HOW TO USE THIS DECK

This slide deck is meant to accompany the Ansible Security workshop, both sections.
Note that this deck is optional - the workshop content explains each and every Ansible idea in detail already.

HOW TO IMPROVE THIS DECK

The workshop is a collaborative effort. Help us to improve it! You can leave comments, and the BU will make sure to work on this. Tag for example Roland (Wolters) or Sean (Cavanaugh) to ensure that they pick it up.

Speaking about the BU: the fact that this deck is now owned by an organization and not individuals anymore hopefully ensures for the future that the deck stays up2date over time as the workshop develops.

WHO IS THE AUDIENCE FOR THIS WORKSHOP

The workshop is intended for people who want to learn how Ansible can be leveraged in security environments. The workshop is intended for technical professionals in automation [supporting horizontally other teams in their company], security operations and vulnerability management.
There is no previous knowledge about Ansible required to access this workshop, though it certainly helps.

Red Hat
Ansible Automation
Platform

Ansible Security Automation Workshop

Introduction to Ansible Security Automation for System Administrators and Security Operators

Red Hat

Housekeeping

- Timing
- Breaks
- Takeaways

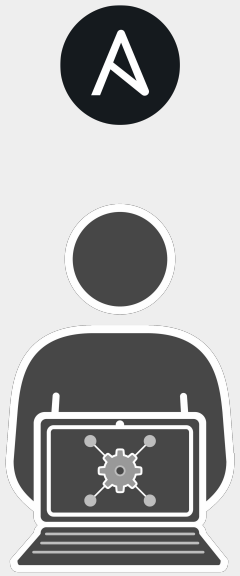
What you will learn

- Introduction to Ansible Security Automation
- How it works
- Understanding modules, tasks & playbooks
- How to use Ansible with various security tools
  - SIEM: QRadar
  - IDS: Snort
  - Firewall: Check Point NGFW

Introduction

Topics Covered:

- What Ansible Automation is
- What it can do



Automation happens when one person meets a problem they never want to solve again

Red Hat

Teams are automating...

Lines Of Business

Network

Security

Operations

Developers

Infrastructure

Ad-hoc Automation is happening in silos

Developers → Ansible used in silo

Security → DIY scripting automation

Infrastructure → Open source config management tool

Network → Proprietary vendor supplied automation

Is organic automation enough?

Why Ansible?

Simple

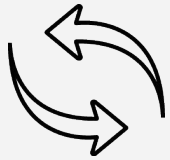
Human readable automation

No special coding skills needed

Tasks executed in order

Usable by every team

Get productive quickly



Powerful

App deployment

Configuration management

Workflow orchestration

Network automation

Orchestrate the app lifecycle

Agentless

Agentless architecture

Uses OpenSSH & WinRM

No agents to exploit or update

Get started immediately

More efficient & more secure

Red Hat

What can I do using Ansible?

Automate the deployment and management of your entire IT footprint.

Do this...

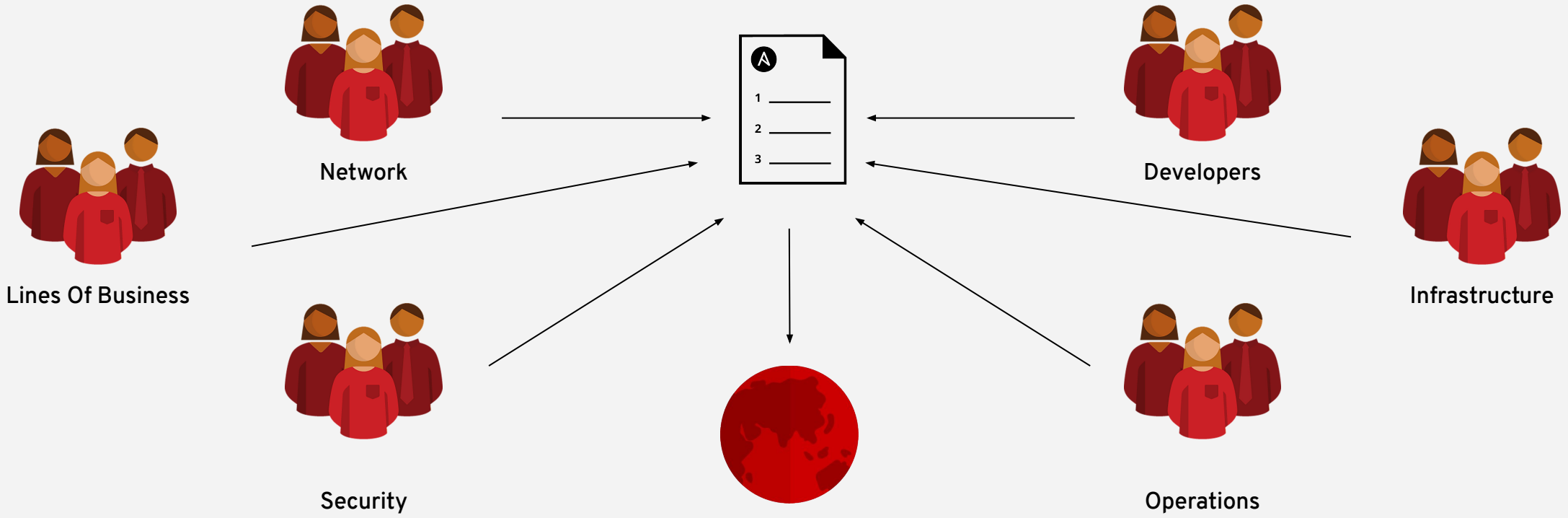
Orchestration	Configuration Management	Application Deployment	Provisioning	Continuous Delivery	Security and Compliance

On these...

Firewalls	Load Balancers	Applications	Containers	Clouds
Servers	Infrastructure	Storage	Network Devices	**And more...**

Red Hat

When automation crosses teams, you need an automation platform



Red Hat Ansible Automation Platform

Network | Lines of business | Security

Operations | Infrastructure | Developers

Engage — **Ansible SaaS:** Engage users with an automation focused experience

Scale — **Ansible Tower:** Operate & control at scale

Create — **Ansible Engine:** Universal language of automation

Fueled by an open source community

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
Linux
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

A10
Arista
Aruba
Cumulus
Bigswitch
Cisco
Dell
Extreme
F5
Lenovo
MikroTik
Juniper
OpenSwitch
+more

Security

Checkpoint
Cisco
CyberArk
F5
Fortinet
Juniper
IBM
Palo Alto
Snort
+more

Monitoring

Dynatrace
Datadog
LogicMonitor
New Relic
Sensu
+more

Devops

Jira
GitHub
Vagrant
Jenkins
Slack
+more

Red Hat Ansible Tower

by the numbers:

94% Reduction in recovery time following a security incident

84% Savings by deploying workloads to generic systems appliances using Ansible Tower

67% Reduction in man hours required for customer deliveries

Financial summary:

146%

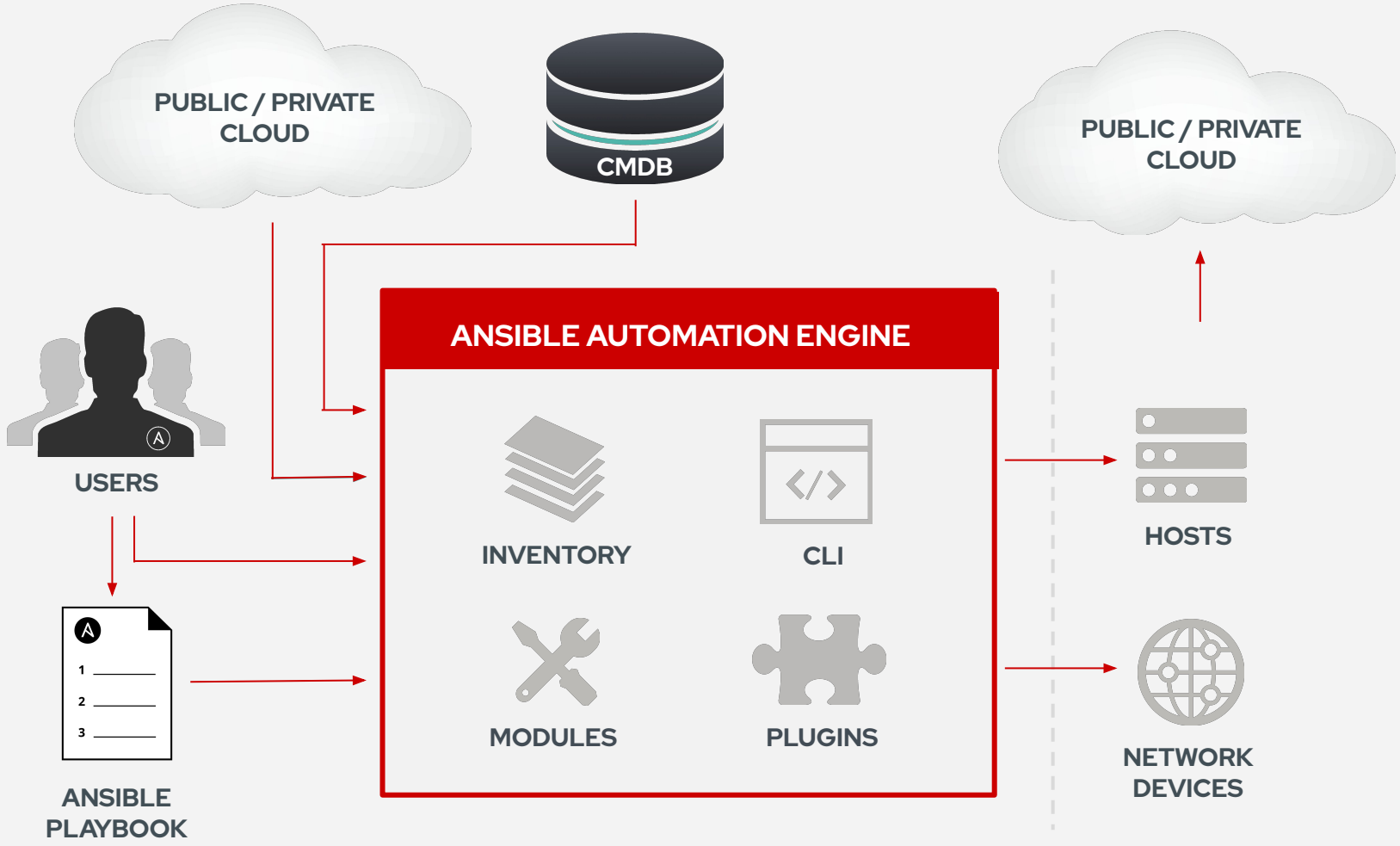
ROI on Ansible Tower

<3 MONTHS

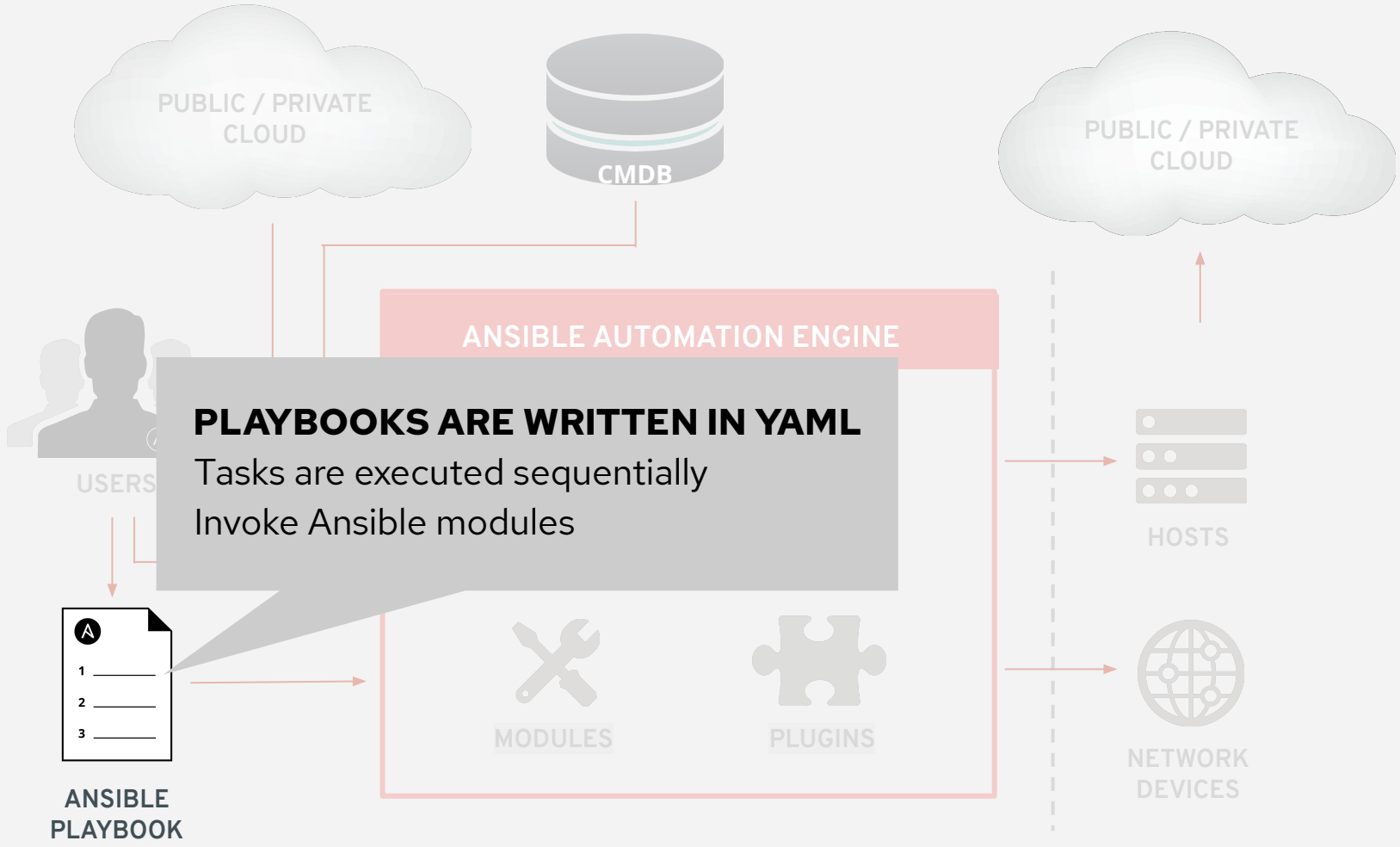
Payback on Ansible Tower

SOURCE: *"The Total Economic Impact™ Of Red Hat Ansible Tower, a June 2018 commissioned study conducted by Forrester Consulting on behalf of Red Hat."*
redhat.com/en/engage/total-economic-impact-ansible-tower-20180710

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


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

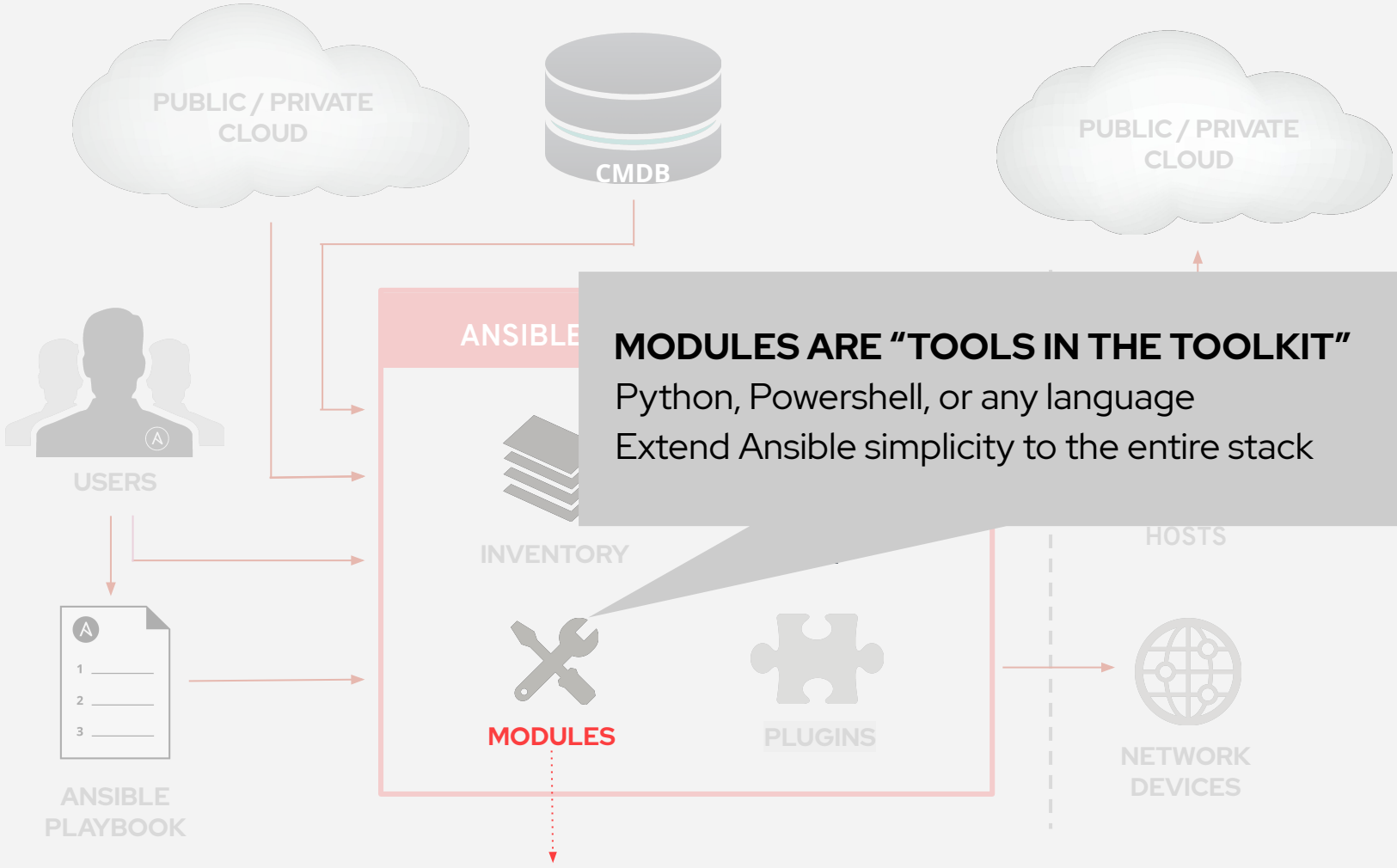
Red Hat

```
---
- 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
      template:
        src: files/index.html
        dest: /var/www/html/

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



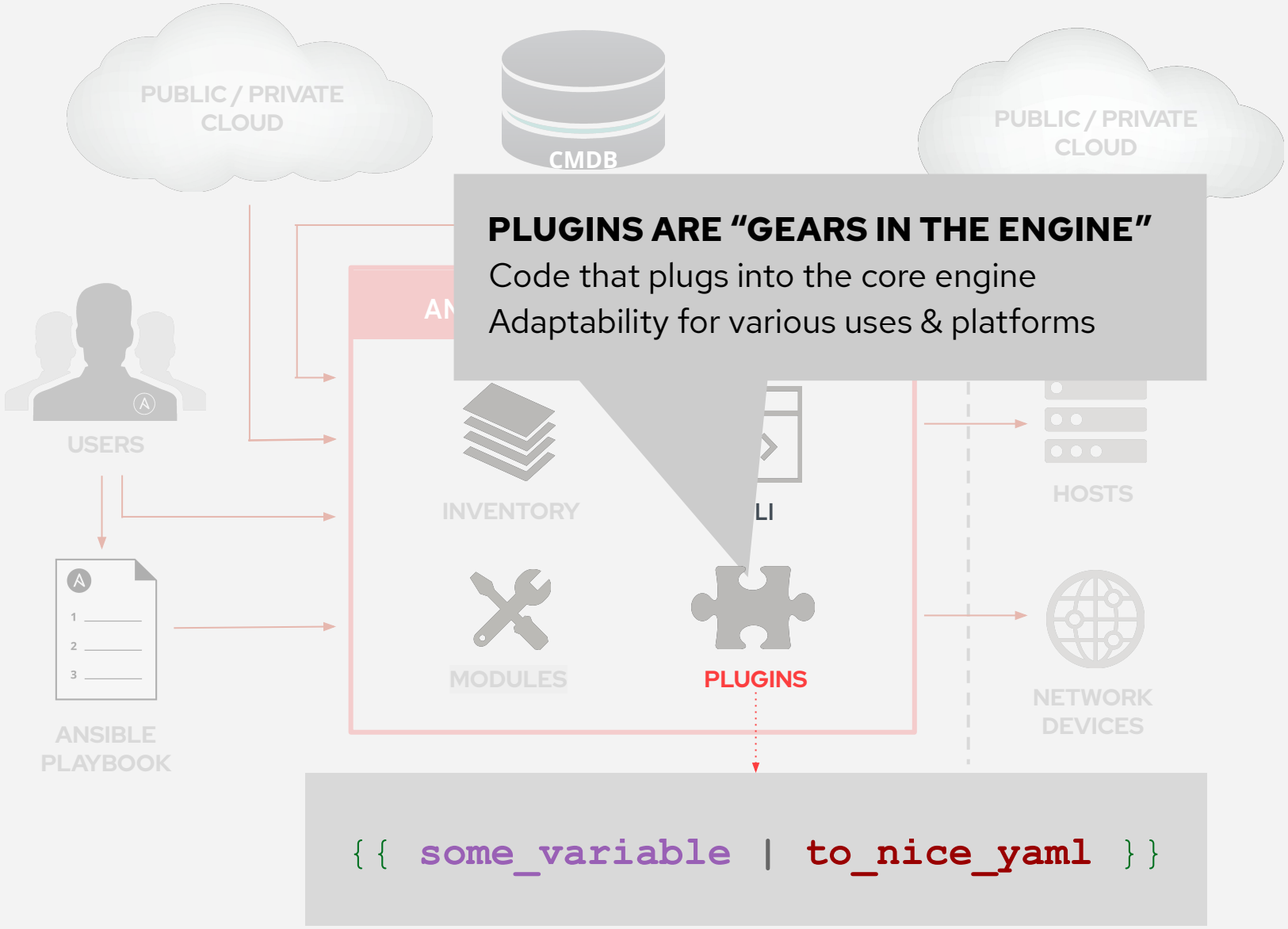
MODULES ARE "TOOLS IN THE TOOLKIT"

Python, Powershell, or any language

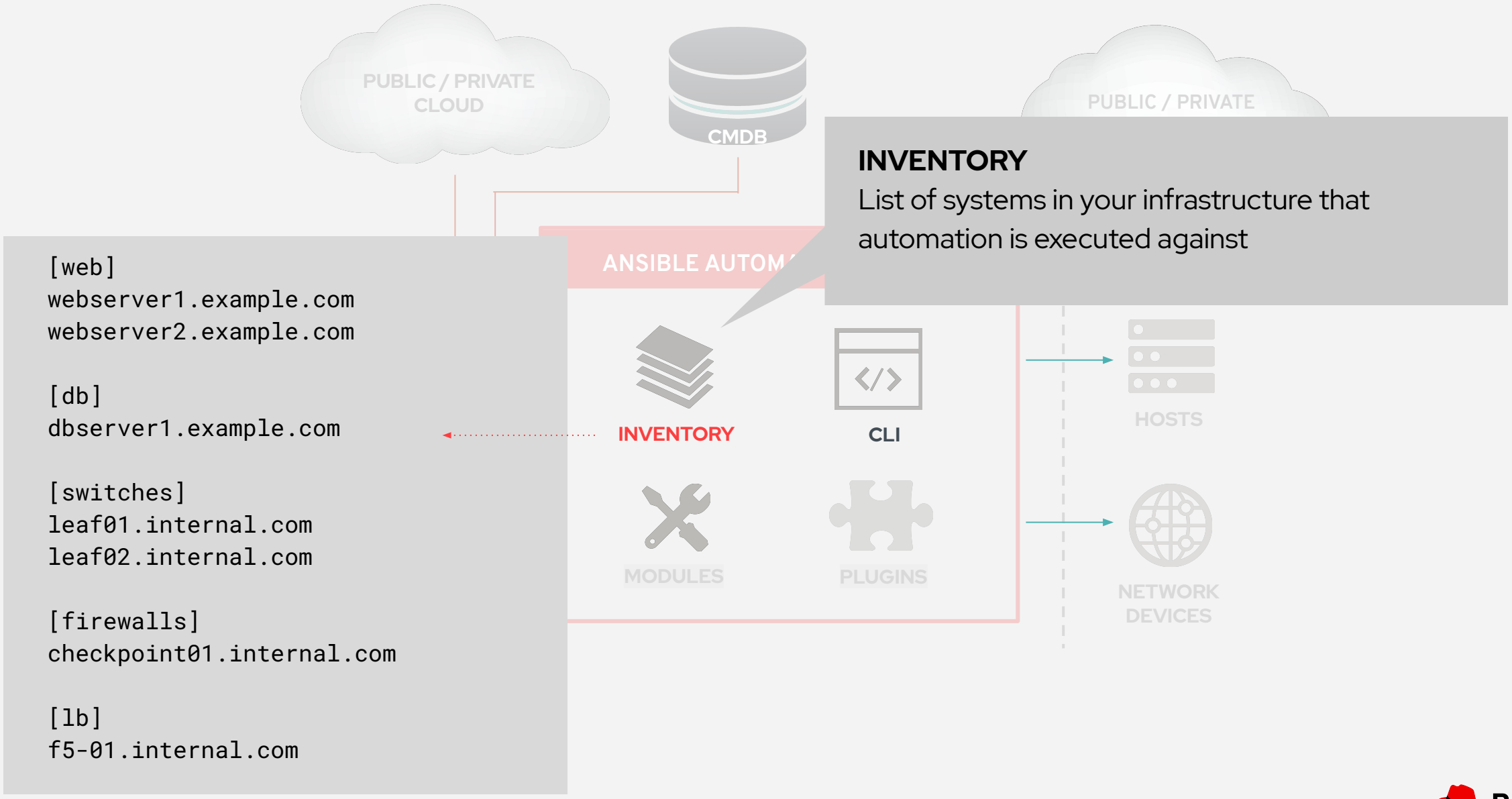
Extend Ansible simplicity to the entire stack

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

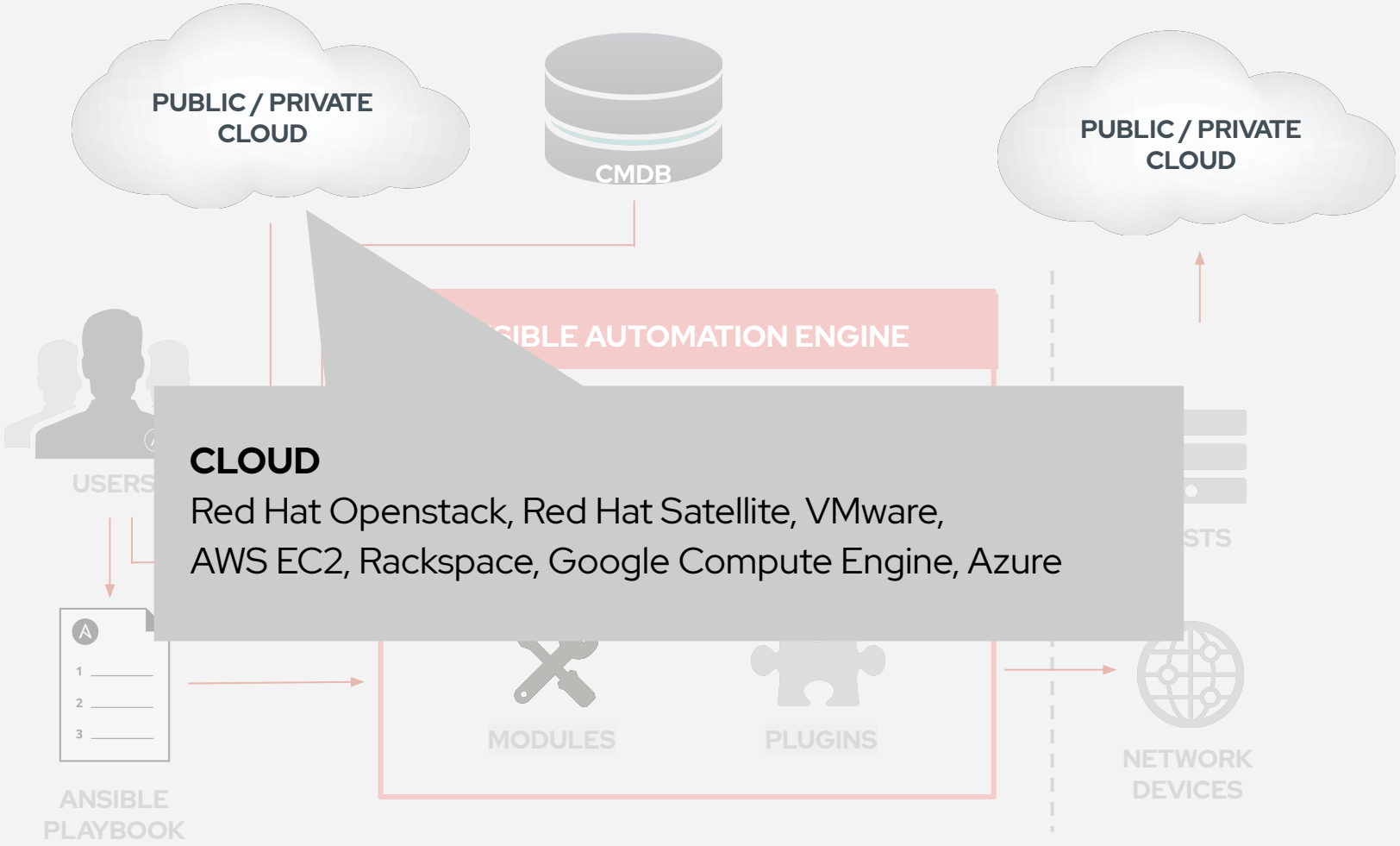
Red Hat


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
PLUGINS ARE “GEARS IN THE ENGINE”
Code that plugs into the core engine
Adaptability for various uses & platforms
USERS
INVENTORY
HOSTS
MODULES
PLUGINS
NETWORK DEVICES
ANSIBLE PLAYBOOK
{{ some_variable | to_nice_yaml }}

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

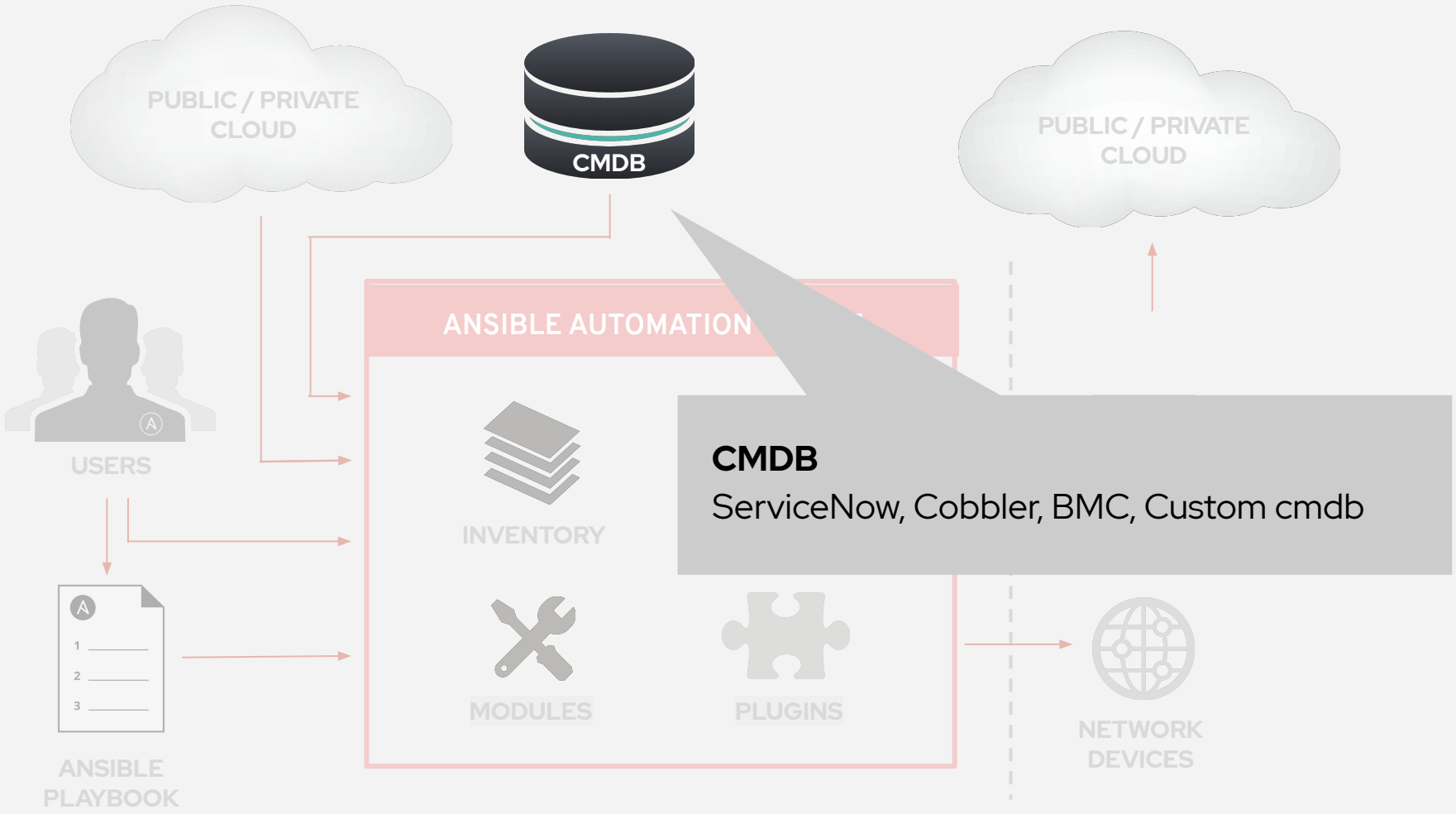
Red Hat



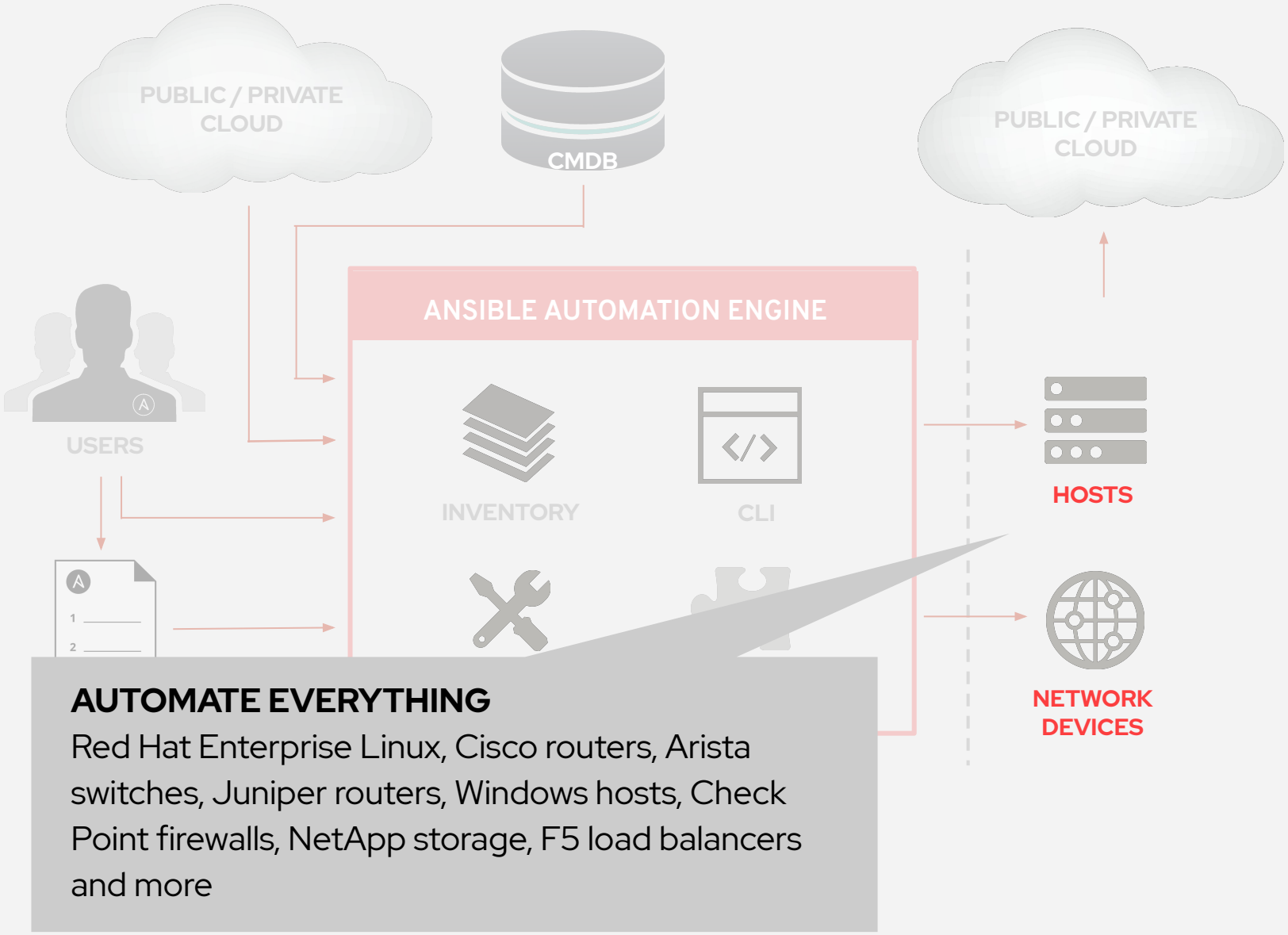
CLOUD

Red Hat Openstack, Red Hat Satellite, VMware,
AWS EC2, Rackspace, Google Compute Engine, Azure

Red Hat


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
ANSIBLE AUTOMATION
USERS
INVENTORY
CMDB
ServiceNow, Cobbler, BMC, Custom cmdb
MODULES
PLUGINS
NETWORK DEVICES
ANSIBLE PLAYBOOK

Red Hat


PUBLIC / PRIVATE CLOUD
CMDB
PUBLIC / PRIVATE CLOUD
ANSIBLE AUTOMATION ENGINE
USERS
INVENTORY
CLI
HOSTS
NETWORK DEVICES
AUTOMATE EVERYTHING
Red Hat Enterprise Linux, Cisco routers, Arista switches, Juniper routers, Windows hosts, Check Point firewalls, NetApp storage, F5 load balancers and more

Red Hat

LINUX AUTOMATION

150+

Linux Modules

AUTOMATE EVERYTHING LINUX

Red Hat Enterprise Linux, BSD, Debian, Ubuntu and many more!

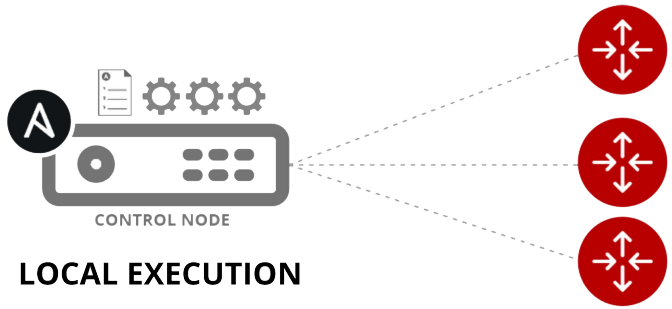
ONLY REQUIREMENTS:
Python 2 (2.6 or later)
***or* Python 3 (3.5 or later)**

ansible.com/get-started

Red Hat

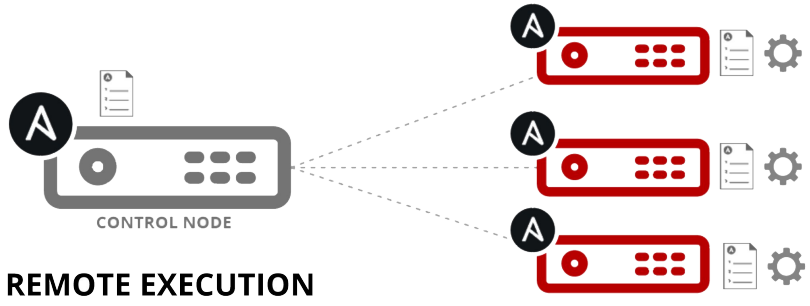
How Ansible Automation works

Module code is executed locally on the control node



NETWORKING DEVICES

Module code is copied to the managed node, executed, then removed



LINUX/WINDOWS HOSTS

Red Hat

Section 1 Introduction to Ansible Security Automation Basics

Red Hat
Ansible Automation
Platform

Exercise 1.1

Topics Covered:

- What Ansible Security Automation is about
- The lab infrastructure

Red Hat

Ansible Security - What Is It?

Ansible Security Automation is our expansion deeper into the security use case. The goal is to provide a more efficient, streamlined way for security teams to automate their various processes for the identification, search, and response to security events. This is more complex and higher-value than the application of a security baseline (PCI, STIG, CIS) to a server.

Ansible Security Automation is a supported set of Ansible modules, roles and playbooks designed to unify the security response to cyberattacks.

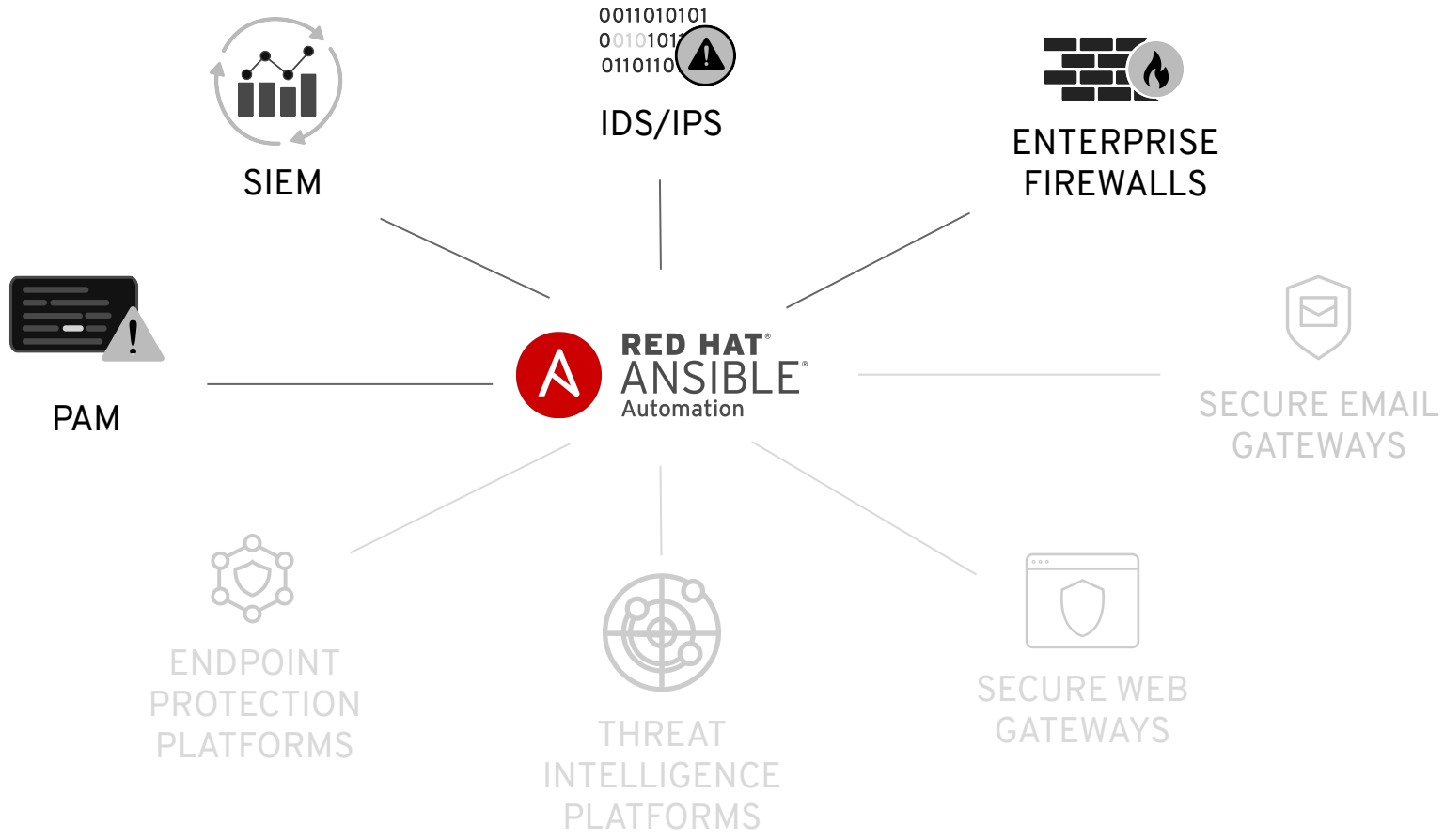
Is It A Security Solution?

No. Ansible can help Security teams “stitch together” the numerous security solutions and tools already in their IT environment for a more effective cyber defense.

By automating security capabilities, organizations can better unify responses to cyberattacks through the coordination of multiple, disparate security solutions, helping these technologies to act as one in the face of an IT security event.

Red Hat will not become a security vendor, we want to be a security enabler.

Ansible Security Automation



Red Hat

In this exercise: Verify Access

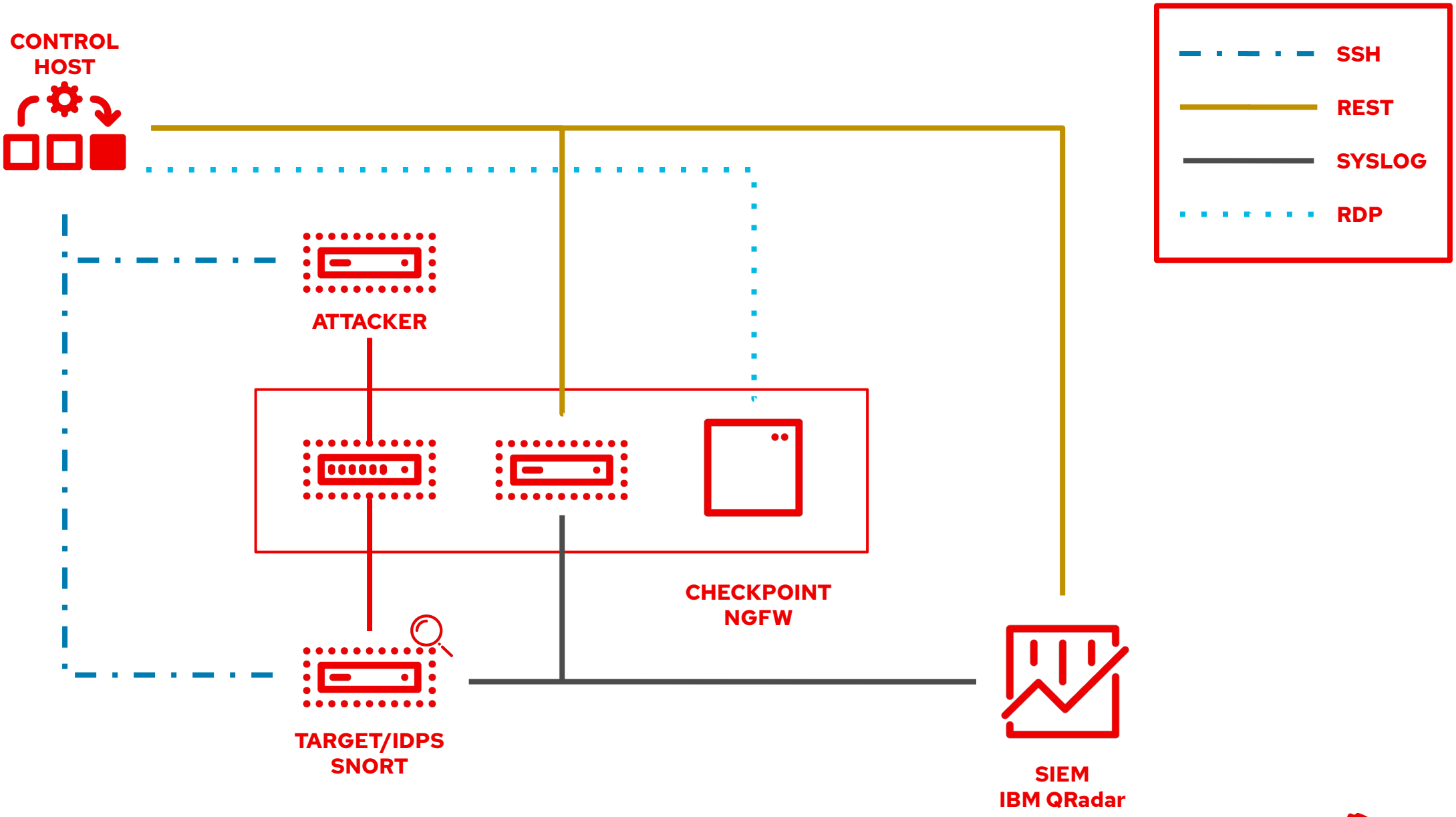
- Follow the steps to access environment
- Use the IP provided to you, the script only has example IPs
- Access to machines is done via online editor with a built-in terminal

Ansible 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

Your inventory

- Contains all machines of your environment
- Setup up just for you, individually
- Note your individual IP addresses for each machine - often in the script you need to replace example IP addresses with your individual ones


CONTROL
HOST
ATTACKER
CHECKPOINT
NGFW
TARGET/IDPS
SNORT
SIEM
IBM QRadar
SSH
REST
SYSLOG
RDP

Red Hat

Your inventory

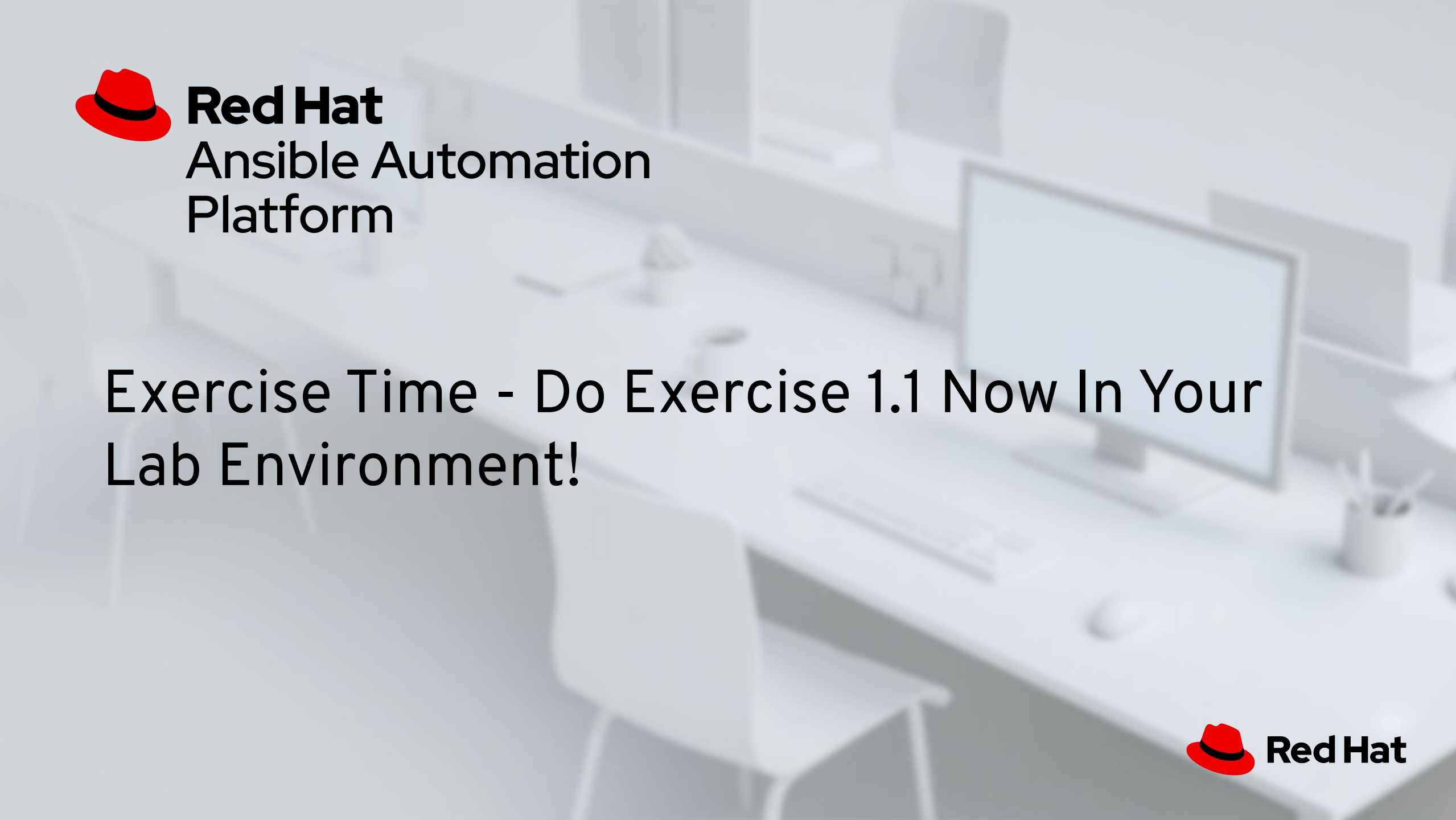
```
[all:vars]
ansible_user=student1
ansible_ssh_pass=ansible
ansible_port=22

[control]
ansible ansible_host=22.33.44.55 ansible_user=ec2-user private_ip=192.168.2.3

[siem]
qradar ansible_host=22.44.55.77 ansible_user=admin private_ip=172.16.3.44
ansible_httpapi_pass="Ansible1!" ansible_connection=httpapi ansible_httpapi_use_ssl=yes
ansible_httpapi_validate_certs=False ansible_network_os=ibm.qradar.qradar

[ids]
snort ansible_host=33.44.55.66 ansible_user=ec2-user private_ip=192.168.3.4

[firewall]
[...]
```


Red Hat
Ansible Automation
Platform
Exercise Time - Do Exercise 1.1 Now In Your Lab Environment!
Red Hat

Red Hat
Ansible Automation
Platform

Exercise 1.2

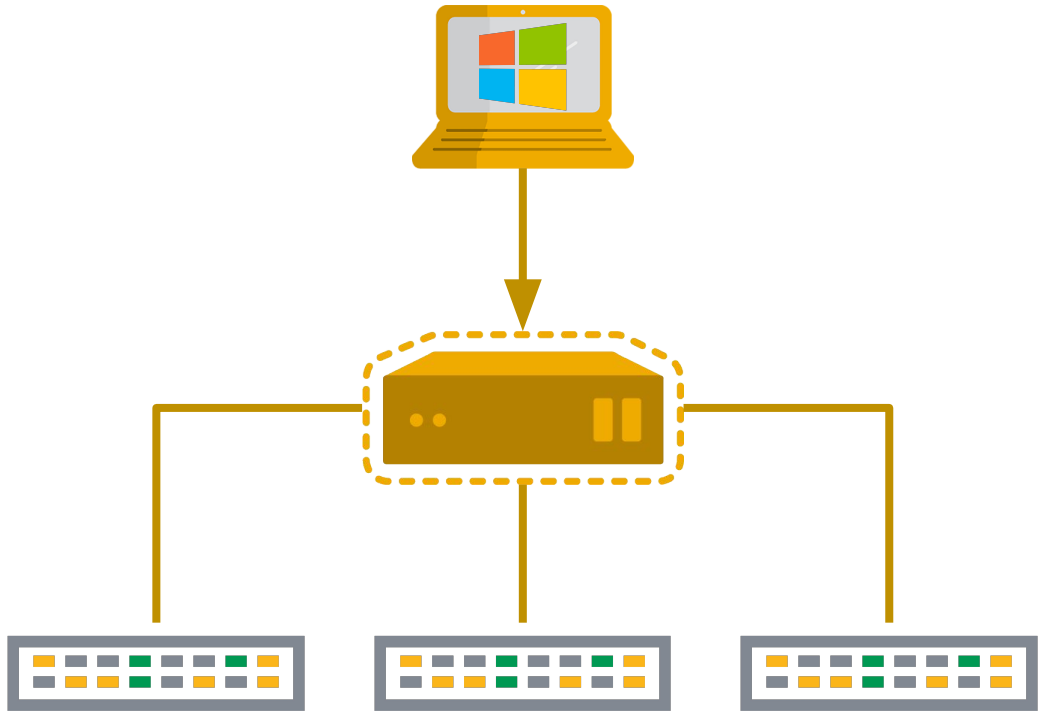
Topics Covered:

- Check Point Next Generation Firewall
- Access via Windows + SmartConsole
- Example interaction via Ansible
- Verify results in the UI

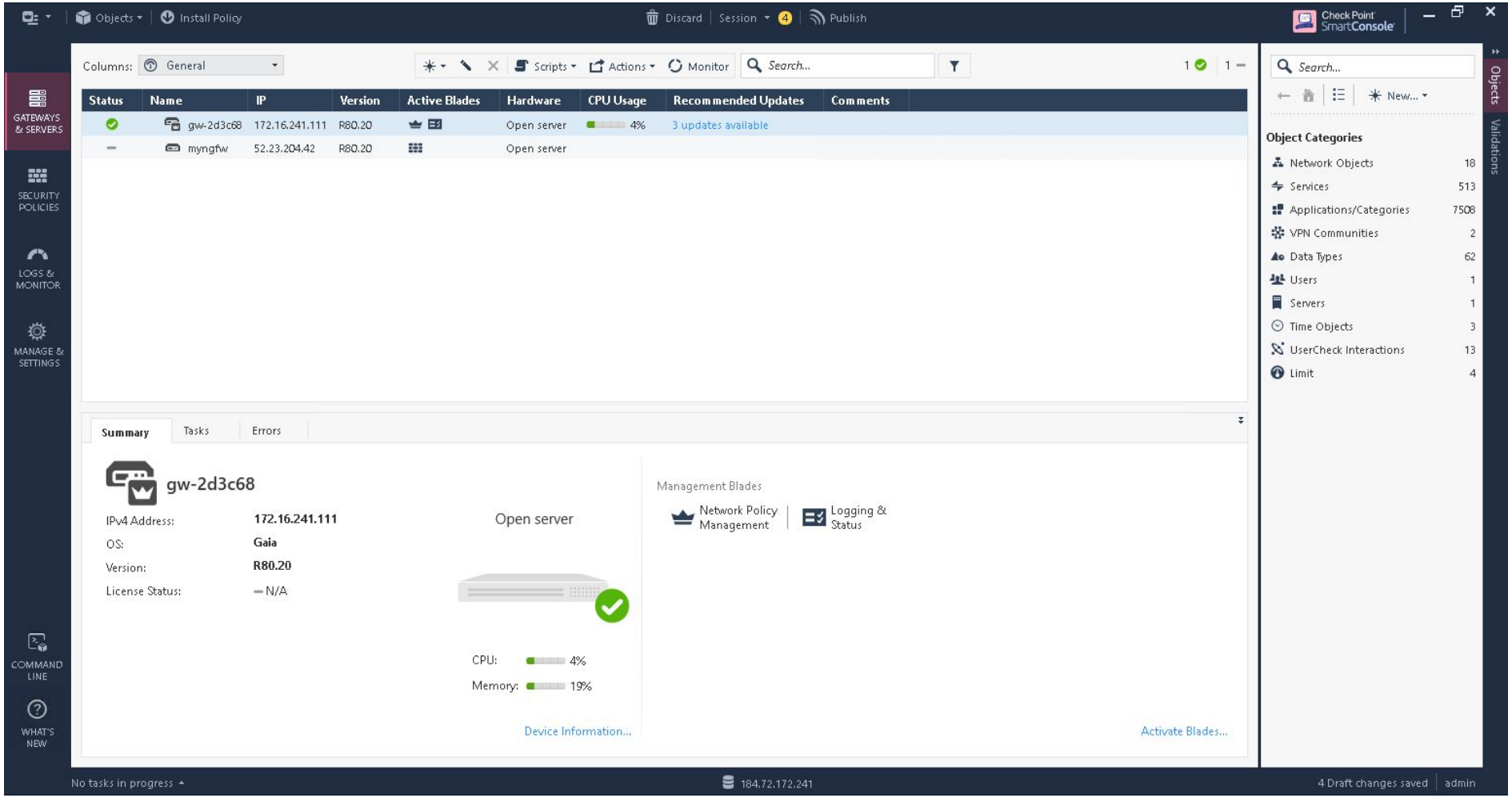
Accessing And Managing
Check Point Next Generation Firewalls

- Access only to central management server
- Via Windows management software, “SmartConsole”
- Automation: HTTP REST API

Lab students: via generic RDP client or RDP-HTML5 client



First Check Point Management Server Login



Red Hat

Run the first playbook

- Playbook is basically list of tasks
- Each task is using a module
- Roles: way to group tasks in re-usable way

```
---
- 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

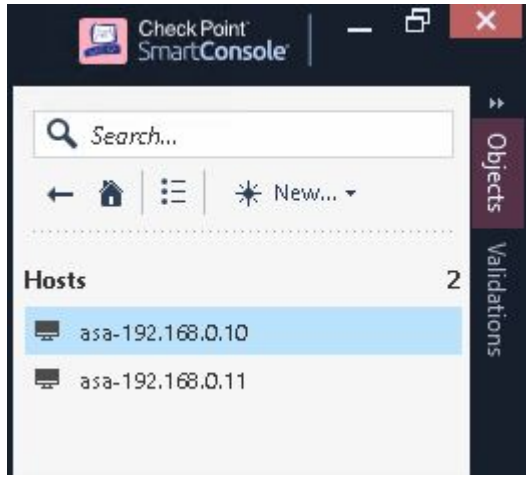
```
A task executed as expected, no change was made.

A task executed as expected, making a change

A task failed to execute successfully
```

Verify Results in UI

- Check network objects for added hosts
- Check policies for added policy



No.	Name	Source	Destination	VPN	Services & Applications	Action	Track	Install On
1	asa-drop-192.168.0.10-to-192.168.0.11	asa-192.168.0.10	asa-192.168.0.11	Any	Any	Drop	None	Policy Targets
2	Cleanup rule	Any	Any	Any	Any	Drop	None	Policy Targets

Red Hat

Exercise Time - Do Exercise 1.2 Now In Your Lab Environment!

Red Hat
Ansible Automation
Platform

Exercise 1.3

Topics Covered:

- Snort rules
- Running a playbook interacting with Snort

Red Hat

Snort - Network Intrusion Detection & Prevention System

- Real time traffic analysis and packet logging on IP networks
- Content search and matching
- Service running on possible targets
- in lab: RHEL instance, victim
- Configuration based on rules
- Access and automation: via SSH

Snort Rules

BASIC OUTLINE OF A SNORT RULE

```
[action][protocol][sourceIP][sourceport] -> [destIP][destport] ( [Rule options] )
```

Rule Header

RULE HEADER

The rule header contains the rule's action, protocol, source and destination IP addresses and netmasks, and the source and destination ports information.

alert **Action to take (option)** The first item in a rule is the rule action. The rule action tells Snort what to do when it finds a packet that matches the rule criteria (usually alert).

tcp **Type of traffic (protocol)** The next field in a rule is the protocol. There are four protocols that Snort currently analyzes for suspicious behavior - TCP, UDP, ICMP, and IP.

$EXTERNAL_NET **Source address(es)** variable or literal

$HTTP_PORTS **Source port(s)** variable or literal

-> **Direction operator** The direction operator -> indicates the orientation of the traffic to which the rule applies.

$HOME_NET **Destination address(es)** variable or literal

any **Destination port(s)** variable or literal

EXAMPLE

Rule Header
```
alert tcp $EXTERNAL_NET $HTTP_PORTS -> $HOME_NET any
```

Message
```
msg: "BROWSER-IE Microsoft Internet Explorer
CacheSize exploit attempt";
```

Flow
```
flow: to_client,established;
```

Detection
```
file_data;
    content:"recordset"; offset:14; depth:9;
    content:".CacheSize"; distance:0; within:100;
    pcre:"/CacheSize\s*=\s*/";
    byte_test:10,>,0x3ffffffe,0,relative,string;
```

Metadata
```
policy max-detect-ips drop, service http;
```

References
```
reference:cve,2016-8077;
```

Classification
```
classtype: attempted-user;
```

Signature ID
```
sid:65535;rev:1;
```

Source: https://www.snort.org/documents/snort-rule-infographic

Ansible Role To Change Rules

- We have an Ansible role to change rules on Snort
- Takes care of service reloading, etc.
- Verification of changes:
  - file system entry
  - another role

What are Ansible roles?

- A way to load tasks, handlers, and variables from separate files
- Roles group content, allowing easy sharing of code with others
- Roles make larger projects more manageable
- Roles can be developed in parallel by different people

There are pre-built roles for Snort interaction available.

Role structure

- Defaults: default variables with lowest precedence (e.g. port)
- Handlers: contains all handlers
- Meta: role metadata including dependencies to other roles
- Tasks: plays or tasks
  Tip: It's common to include tasks in main.yml with "when" (e.g. OS == xyz)
- Templates: templates to deploy
- Tests: place for playbook tests
- Vars: variables (e.g. override port)

```
user/
├── defaults
│   └── main.yml
├── handlers
│   └── main.yml
├── meta
│   └── main.yml
├── README.md
├── tasks
│   └── main.yml
├── templates
├── tests
│   ├── inventory
│   └── test.yml
└── vars
    └── main.yml
```

```
---
- name: install compliance baseline
  hosts: web
  become: yes

  roles:
    - install_compliance_baseline
```

Red Hat

How To Install a Role

- Ansible Galaxy command
- Downloads roles from central Galaxy
- Also our roles written as part of the security initiative

```
$ ansible-galaxy install ansible_security.acl_manager
```

Exercise Time - Do Exercise 1.3 Now In Your Lab Environment!

Exercise 1.4

Topics Covered:

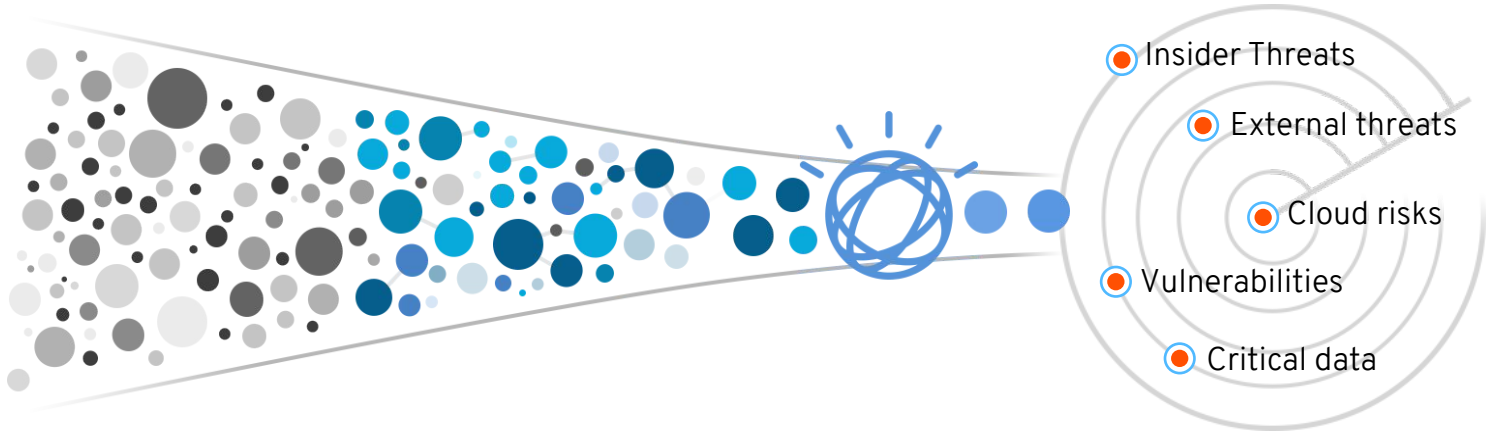
- Understanding QRadar
- Collections

IBM QRadar

Address most important security challenges

Complete Visibility	Prioritized Threats	Automated Investigations	Proactive Hunting

Endpoints
Network activity
Data activity
Users and identities
Threat intelligence
Configuration information
Vulnerabilities and threats
Application activity
Cloud platforms

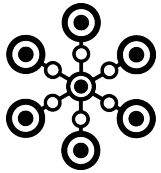


Red Hat

IBM QRadar: Automate Intelligence

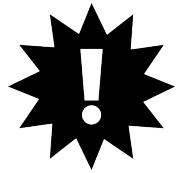
Detect

Known and unknown threats



Connect

Related activity in multi-stage attacks



Prioritize

Business critical events

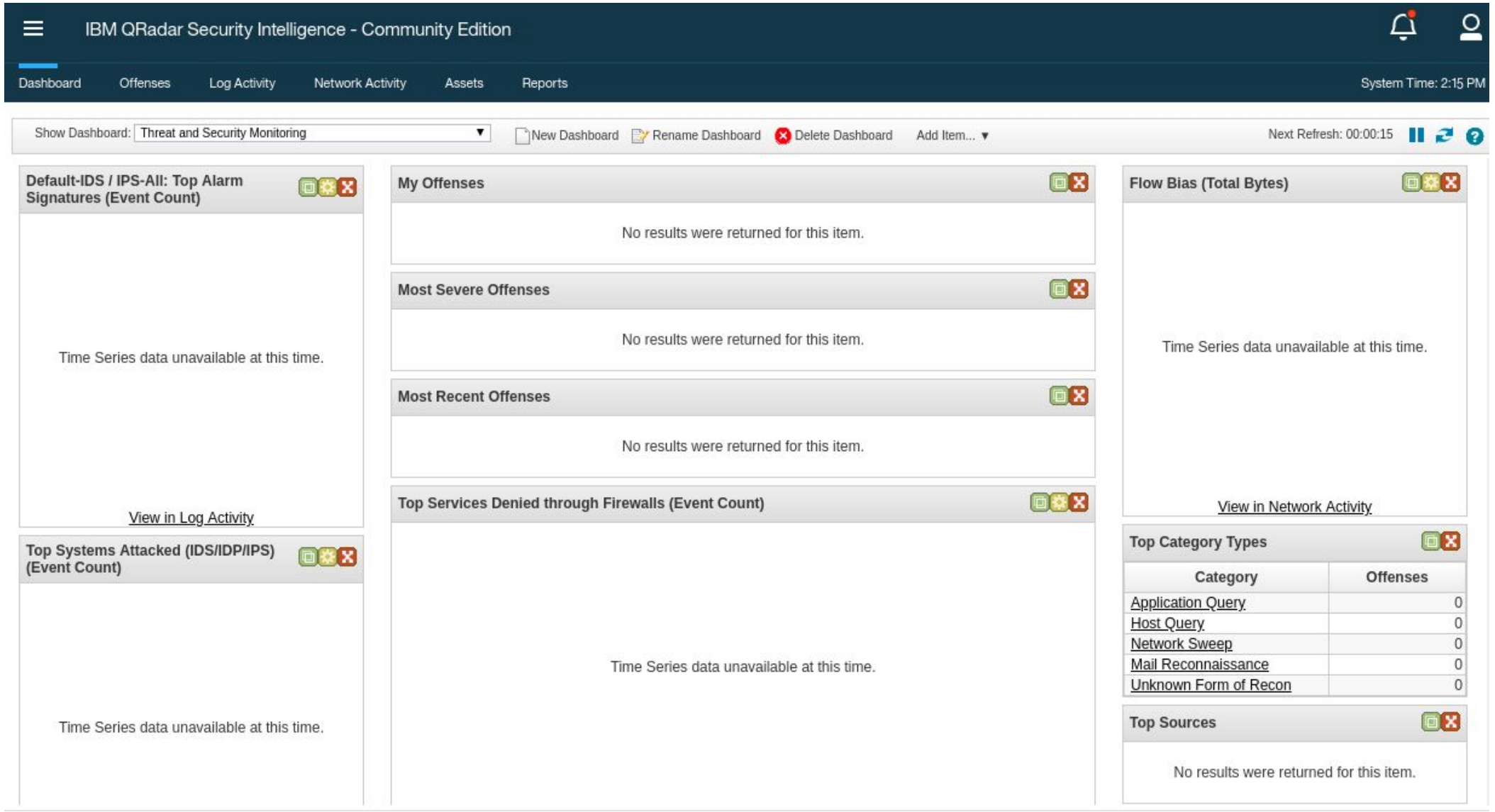
Investigate

Potential incidents to find root cause faster

QRadar

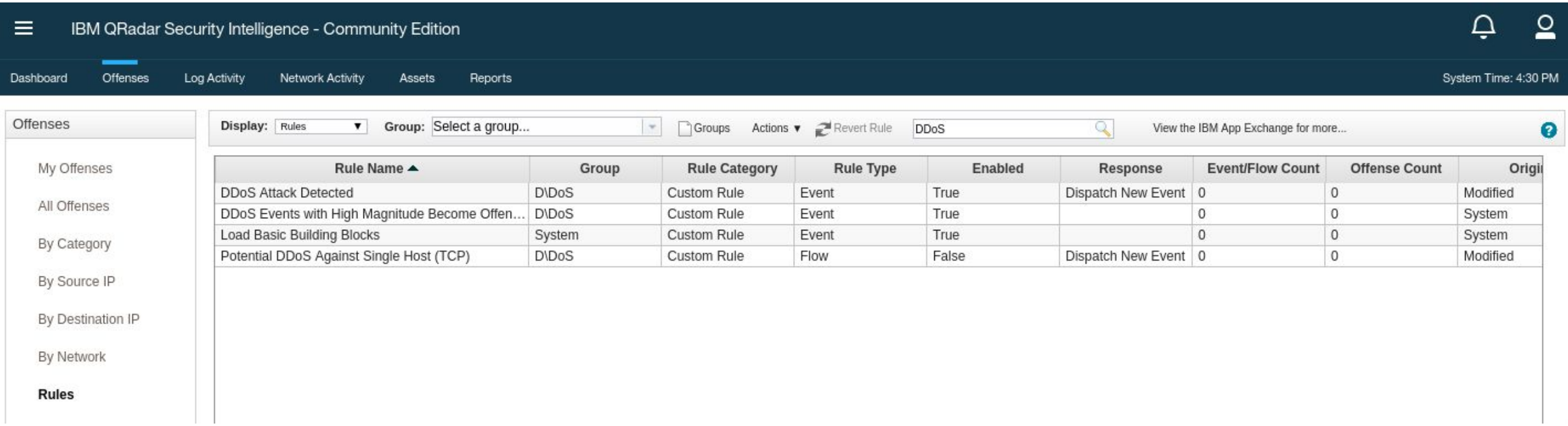
- SIEM - Security Information and Event Management
- Collects & analyses logs
- Can react on specific findings via “Offenses”
- Access via web UI
- Automation via REST API

QRadar



Red Hat

Verification In The UI

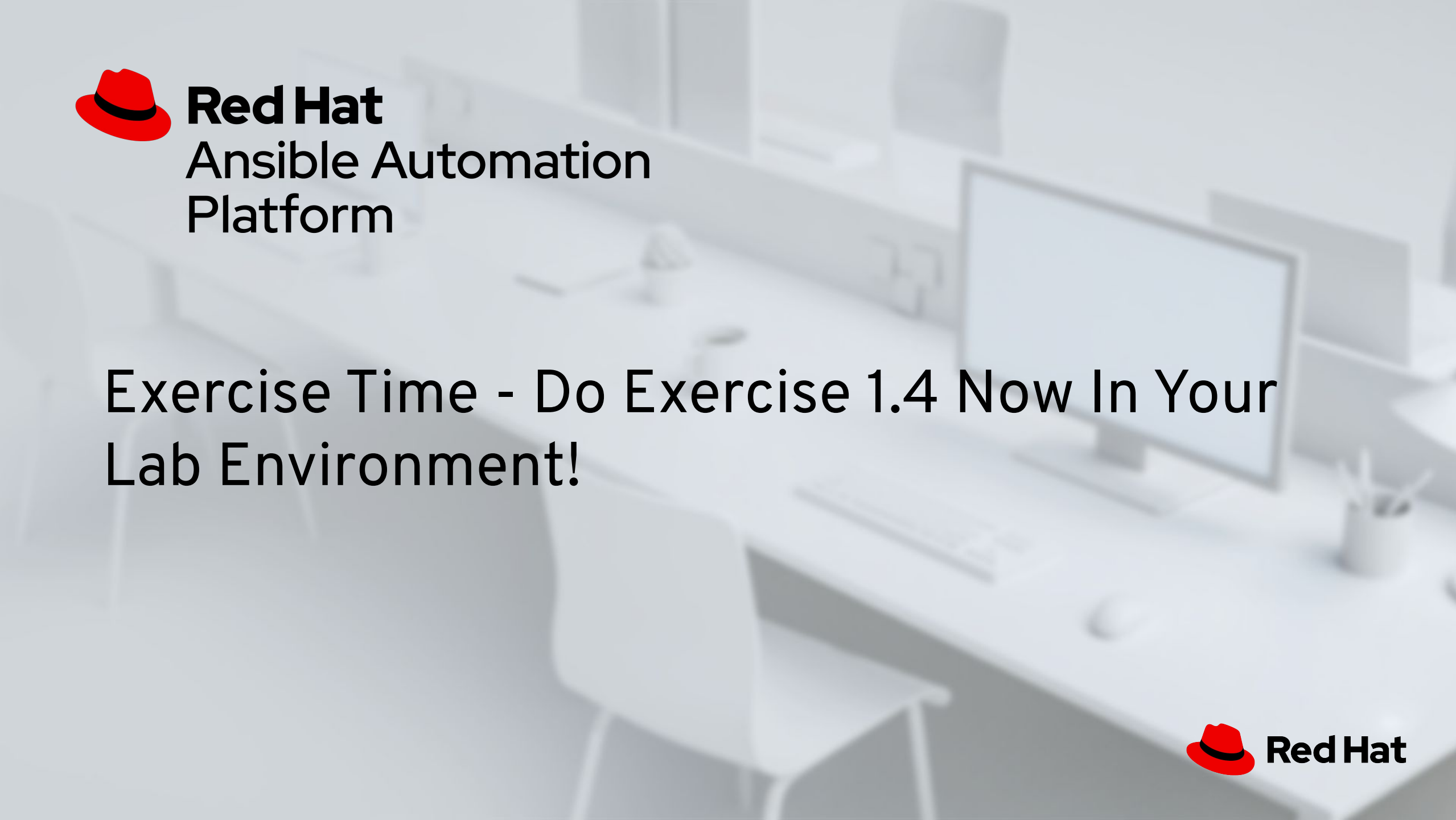


Red Hat

Collections

- Ansible content to interact with QRadar: provided as collections
- Like roles, but even more powerful
- Can also contain modules, connection plugins and so on

```
$ ansible-galaxy collection install ibm.qradar
```


Red Hat
Ansible Automation
Platform
Exercise Time - Do Exercise 1.4 Now In Your Lab Environment!
Red Hat

Section 2
Ansible Security Automation Use Cases

Red Hat

Tower Introduction

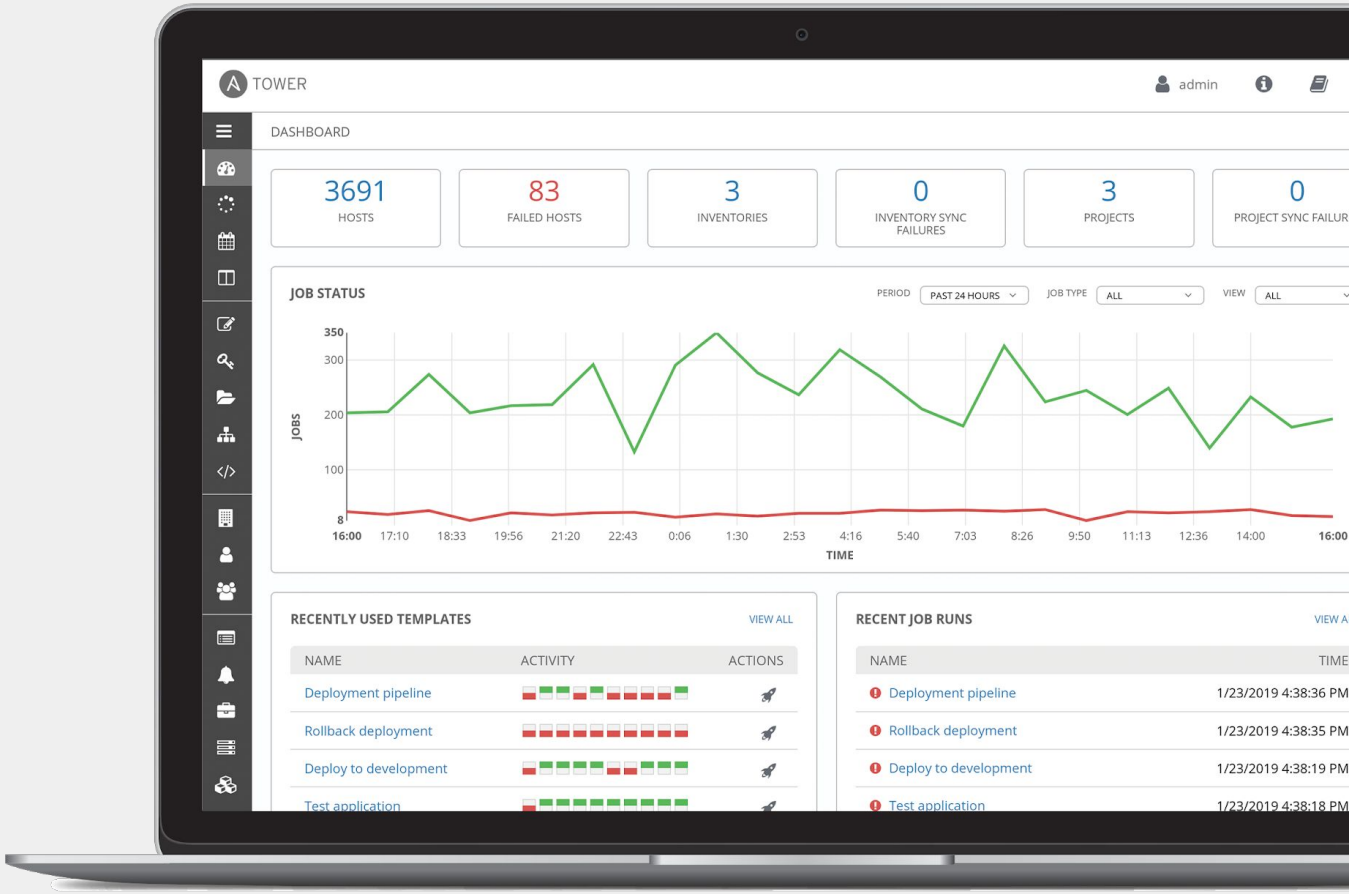
Topics Covered:

- What is Ansible Tower?
- Job Templates
- Inventory
- Credentials

What is Ansible Tower?

Ansible Tower is a UI and RESTful API allowing you to scale IT automation, manage complex deployments and speed productivity.

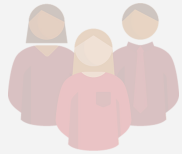
- Role-based access control
- Deploy entire applications with push-button deployment access
- All automations are centrally logged
- Powerful workflows match your IT processes



Red Hat

Red Hat Ansible Automation Platform

Network | Lines of business | Security | Operations | Infrastructure | Developers



Engage

Ansible SaaS: Engage users with an automation focused experience

Scale

Control
Web UI and API

Delegation
Role Based Access Controls

Scale
Scalable Execution Capacity

Create

Ansible Engine: Universal language of automation

Fueled by an open source community

Red Hat

Red Hat Ansible Tower

Push button

An intuitive user interface experience makes it easy for novice users to execute playbooks you allow them access to.

RESTful API

With an API first mentality every feature and function of Tower can be API driven. Allow seamless integration with other tools like ServiceNow and Infoblox.

RBAC

Allow restricting playbook access to authorized users. One team can use playbooks in check mode (read-only) while others have full administrative abilities.

Enterprise integrations

Integrate with enterprise authentication like TACACS+, RADIUS, Azure AD. Setup token authentication with OAuth 2. Setup notifications with PagerDuty, Slack and Twilio.

Centralized logging

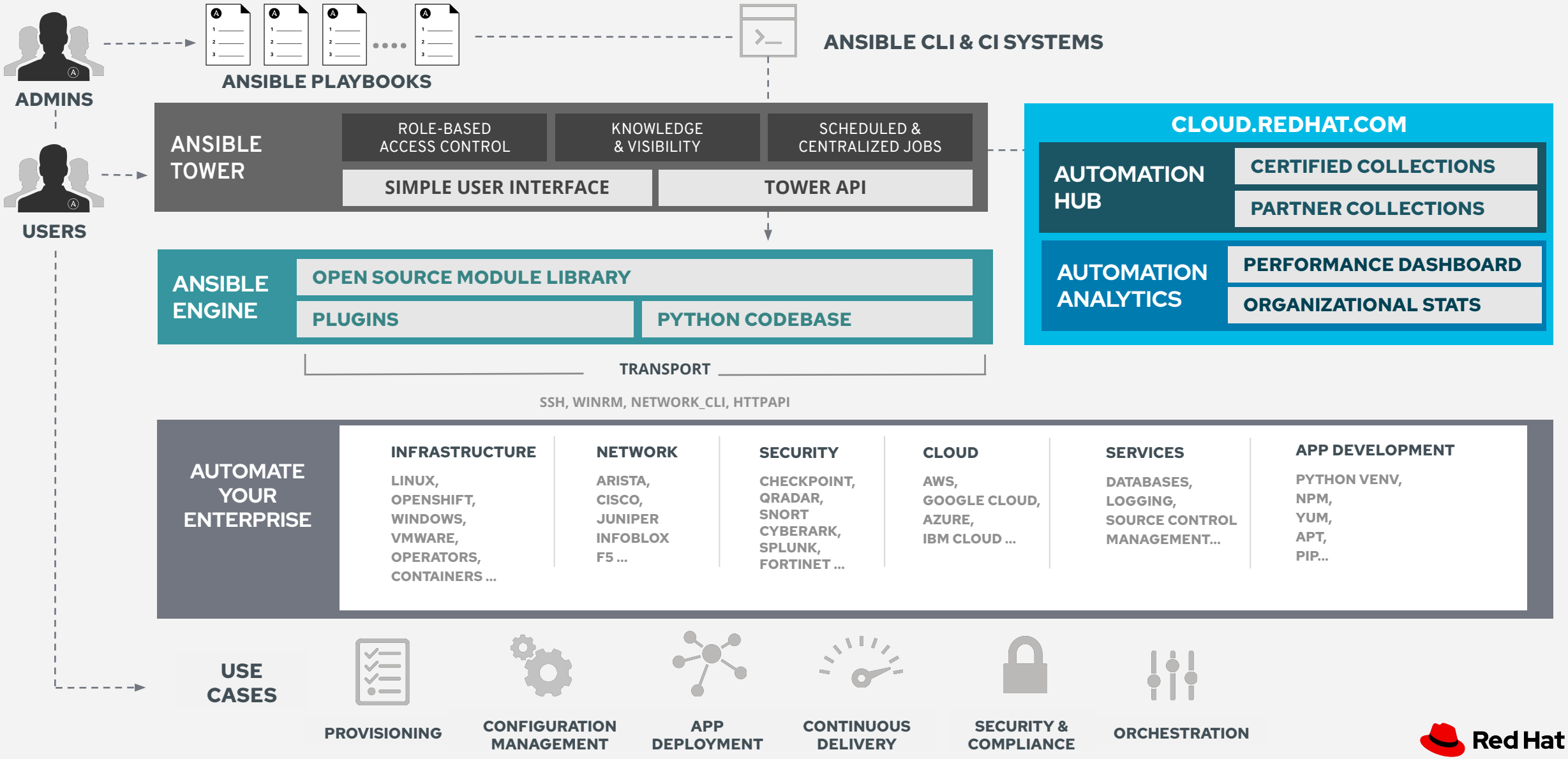
All automation activity is securely logged. Who ran it, how they customized it, what it did, where it happened - all securely stored and viewable later, or exported through Ansible Tower's API.

Workflows

Ansible Tower's multi-playbook workflows chain any number of playbooks, regardless of whether they use different inventories, run as different users, run at once or utilize different credentials.

Red Hat

Ansible Automation Platform



Red Hat

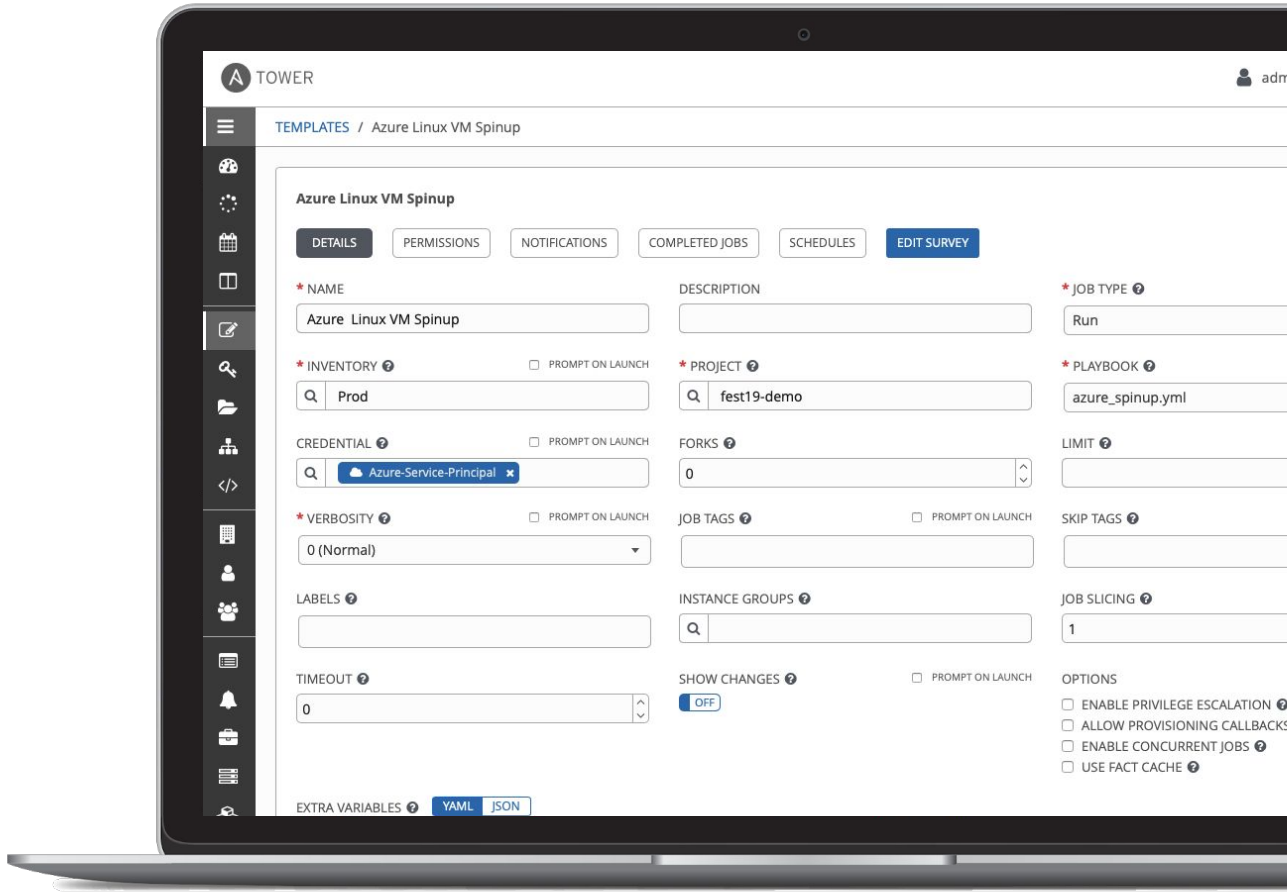
Job Templates

Everything in Ansible Tower revolves around the concept of a **Job Template**. Job Templates allow Ansible Playbooks to be controlled, delegated and scaled for an organization.

Job templates also encourage the reuse of Ansible Playbook content and collaboration between teams.

A **Job Template** requires:

- An **Inventory** to run the job against
- A **Credential** to login to devices.
- A **Project** which contains Ansible Playbooks

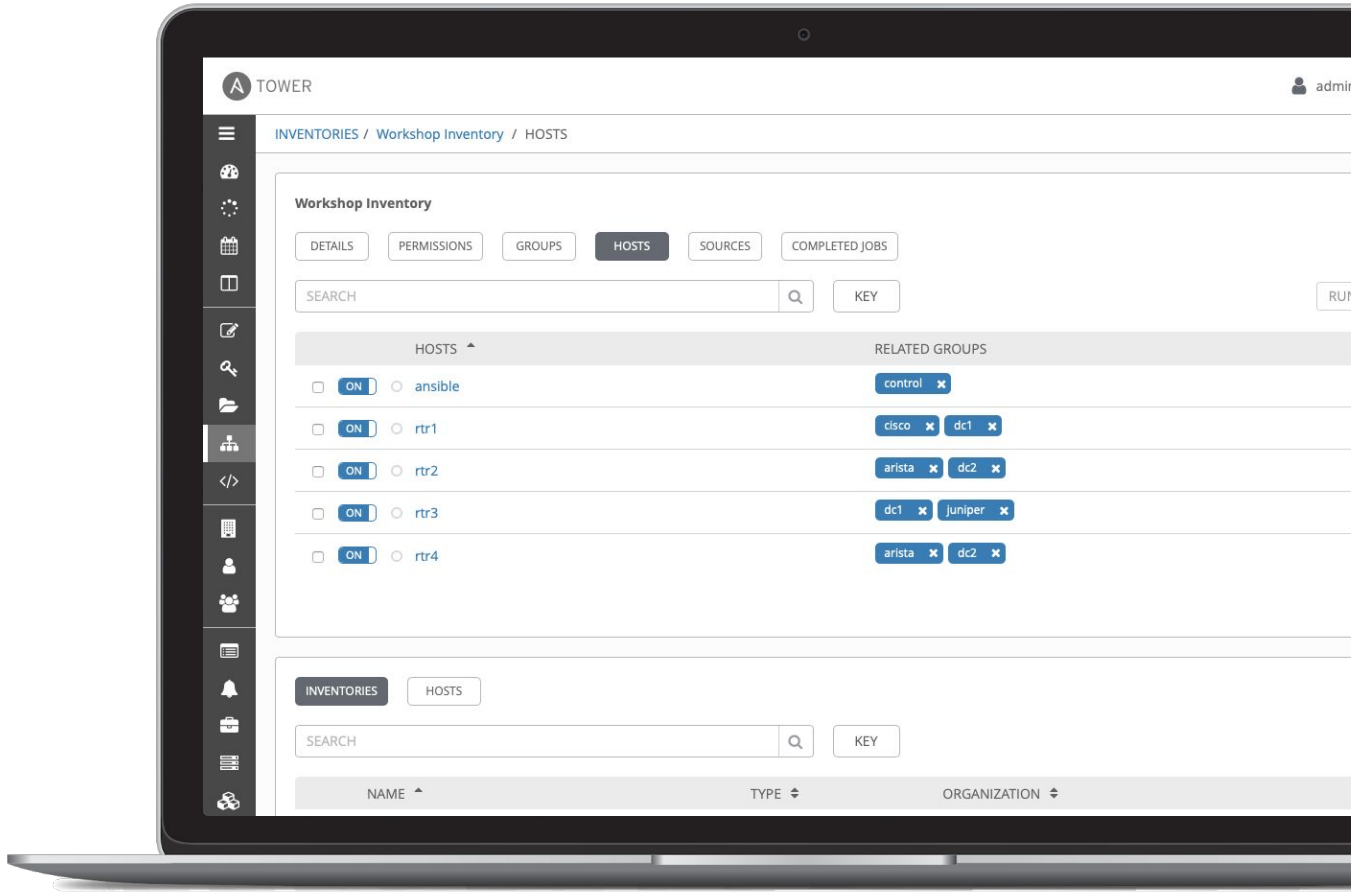


Red Hat

Inventory

Inventory is a collection of hosts (nodes) with associated data and groupings that Ansible Tower can connect to and manage.

- Hosts (nodes)
- Groups
- Inventory-specific data (variables)
- Static or dynamic sources

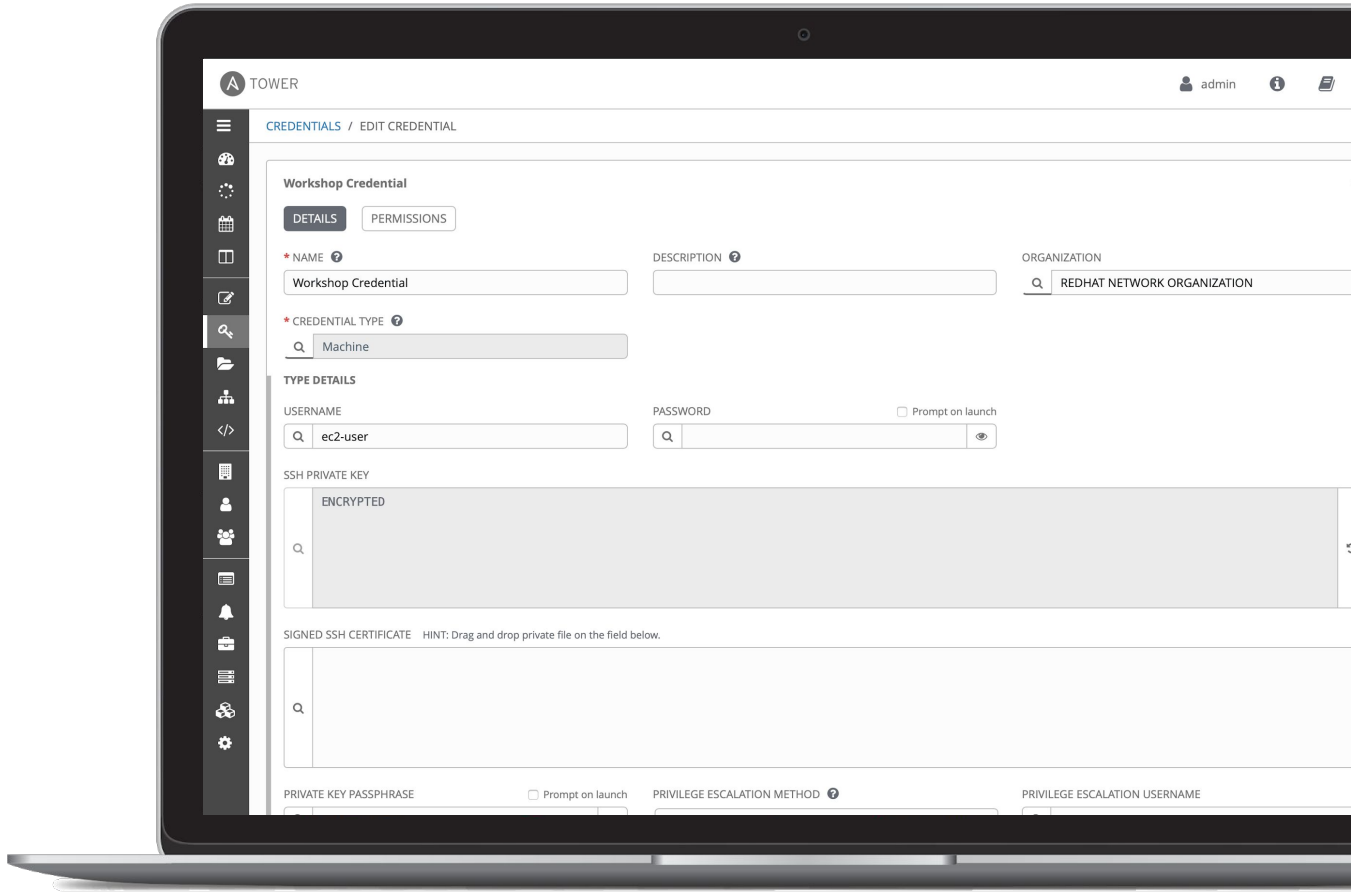


Red Hat

Credentials

Credentials are utilized by Ansible Tower for authentication with various external resources:

- Connecting to remote machines to run jobs
- Syncing with inventory sources
- Importing project content from version control systems
- Connecting to and managing network devices



Centralized management of various credentials allows end users to leverage a secret without ever exposing that secret to them.

Red Hat

Exercise 2.1

Topics Covered:

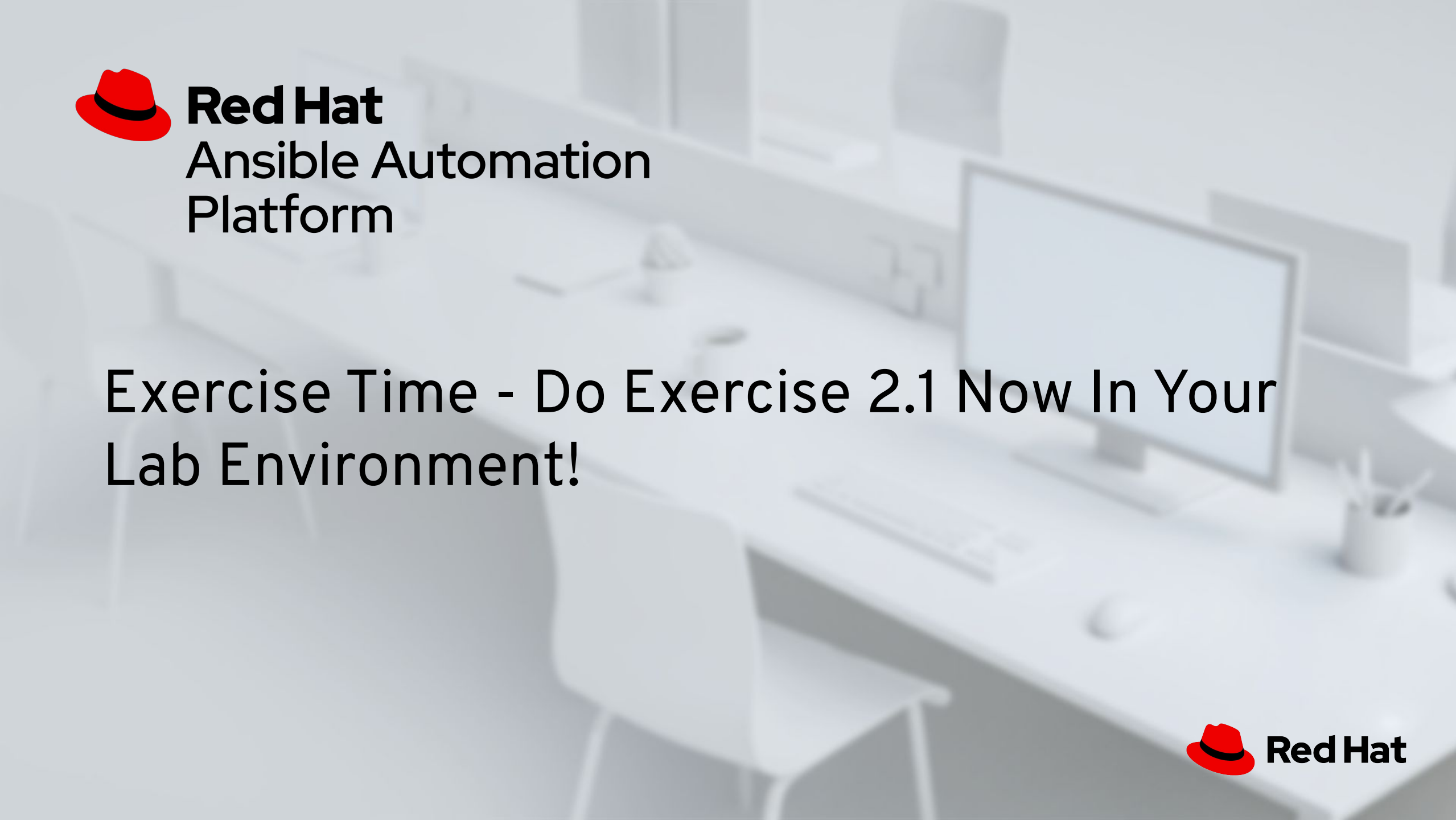
- Investigation Enrichment

Red Hat
Ansible Automation Platform

Red Hat

Persona & Situation

- Persona:
  - Security analyst
  - your main tool: SIEM
- Situation:
  - informed of app anomaly
  - need to figure out if good or bad



Exercise Time - Do Exercise 2.1 Now In Your Lab Environment!

Exercise 2.2

Topics Covered:

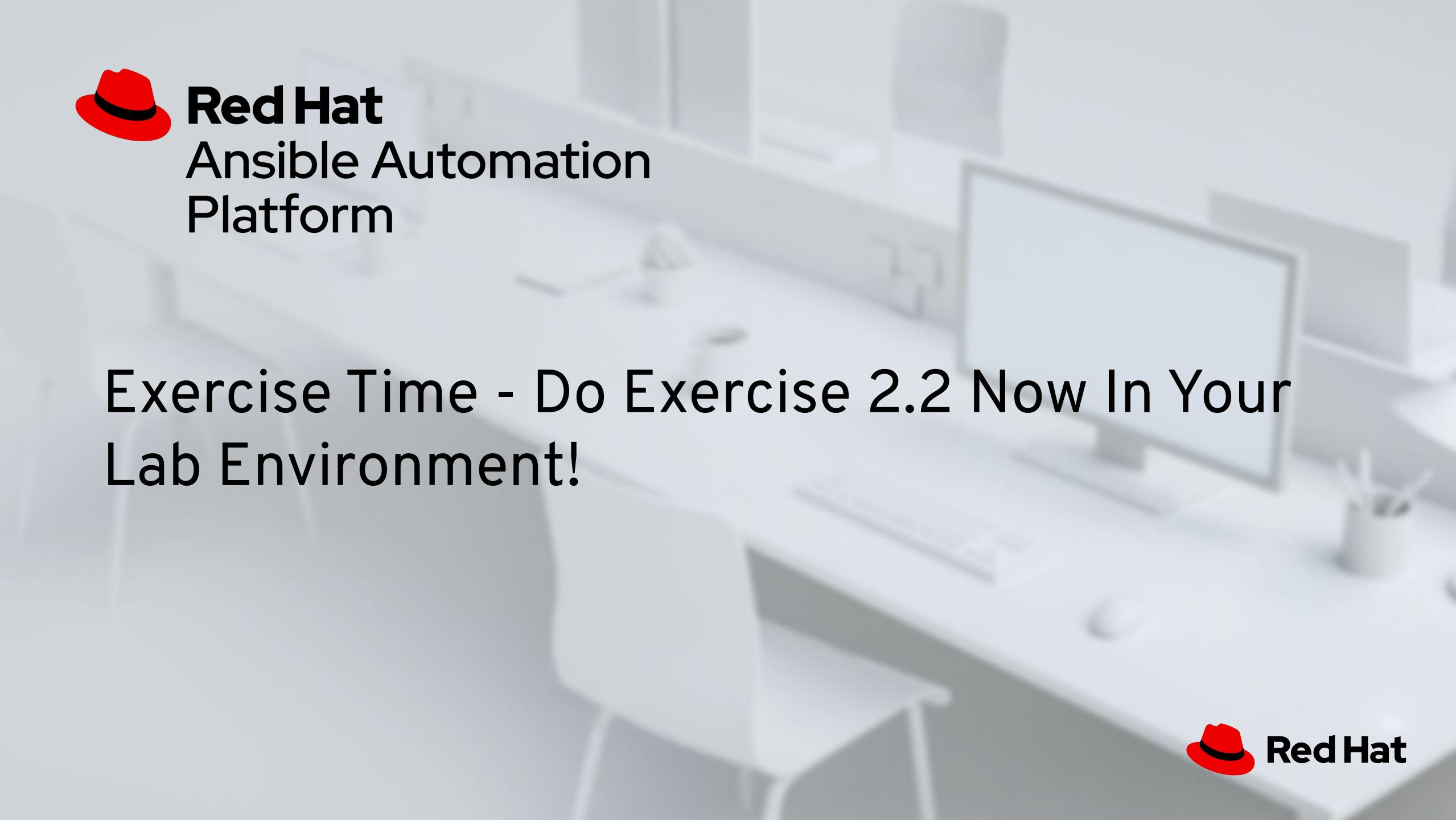
- Threat hunting
- How Tower helps bringing together the automation of different teams

Persona & Situation

- Persona:
  - Security operator
  - your main tool: Firewall
- Situation:
  - suspicious traffic hitting the FW
  - decide to whitelist or not
  - interactions between different teams via Ansible Tower

Tower

- Already installed
- Pre-populated with inventories, teams, users, job templates and so on
- Will be used by different personas during different steps
- Used to highlight how different IT teams can work together, how RBAC can help providing access to automation without losing control of the environment



Exercise Time - Do Exercise 2.2 Now In Your Lab Environment!

Red Hat

Red Hat
Ansible Automation
Platform

Exercise 2.3

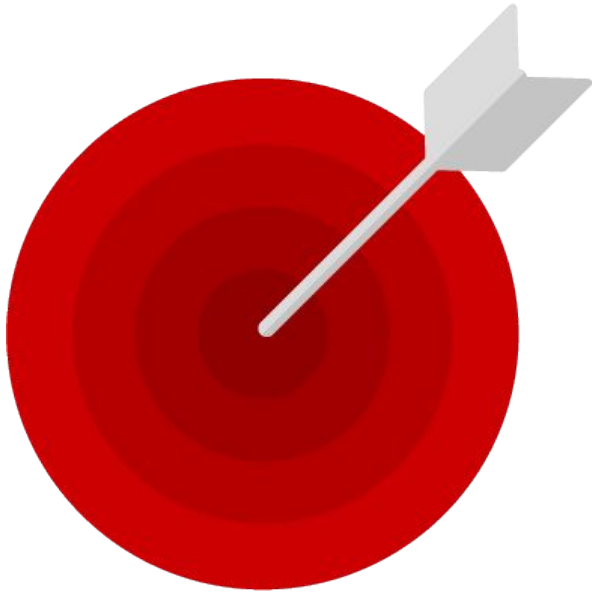
Topics Covered:

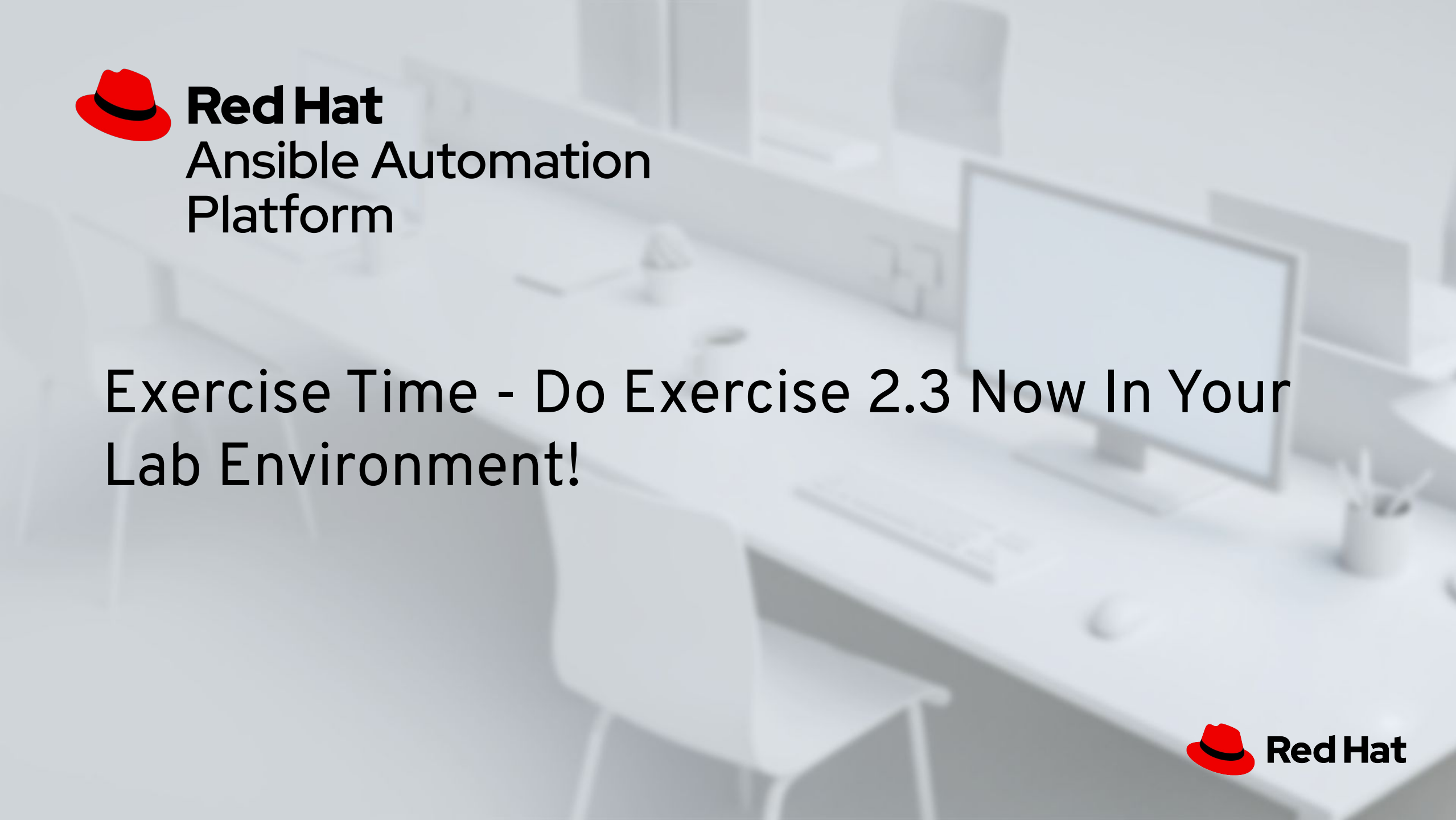
- Incident response

Red Hat

Persona & Situation

- Persona:
  - Security operator
  - your main tool: IDS
- Situation:
  - you see IDS warnings
  - create marker, blacklist





Exercise Time - Do Exercise 2.3 Now In Your Lab Environment!

Exercise 2.4

Topics Covered:

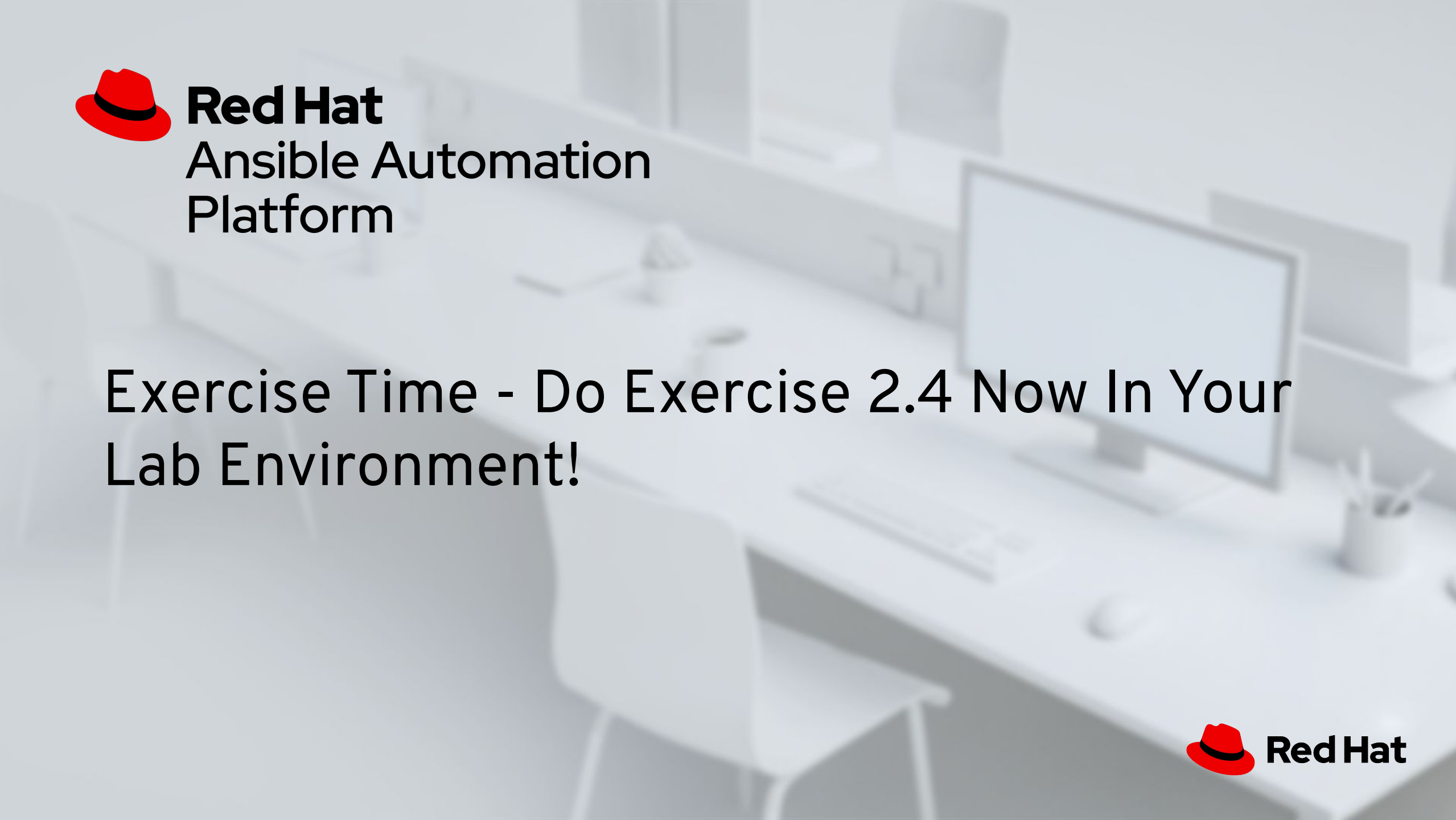
- Wrap it all up

Red Hat
Ansible Automation
Platform

Red Hat

You Are Done!

You finished the workshop! Just read the final words, and you can soon apply your new knowledge on your own environments!



Exercise Time - Do Exercise 2.4 Now In Your Lab Environment!

Red Hat

AnsibleFest

October 13-14, 2020 | Virtual Experience

Red Hat

Thank you

linkedin.com/company/red-hat

youtube.com/user/RedHatVideos

facebook.com/ansibleautomation

twitter.com/ansible

github.com/ansible

Red Hat