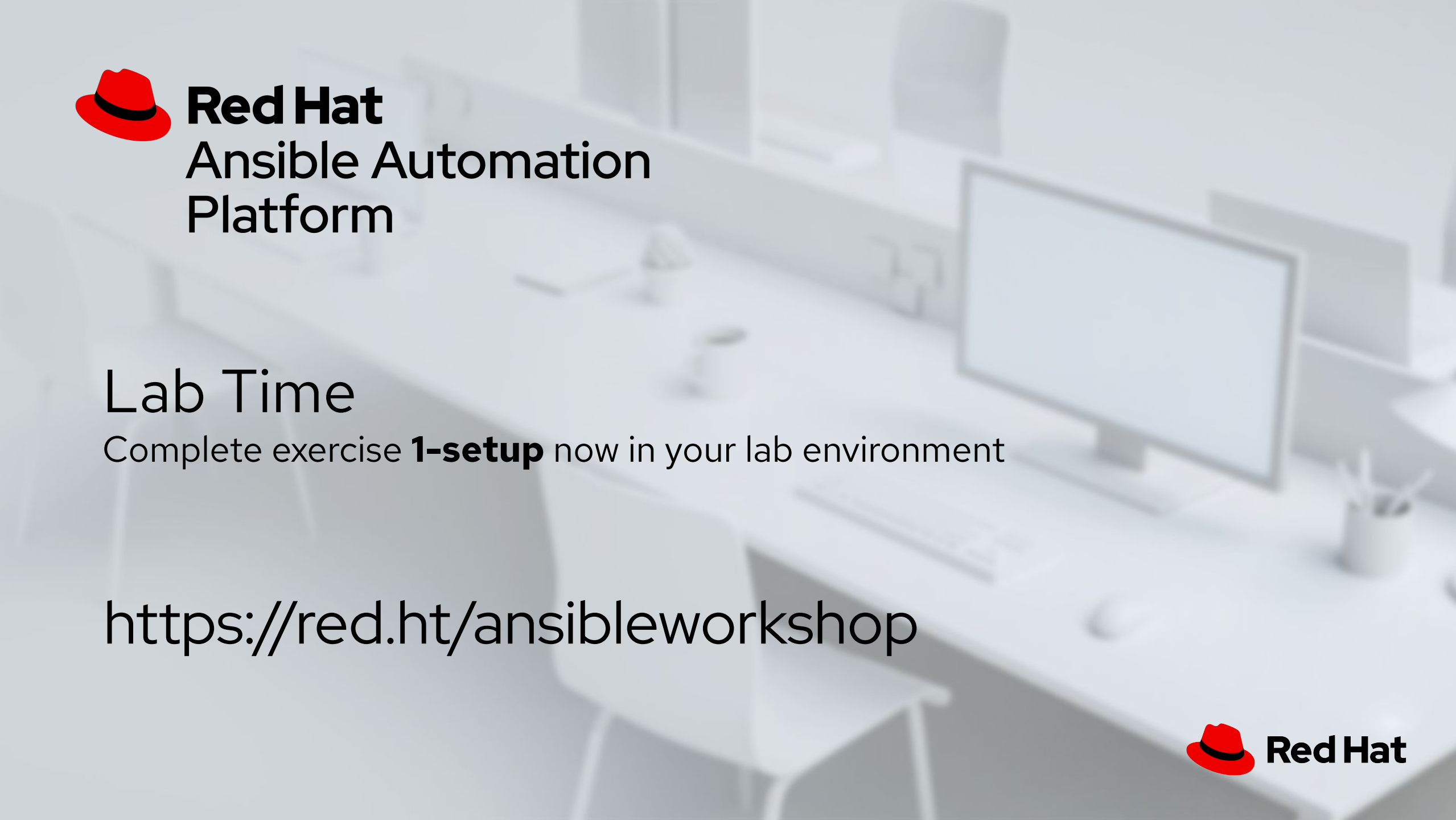


Red Hat

How does it work?



Red Hat



Red Hat
Ansible Automation
Platform

Lab Time

Complete exercise **1-setup** now in your lab environment

https://red.ht/ansibleworkshop

Red Hat

Red Hat
Ansible Automation
Platform

Exercise 2

Topics Covered:

- Ansible inventories
- Main Ansible config file
- Modules and ad-hoc commands
- Example: Bash vs. Ansible

Red Hat

Inventory

- Ansible works against multiple systems in an inventory
- Inventory is usually file based
- Can have multiple groups
- Can have variables for each group or even host

Understanding Inventory - Basic

```
node1
node2
node3
ansible
10.20.30.40
```

Understanding Inventory - Basic

```
[web]
node1 ansible_host=3.22.77.141
node2 ansible_host=3.15.193.71
node3 ansible_host=3.15.1.72

[control]
ansible ansible_host=18.217.162.148
```

Understanding Inventory - Variables

```
[all:vars]
ansible_user=student1
ansible_ssh_pass=ansible1234
ansible_port=22

[web]
node1 ansible_host=3.22.77.141
node2 ansible_host=3.15.193.71
node3 ansible_host=3.15.1.72

[control]
ansible ansible_host=18.217.162.148
```

First Ad-Hoc Command: ping

- Single Ansible command to perform a task quickly directly on command line
- Most basic operation that can be performed
- Here: an example Ansible ping - not to be confused with ICMP

```
$ ansible all -m ping
```

Ad-Hoc Commands `ping`

```
# Check connections (submarine ping, not ICMP)
[user@ansible] $ ansible all -m ping

node1 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python":
"/usr/bin/python"
    },
    "changed": false,
    "ping": "pong"
}
```

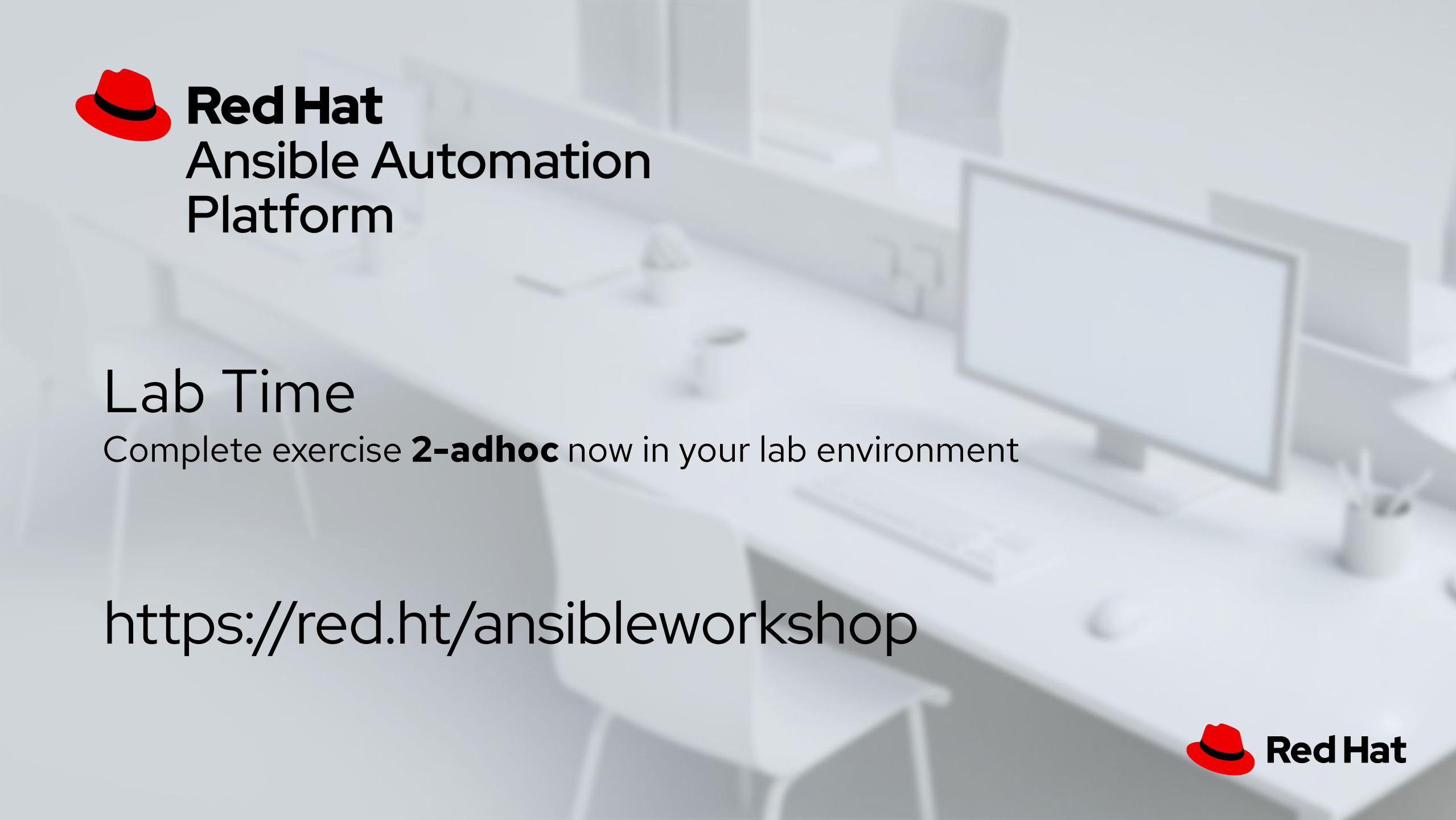
Bash vs. Ansible

```
echo Running mssql-conf setup...
sudo
MSSQL_SA_PASSWORD=$MSSQL_SA_PASSWORD \
   MSSQL_PID=$MSSQL_PID \
   /opt/mssql/bin/mssql-conf -n setup accept-eula


echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >>
~/.bash_profile
echo 'export PATH="$PATH:/opt/mssql-tools/bin"' >>
~/.bashrc
source ~/.bashrc
```

```
- name: Run mssql-conf setup
  command: /opt/mssql/bin/mssql-conf -n setup
accept-eula
  environment:
    - MSSQL_SA_PASSWORD: "{{ MSSQL_SA_PASSWORD }}"
    - MSSQL_PID: "{{ MSSQL_PID }}"
  when: install is changed

- name: Add mssql-tools to $PATH
  lineinfile:
    path: "{{ item }}"
    line: export PATH="$PATH:/opt/mssql-tools/bin"
  loop:
    - ~/.bash_profile
    - ~/.bashrc
```


Red Hat
Ansible Automation
Platform
Lab Time
Complete exercise **2-adhoc** now in your lab environment
https://red.ht/ansibleworkshop
Red Hat

Red Hat
Ansible Automation
Platform

Exercise 3

Topics Covered:

- Playbooks basics
- Running a playbook

Red Hat

An Ansible Playbook

A play

```
---
- name: install and start apache
  hosts: web
  become: yes

  tasks:
    - name: httpd package is present
      yum:
        name: httpd
        state: latest

    - name: latest index.html file is present
      template:
        src: files/index.html
        dest: /var/www/html/

    - name: httpd is started
      service:
        name: httpd
        state: started
```

An Ansible Playbook

A task

```
---
- name: install and start apache
  hosts: web
  become: yes

  tasks:
    - name: httpd package is present
      yum:
        name: httpd
        state: latest

    - name: latest index.html file is present
      template:
        src: files/index.html
        dest: /var/www/html/

    - name: httpd is started
      service:
        name: httpd
        state: started
```

An Ansible Playbook

module

```
---
- name: install and start apache
  hosts: web
  become: yes

  tasks:
    - name: httpd package is present
      yum:
        name: httpd
        state: latest

    - name: latest index.html file is present
      template:
        src: files/index.html
        dest: /var/www/html/

    - name: httpd is started
      service:
        name: httpd
        state: started
```

Running an Ansible Playbook:

The most important colors of Ansible

```
A task executed as expected, no change was made.

A task executed as expected, making a change

A task failed to execute successfully
```

Running an Ansible Playbook

```
[user@ansible] $ ansible-playbook apache.yml

PLAY [webservers] ***************************************************************

TASK [Gathering Facts] **********************************************************
ok: [web2]
ok: [web1]
ok: [web3]

TASK [Ensure httpd package is present] ******************************************
changed: [web2]
changed: [web1]
changed: [web3]

TASK [Ensure latest index.html file is present] *********************************
changed: [web2]
changed: [web1]
changed: [web3]

TASK [Restart httpd] ************************************************************
changed: [web2]
changed: [web1]
changed: [web3]

PLAY RECAP **********************************************************************
web2                       : ok=1    changed=3 unreachable=0   failed=0
web1                       : ok=1    changed=3 unreachable=0   failed=0
web3                       : ok=1    changed=3 unreachable=0   failed=0
```

Red Hat
Ansible Automation
Platform

Lab Time

Complete exercise **3-playbooks** now in your lab environment

https://red.ht/ansibleworkshop

Red Hat

Red Hat
Ansible Automation
Platform

Exercise 4

Topics Covered:

- Working with variables
- What are facts?

Red Hat

An Ansible Playbook Variable Example

```
---
- name: variable playbook test
  hosts: localhost

  vars:
    var_one: awesome
    var_two: ansible is
    var_three: "{{ var_two }} {{ var_one }}"

  tasks:

    - name: print out var_three
      debug:
        msg: "{{var_three}}"
```

An Ansible Playbook Variable Example

```
---
- name: variable playbook test
  hosts: localhost

  vars:
    var_one: awesome
    var_two: ansible is
    var_three: "{{ var_two }} {{ var_one }}"

  tasks:

    - name: print out var_three
      debug:
        msg: "{{var_three}}"
```

ansible is awesome

Red Hat

Facts

- Structured data in the form of Ansible variables
- Information is capture from the host
- Ad-hoc command **setup** will show facts

```
    "ansible_facts": {
        "ansible_default_ipv4": {
            "address": "10.41.17.37",
            "macaddress": "00:69:08:3b:a9:16",
            "interface": "eth0",
...
```

Ansible Variables and Facts

```
---
- name: Output facts within a playbook
  hosts: all

  tasks:

    - name: Prints Ansible facts
      debug:
        msg: "The default IPv4 address of {{ ansible_fqdn }}
              is {{ ansible_default_ipv4.address }}"
```

```
TASK [Prints Ansible facts] ****************************************************
ok: [node3] =>
  msg: The default IPv4 address of node3 is 172.16.63.104
ok: [node1] =>
  msg: The default IPv4 address of node1 is 172.16.178.80
ok: [node2] =>
  msg: The default IPv4 address of node2 is 172.16.166.120
ok: [ansible] =>
  msg: The default IPv4 address of student1.sean-may4.rhdemo.io is 172.16.86.242
```

Red Hat

Ansible Inventory - Managing Variables In Files

```
$ tree ansible-files/

├── deploy_index_html.yml
├── files
│   ├── dev_web.html
│   └── prod_web.html
├── group_vars
│   └── web.yml
└── host_vars
    └── node2.yml
```

Ansible Inventory - Managing Variables In Files

```
├──
deploy_index_html.yml
├── files
│   ├── dev_web.html
│   └── prod_web.html
├── group_vars
│   └── web.yml
└── host_vars
    └── node2.yml
```

```
$ cat group_vars/web.yml
---
stage: dev
```

```
$ cat host_vars/node2.yml
---
stage: prod
```

```
- name: copy web.html
  copy:
    src: "{{ stage }}_web.html"
    dest: /var/www/html/index.html
```

Red Hat
Ansible Automation
Platform

Lab Time

Complete exercise **4-variables** now in your lab environment

https://red.ht/ansibleworkshop

Red Hat

Red Hat
Ansible Automation
Platform

Exercise 5

Topics Covered:

- Surveys

Red Hat

Surveys

Tower surveys allow you to configure how a job runs via a series of questions, making it simple to customize your jobs in a user-friendly way.

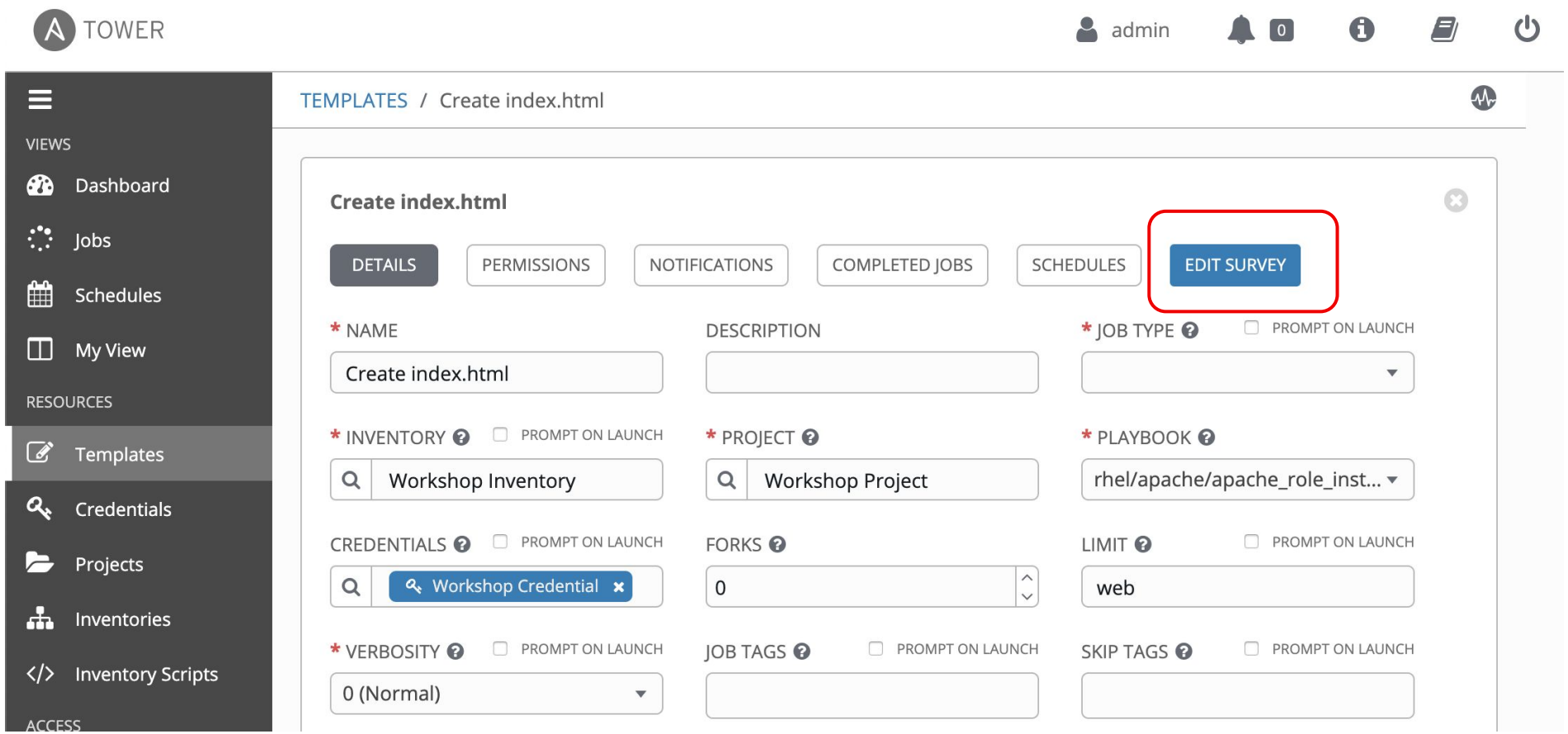
An Ansible Tower survey is a simple question-and-answer form that allows users to customize their job runs. Combine that with Tower's role-based access control, and you can build simple, easy self-service for your users.

Red Hat

Creating a Survey (1/2)

Once a Job Template is saved, the **Add Survey Button** will appear

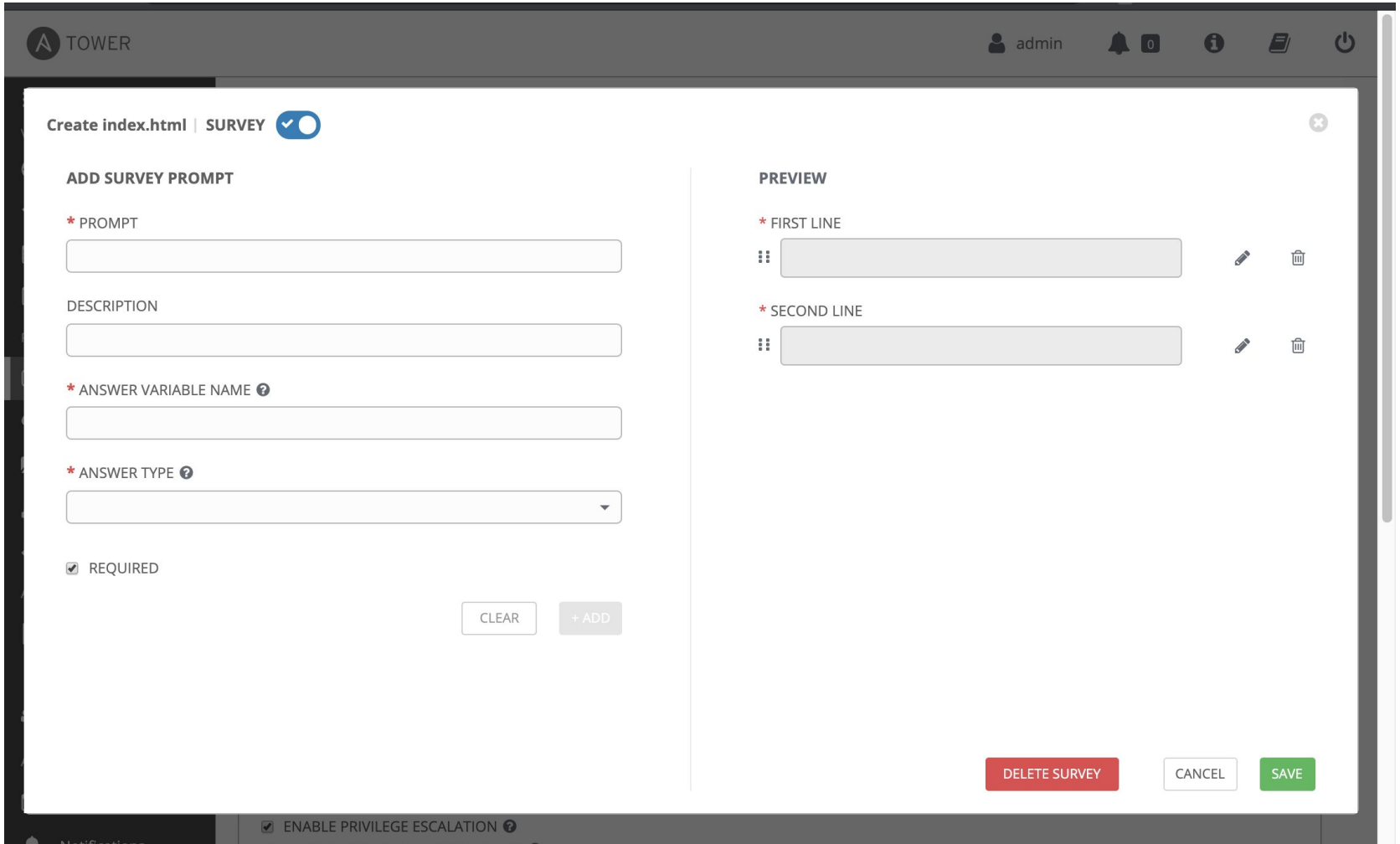
Click the button to open the Add Survey window.



Red Hat

Creating a Survey (2/2)

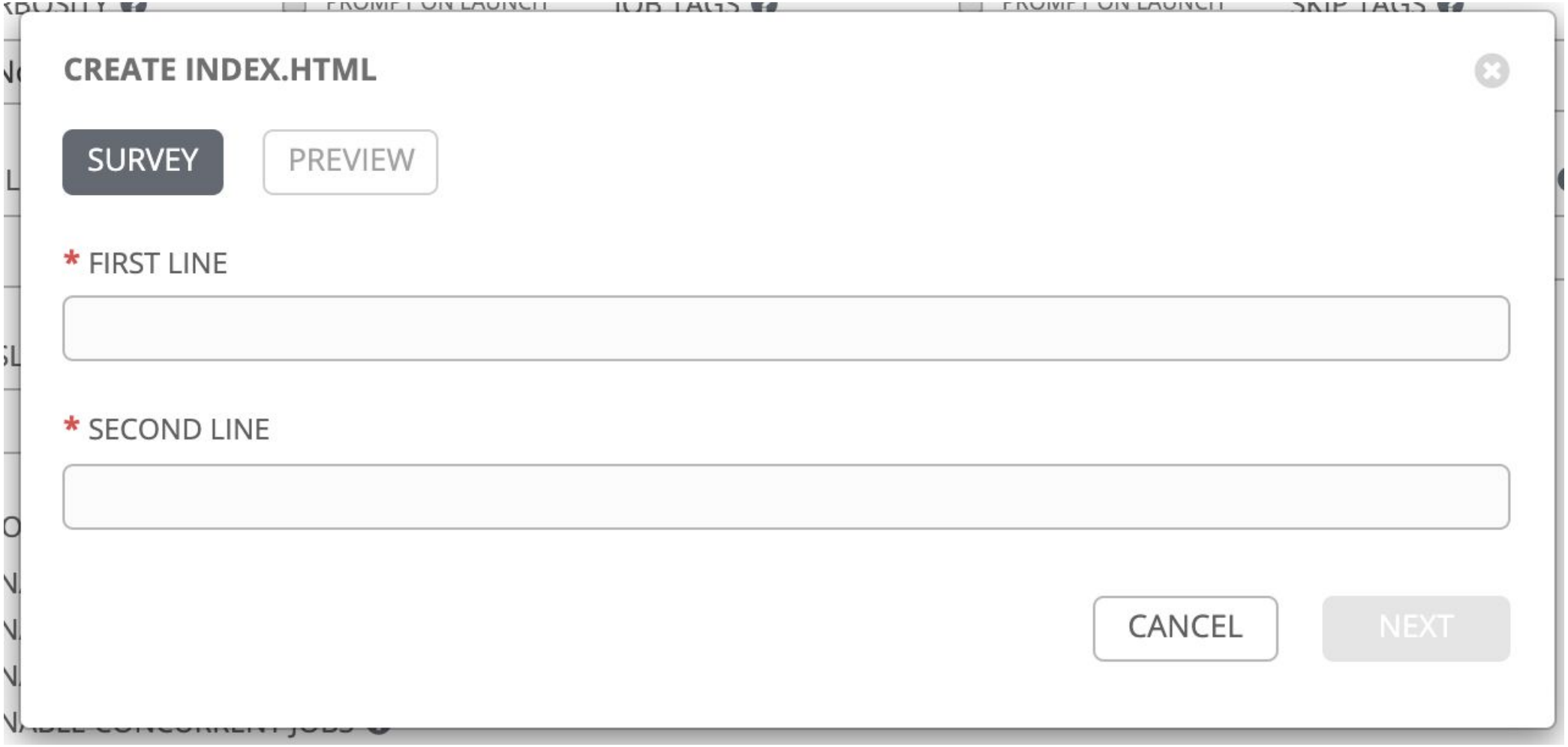
The Add Survey window allows the Job Template to prompt users for one or more questions. The answers provided become variables for use in the Ansible Playbook.



Red Hat

Using a Survey

When launching a job, the user will now be prompted with the Survey. The user can be required to fill out the Survey before the Job Template will execute.



Red Hat


Red Hat
Ansible Automation
Platform
Lab Time
Complete exercise **5-surveys** now in your lab environment
https://red.ht/ansibleworkshop
Red Hat

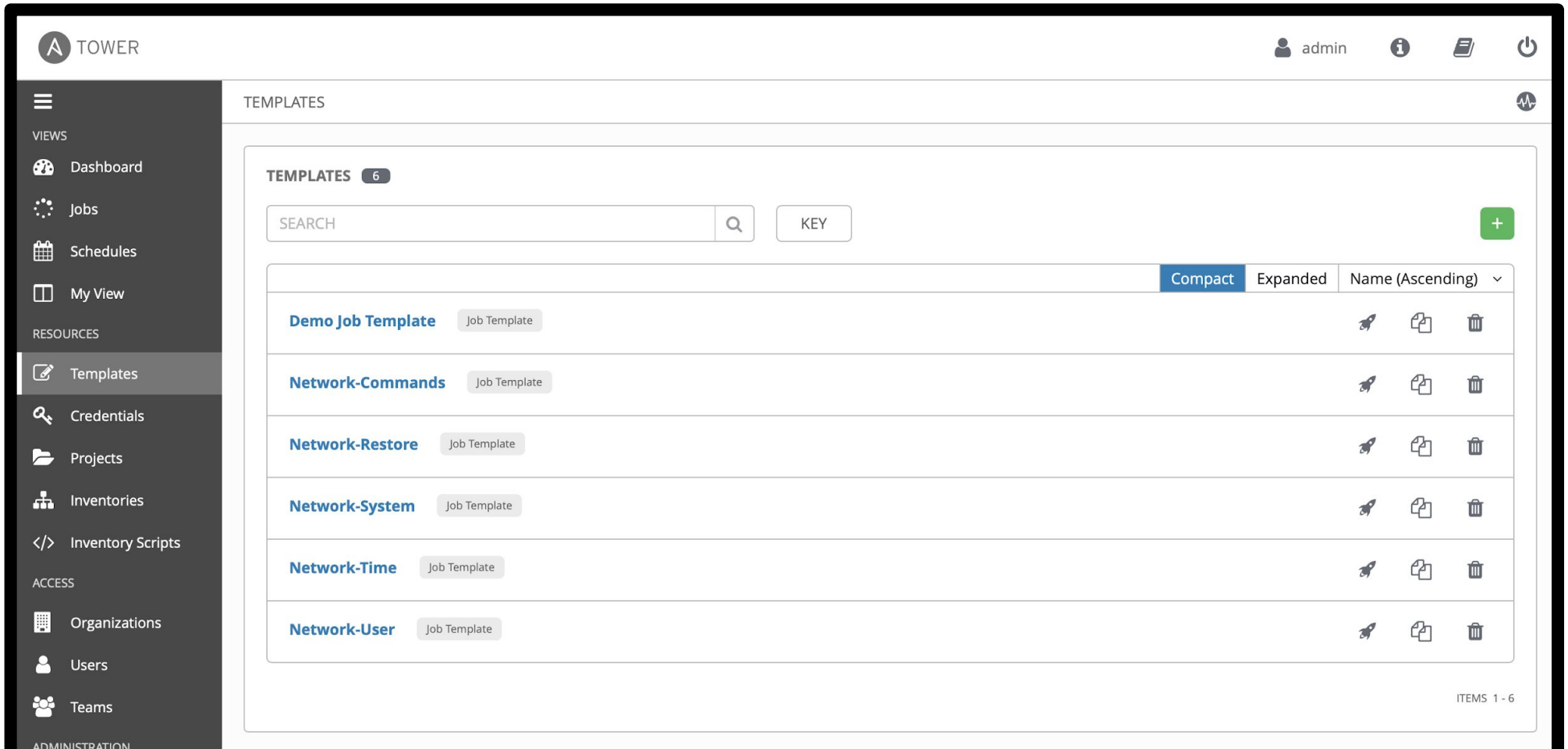
Exercise 6

Topics Covered:

- Workflows

Workflows

Workflows can be found alongside Job Templates by clicking the **Templates** button under the *RESOURCES* section on the left menu.

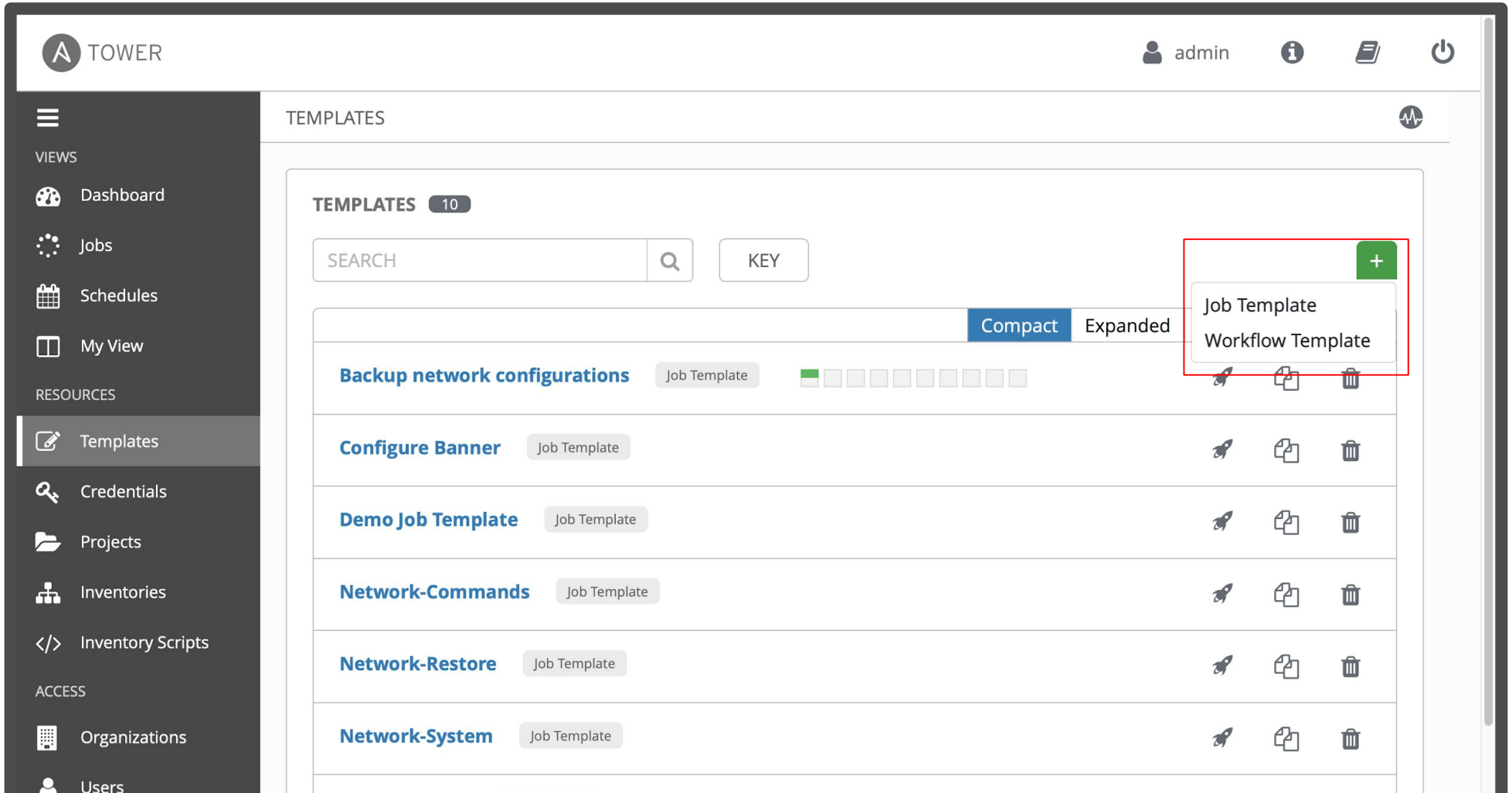


Red Hat

Adding a new Workflow Template

To add a new **Workflow** click on the green + button 

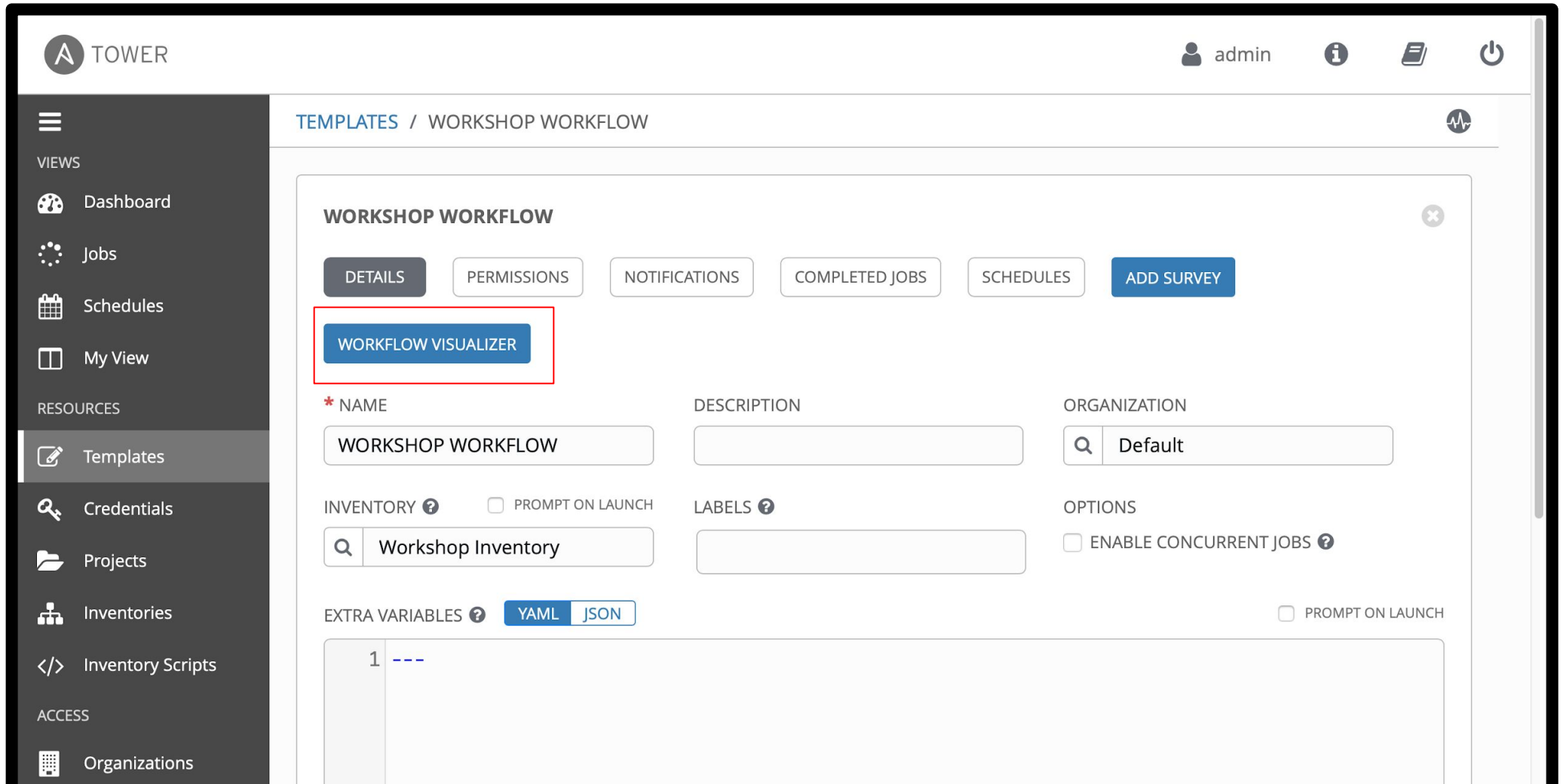
This time select the **Workflow Template**



Red Hat

Creating the Workflow

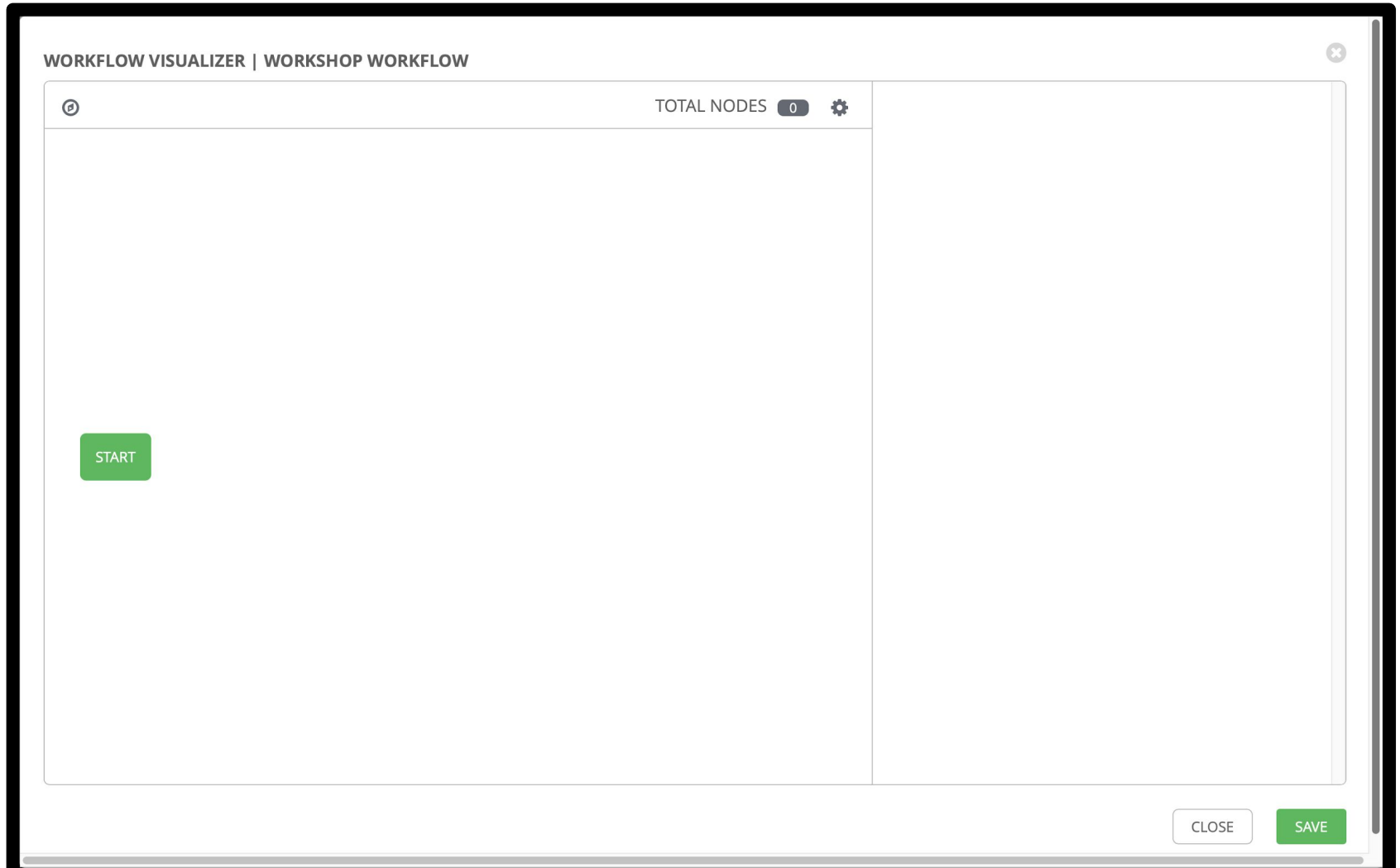
Fill out the required parameters and click **SAVE.** As soon as the Workflow Template is saved the WORKFLOW VISUALIZER will open.



Red Hat

Workflow Visualizer

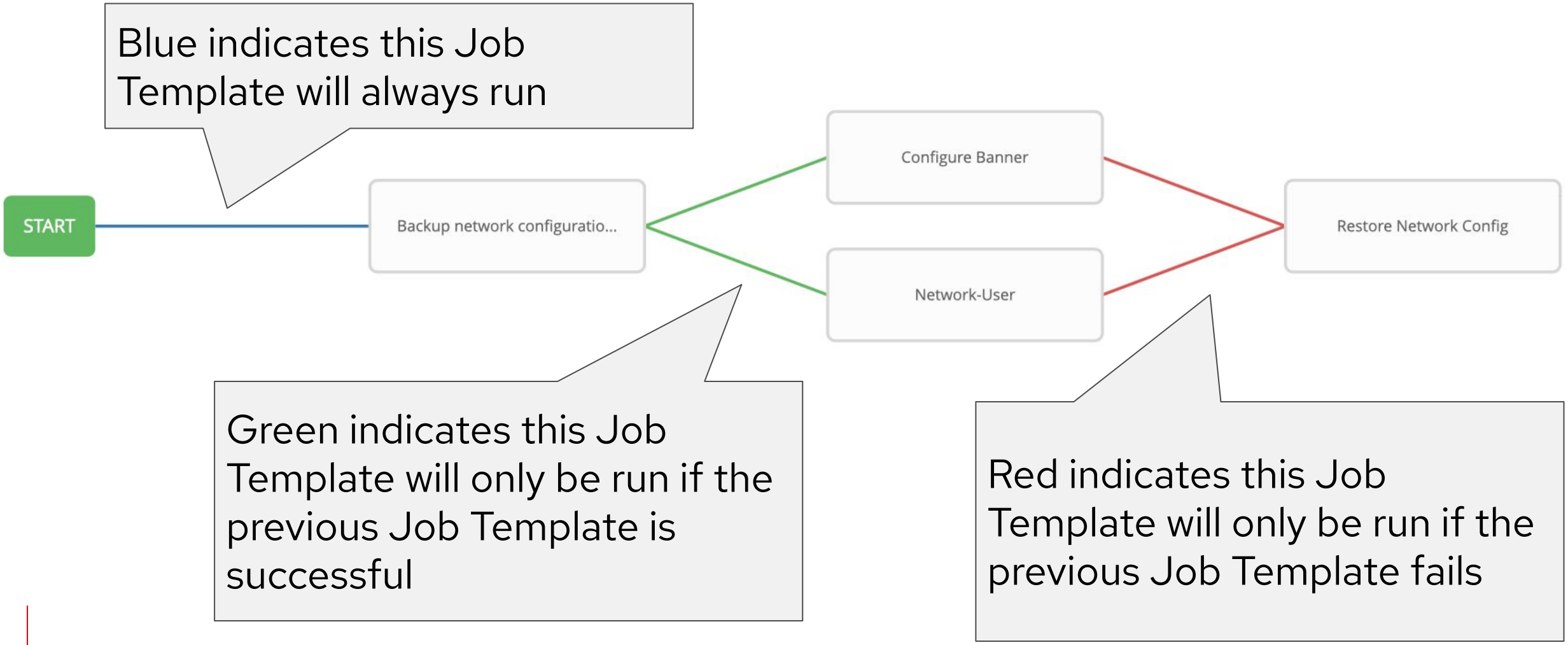
The workflow visualizer will start as a blank canvas.



Red Hat

Visualizing a Workflow

Workflows can branch out, or converge in.



Red Hat
Ansible Automation
Platform

Lab Time

Complete exercise **6-workflow** now in your lab environment

https://red.ht/ansibleworkshop

Red Hat

Next Steps

GET STARTED

ansible.com/get-started

ansible.com/tower-trial

JOIN THE COMMUNITY

ansible.com/community

WORKSHOPS & TRAINING

ansible.com/workshops

Red Hat Training

SHARE YOUR STORY

Follow us @Ansible

Friend us on Facebook

Red Hat

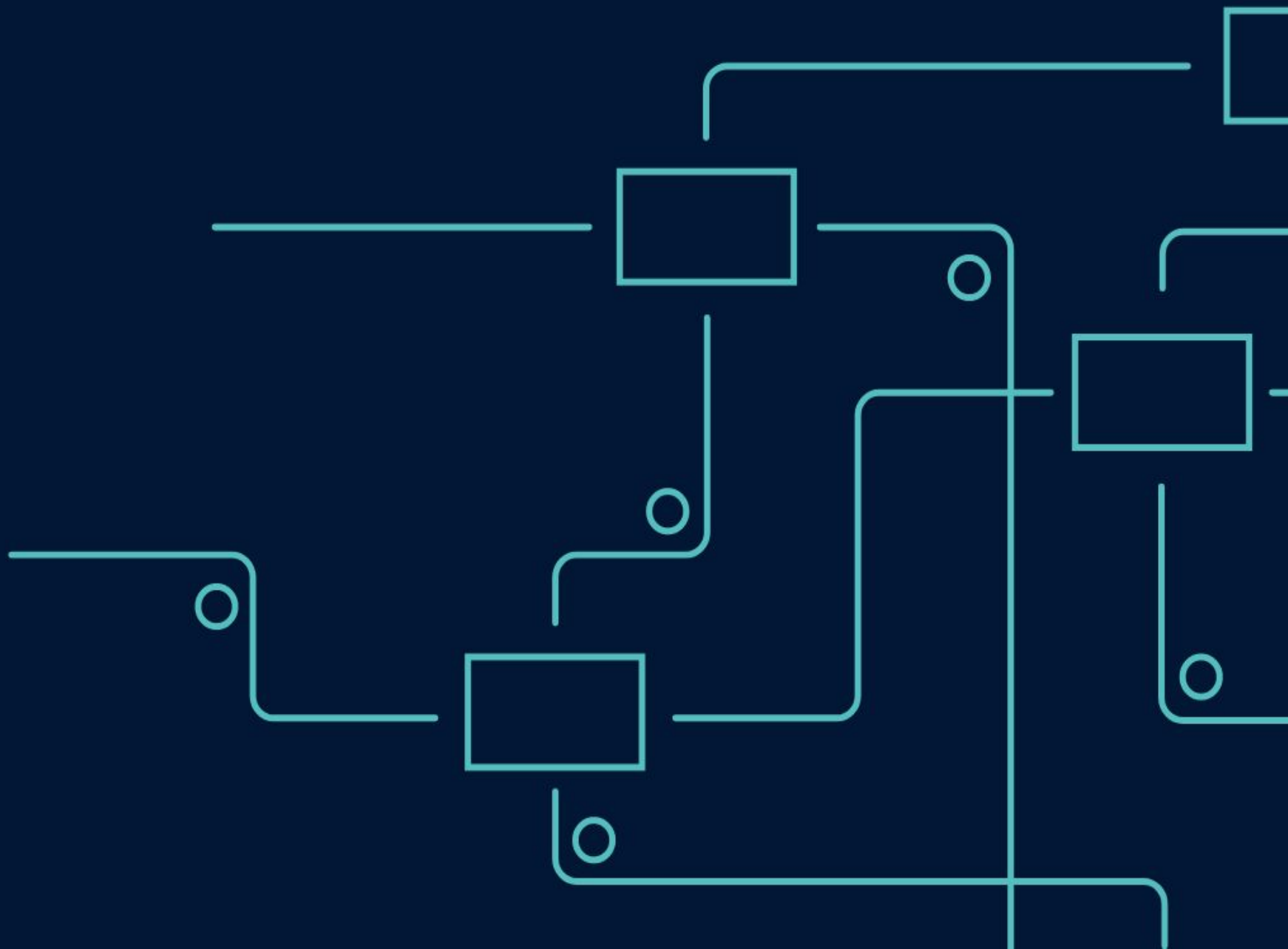
AnsibleFest

Red Hat
Ansible Automation
Platform

AnsibleFest 2021

Automate. Innovate. Accelerate.
September 29–30

Register now



Thank you

http://bit.ly/3jZSbJH

linkedin.com/company/red-hat

youtube.com/AnsibleAutomation

facebook.com/ansibleautomation

twitter.com/ansible

github.com/ansible

Red Hat