Red Hat
Ansible Automation
Platform

# Ansible Security Workshop

**Microlearning Edition**

Red Hat

# Day 1

- Administration
- Know each other
- What you will learn
- Intro to Ansible
- Exercises 1.1, 1.2



Red Hat
Ansible Automation
Platform

Red Hat

# Administration

- Explain the format
- Clarify group rules (if any)
- Resources (exercises, decks, scripts, etc.)
- Feedback form
- Lab

# Know each other

- Introduce yourself
- Your experience with Automation and Ansible
- Your goals & expectations
- Your use case
- Your position

# 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

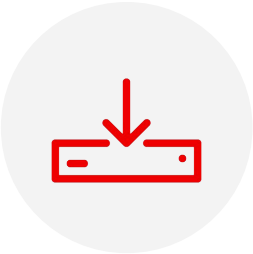
Red Hat

# INTRODUCTION TO ANSIBLE



Red Hat

# Growth by the numbers

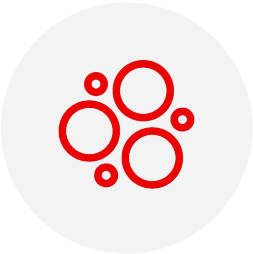


**2M**

downloads per month

**4K**

modules



**7th**

of 96M projects on GitHub by contributors

**4M+**

systems managed by Red Hat



Source:

# Why Ansible?

## Simple

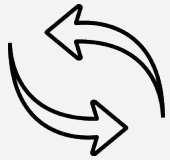
Human readable automation

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

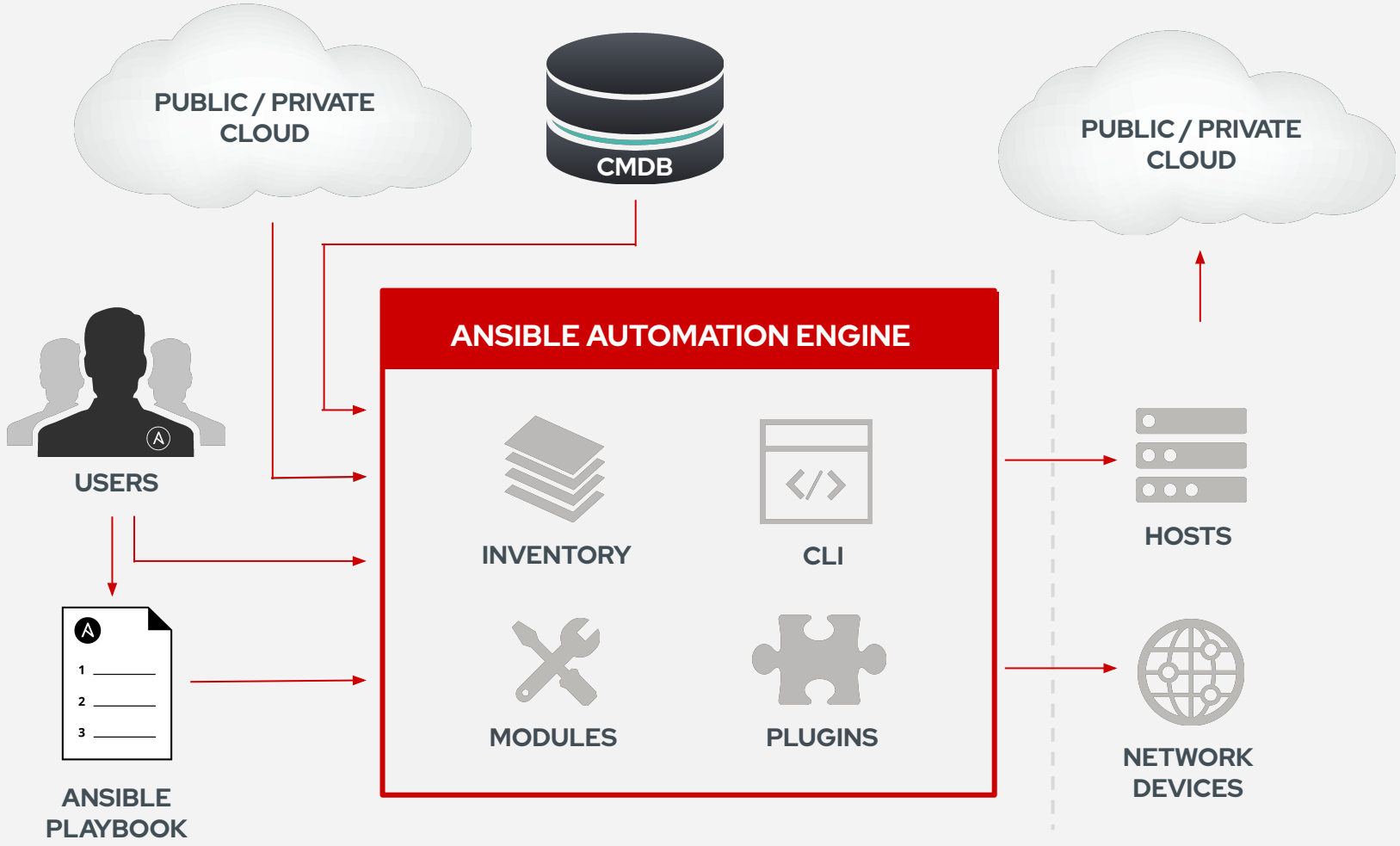
Uses OpenSSH & WinRM

No agents to exploit or update

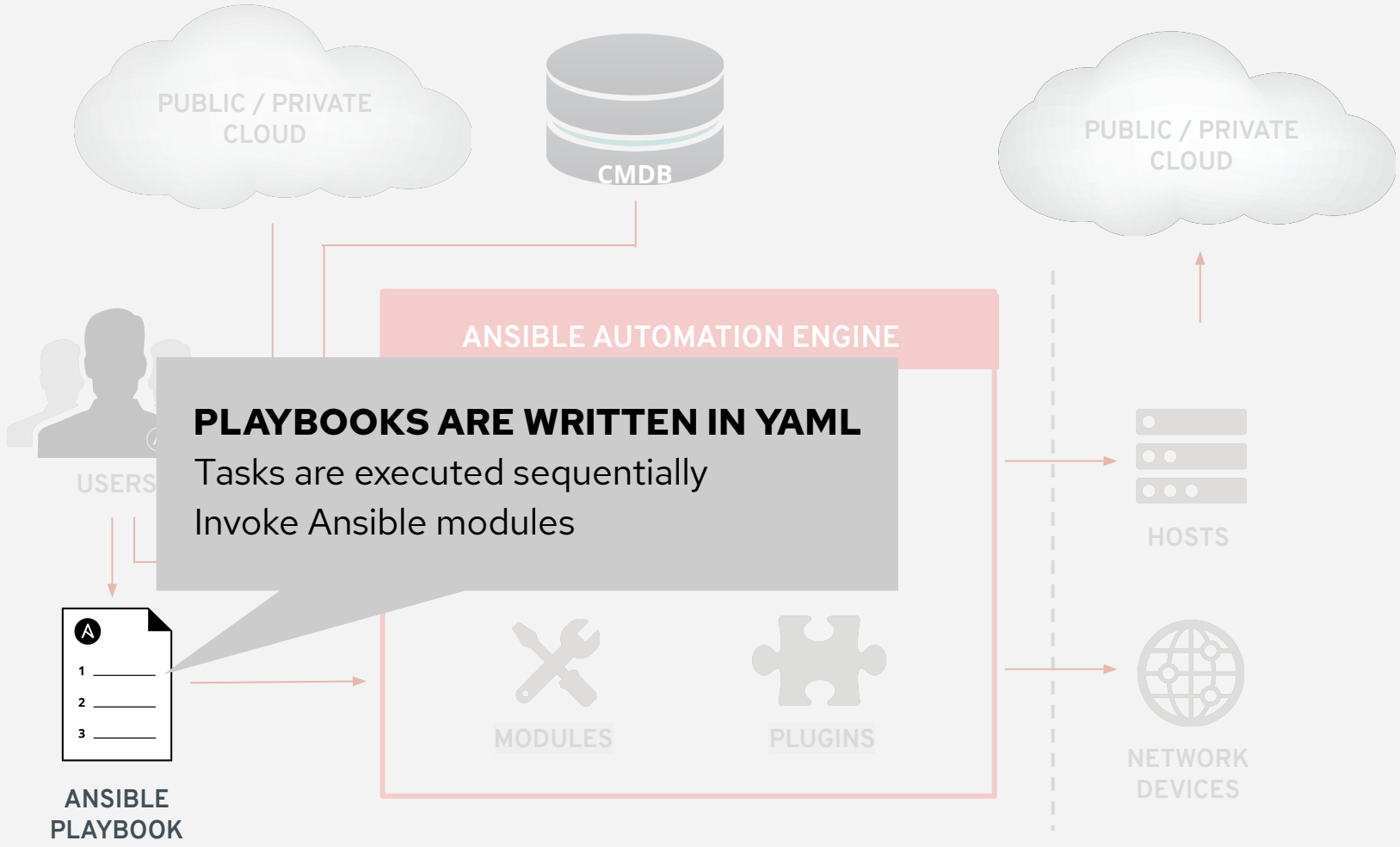
Get started immediately

**More efficient & more secure**

Red Hat


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

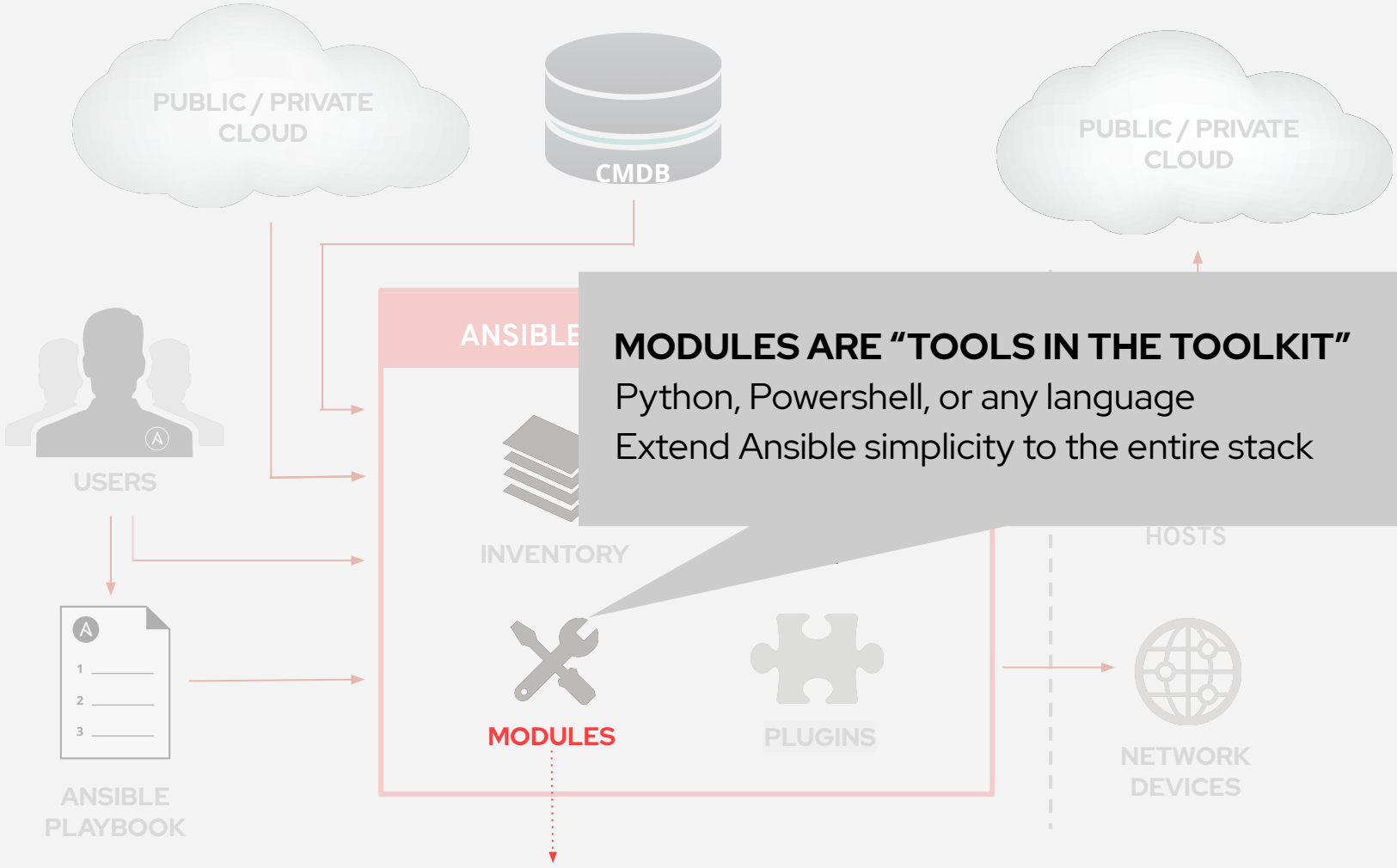
Red Hat

```
---
- 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
```



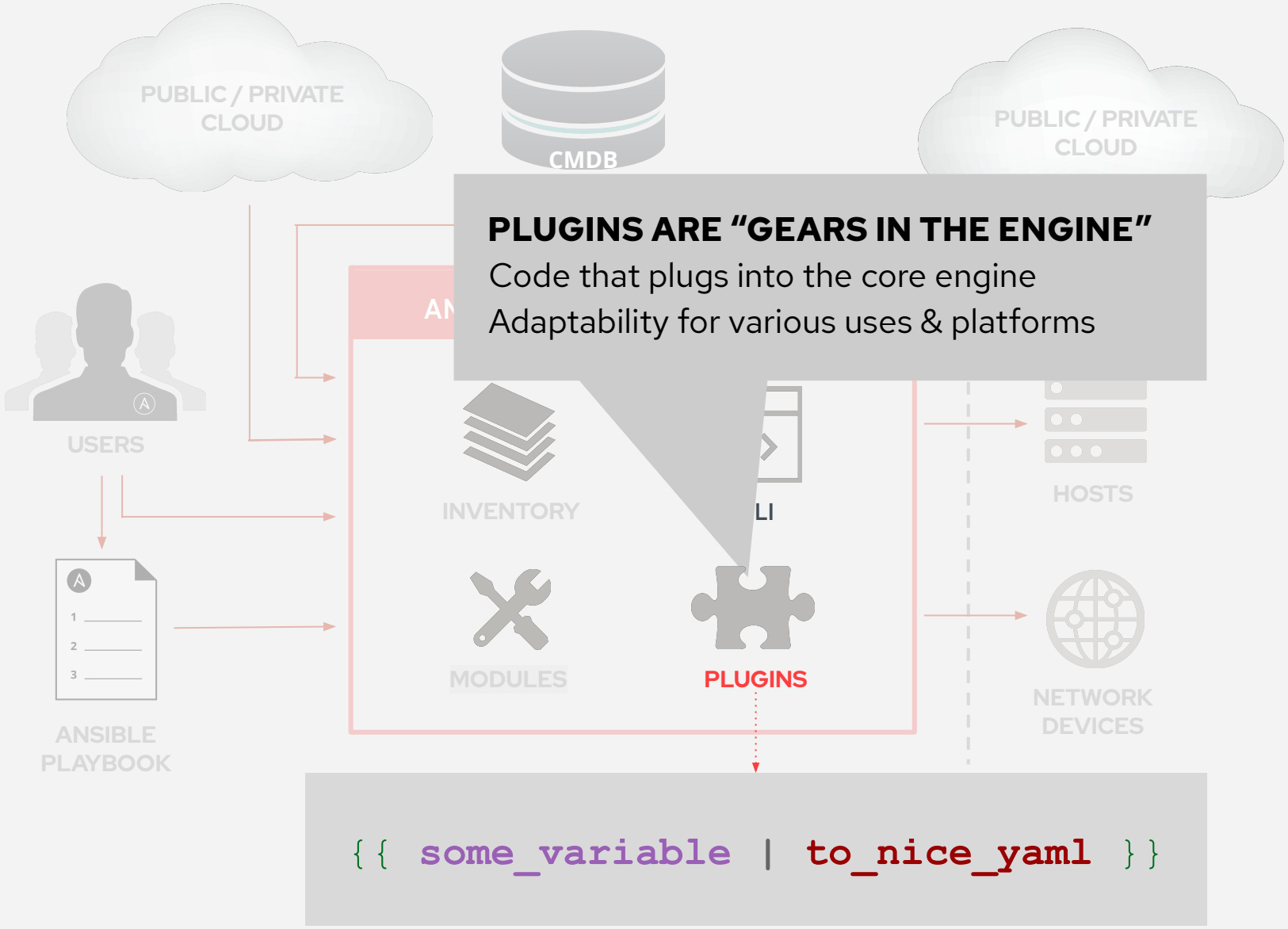

**MODULES ARE "TOOLS IN THE TOOLKIT"**

Python, Powershell, or any language

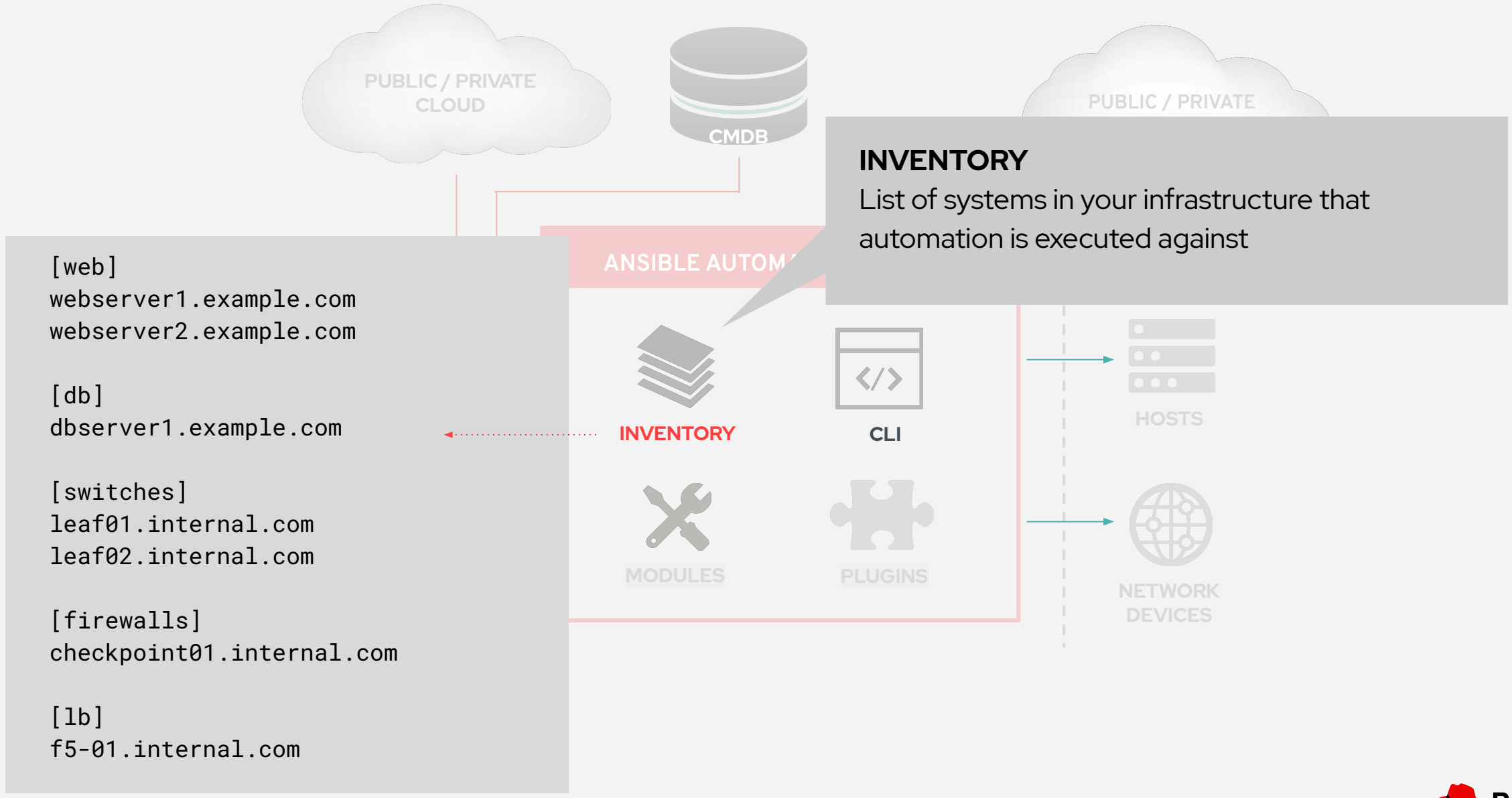
Extend Ansible simplicity to the entire stack

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

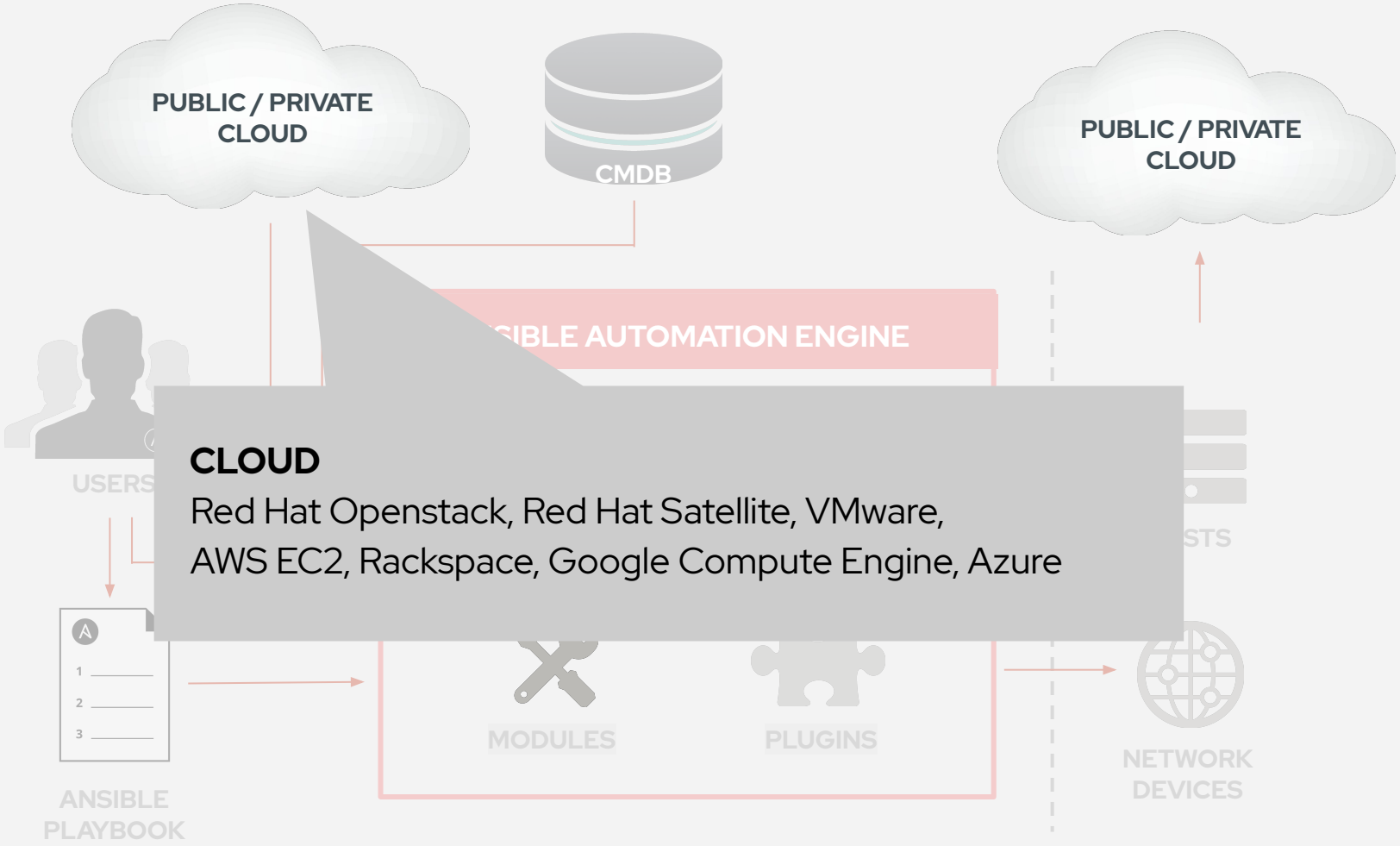
Red Hat


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

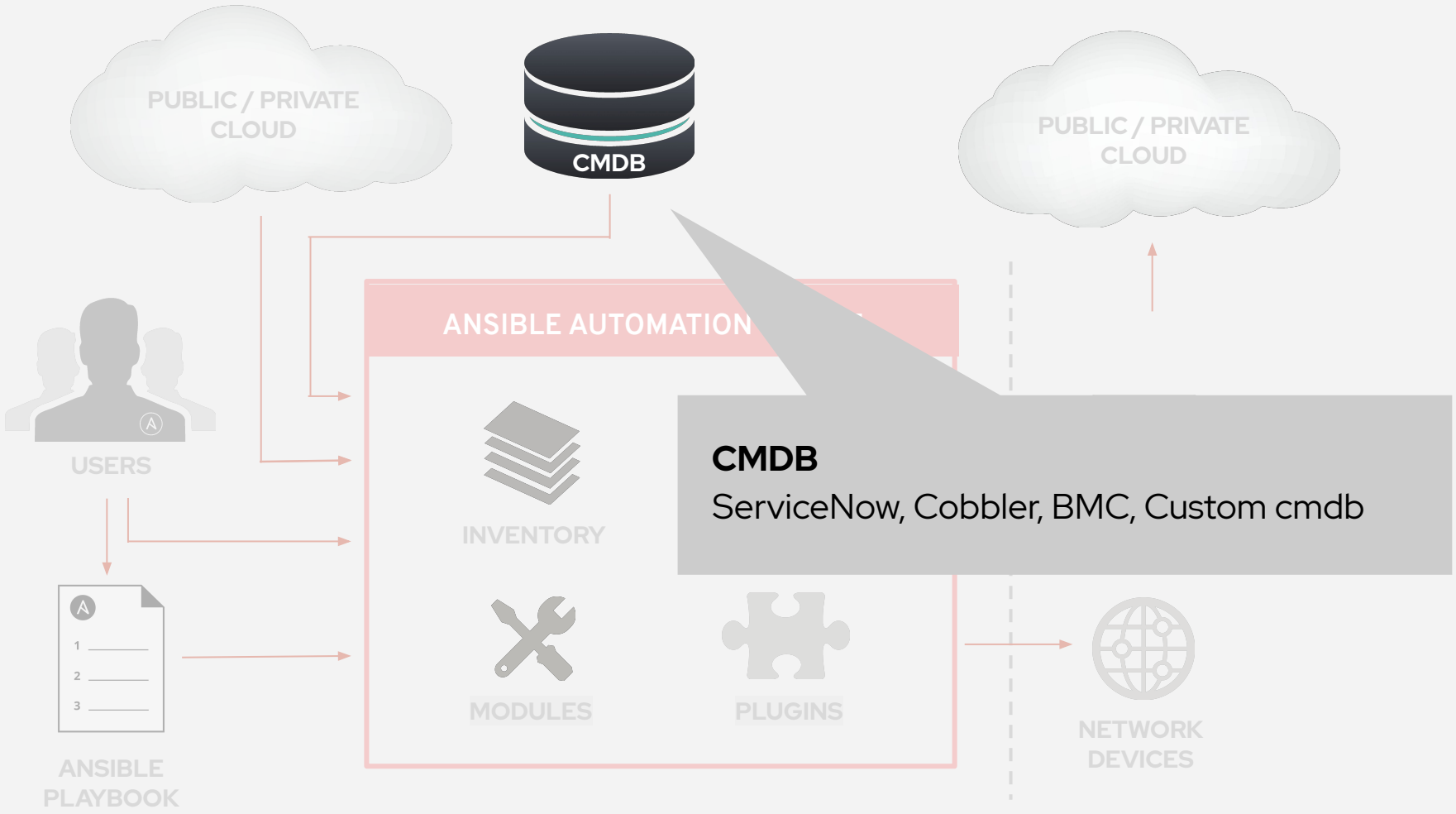
Red Hat



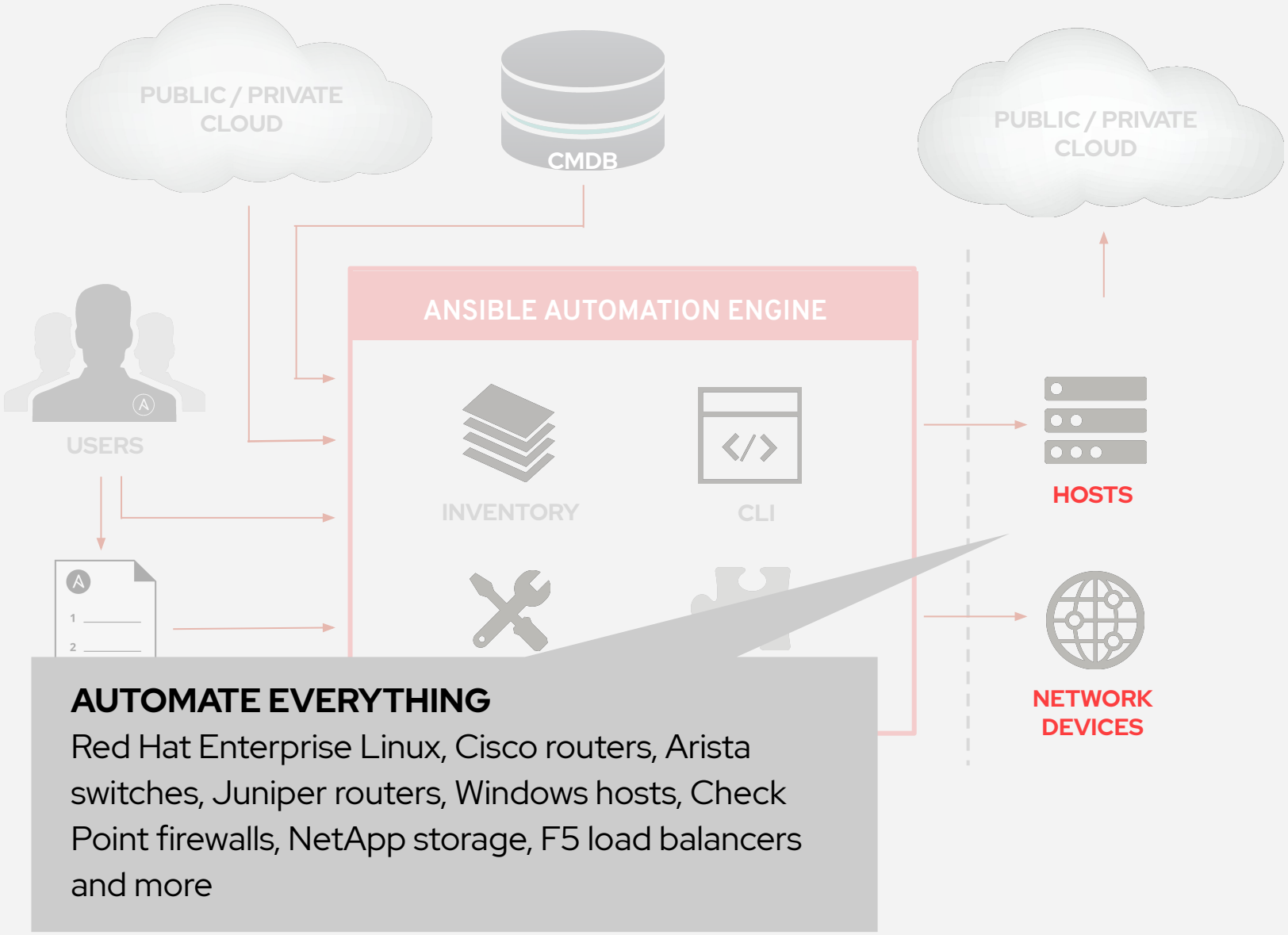

**CLOUD**

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

Red Hat
Ansible Automation
Platform

# Exercise 1.1

Topics Covered:

- What Ansible Security Automation is about
- The lab infrastructure



# 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.

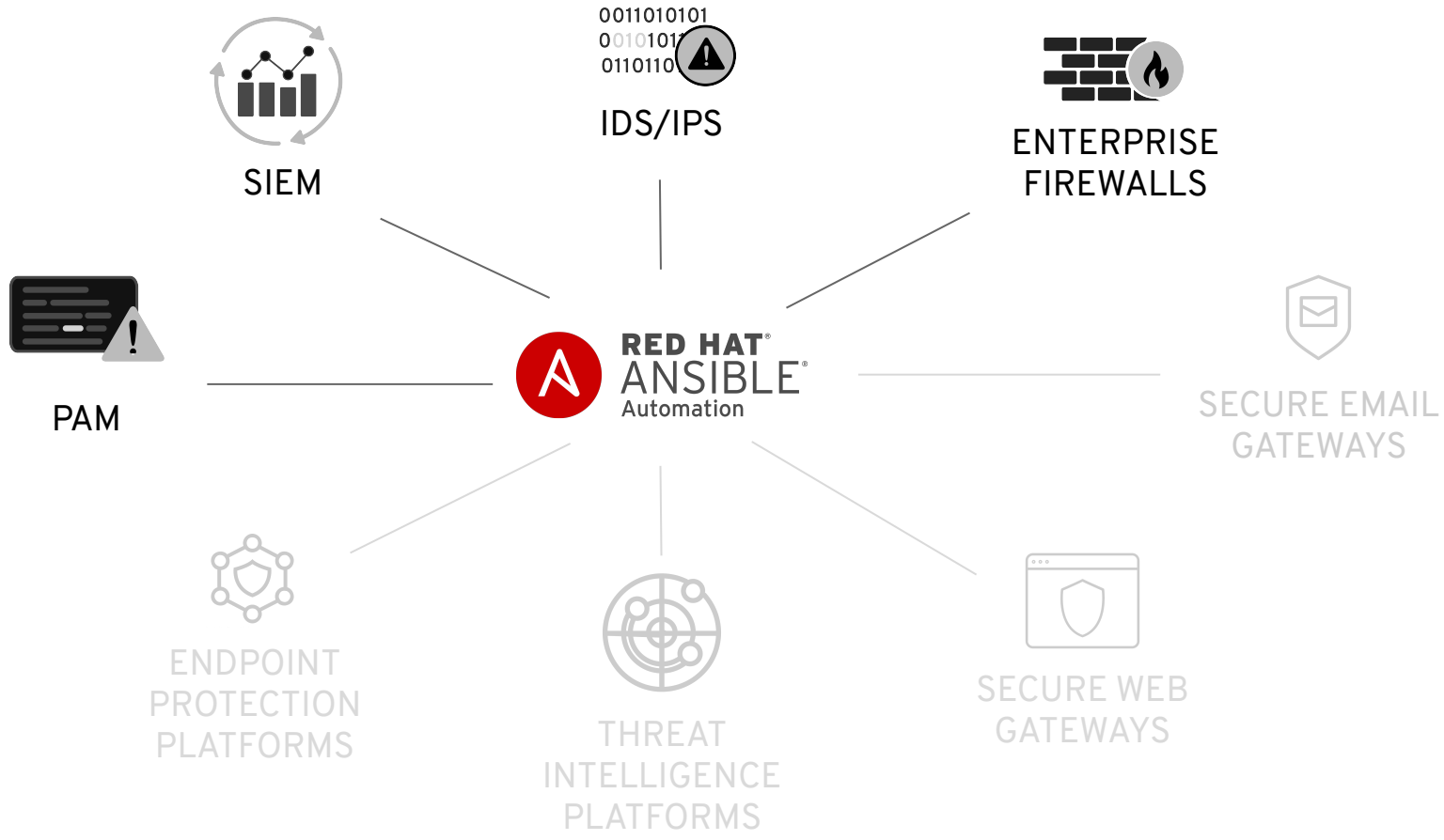
# Is It A Security Solution?

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

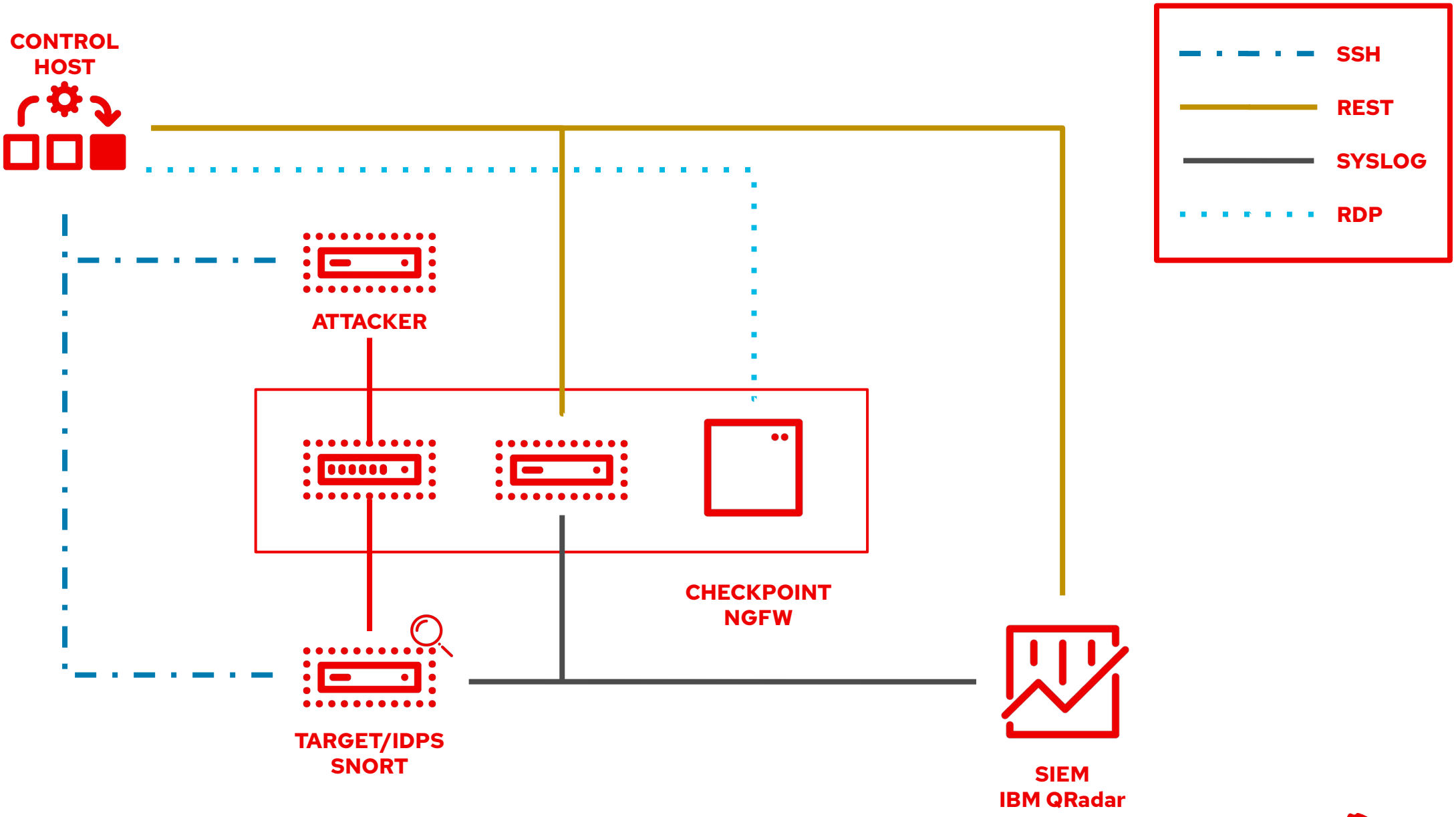
- Follow the steps to access environment
- Use the IP provided to you, the script only has example IPs
- Access to machines is done via online editor with a built-in terminal

# 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

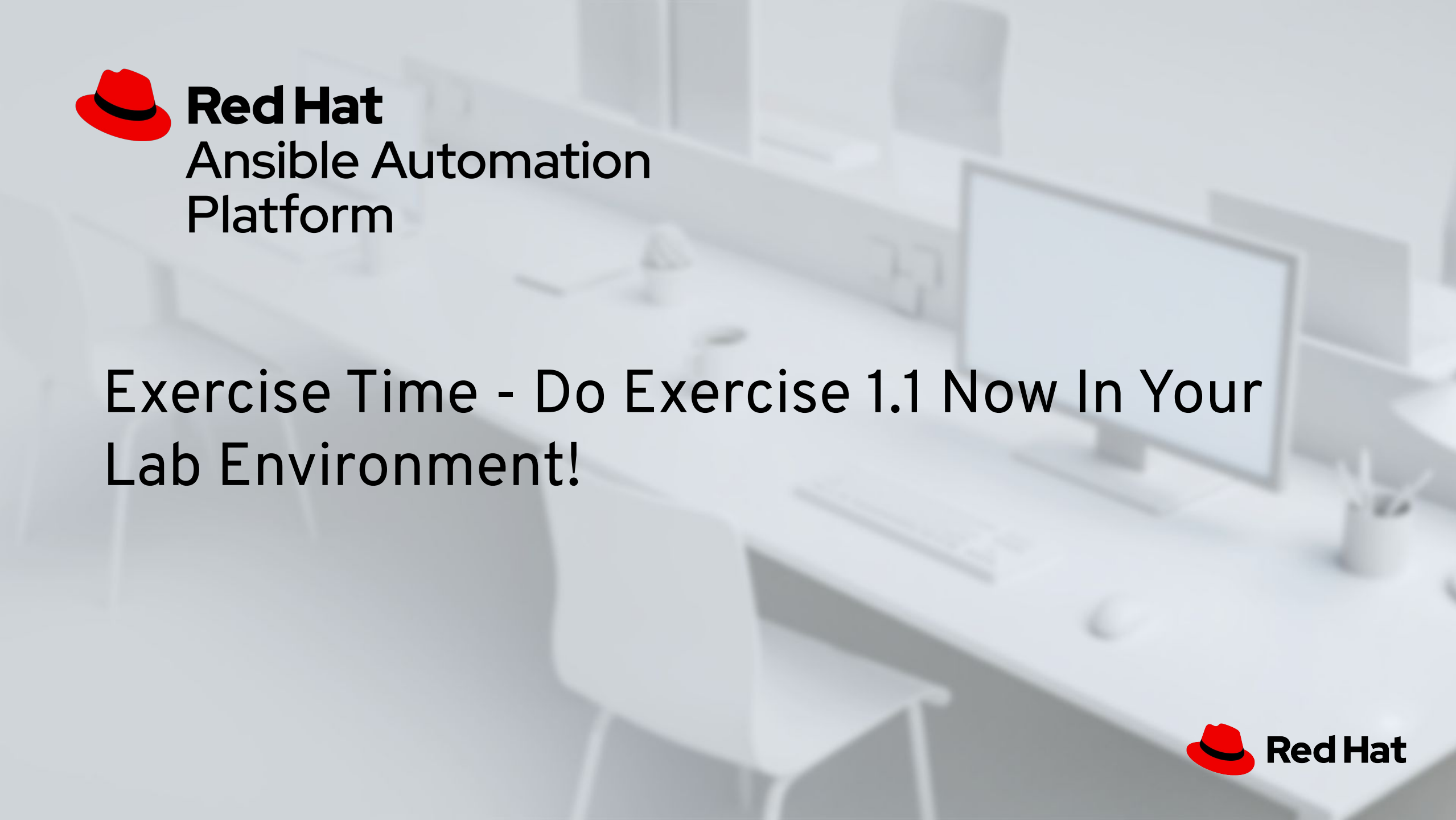
```
[all:vars]
ansible_user=student1
ansible_ssh_pass=ansible
ansible_port=22

[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

# Exercise 1.2

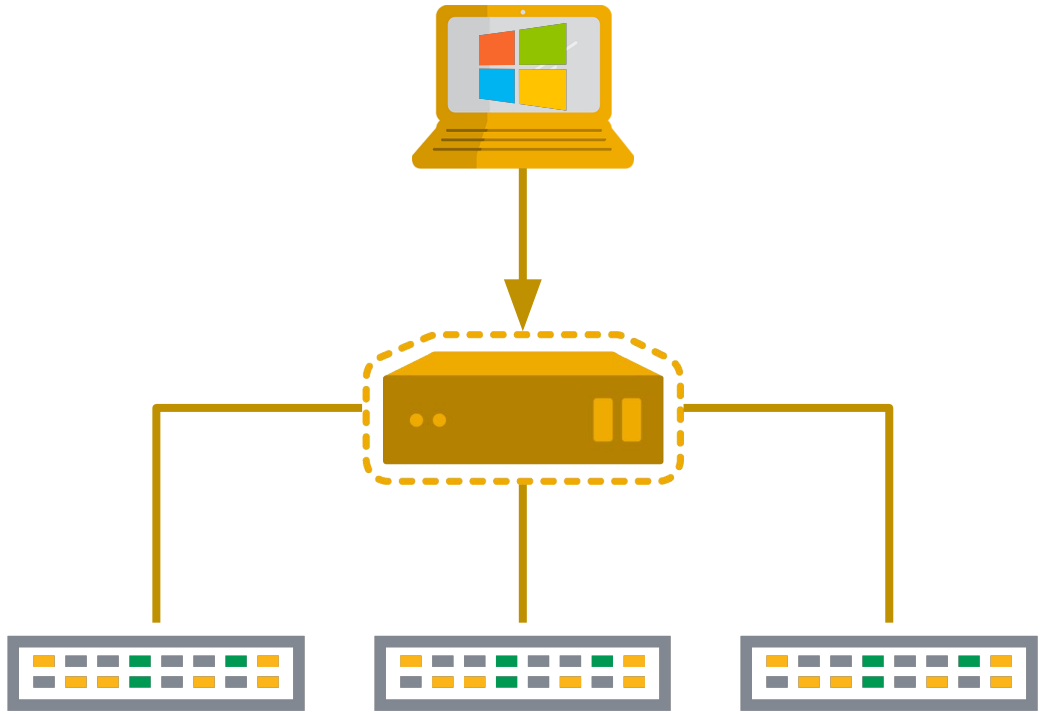
Topics Covered:

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

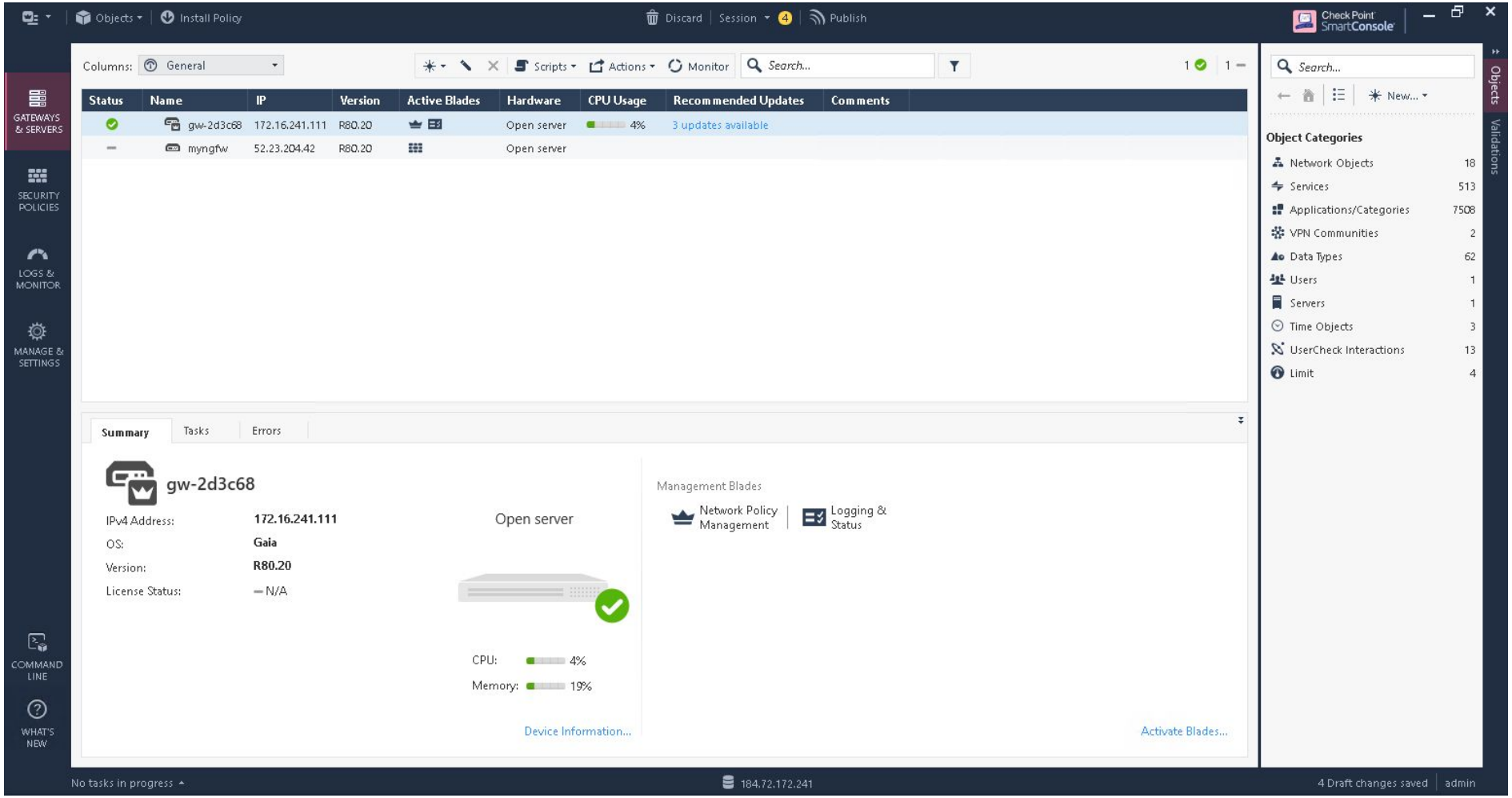
# Accessing And Managing
# Check Point Next Generation Firewalls

- 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

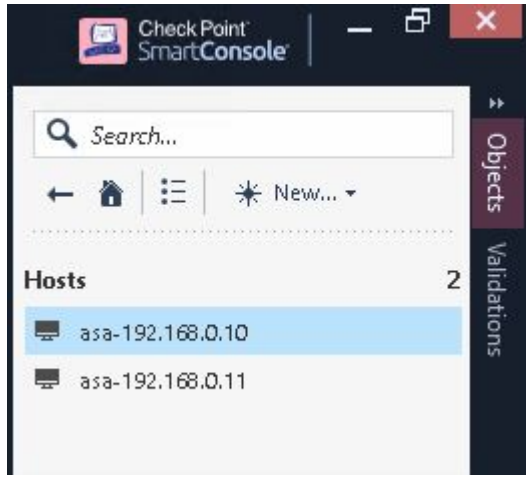
```
A task executed as expected, no change was made.

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

Red Hat
Ansible Automation
Platform

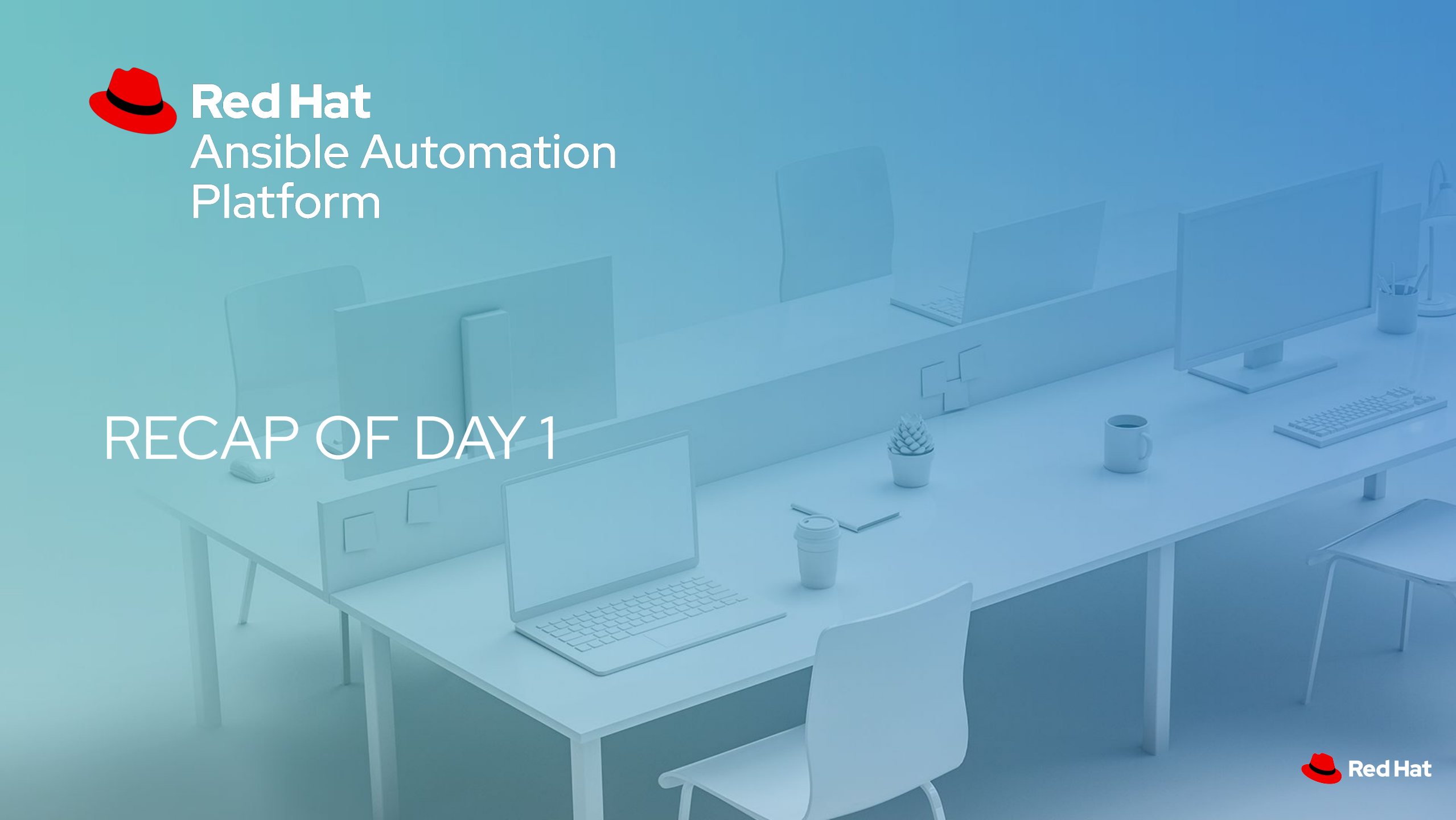
# Day 2:

Topics Covered:

- Recap of Day 1
- Snort rules
- Running a playbook interacting with Snort
- Exercise 1.3



Red Hat



Red Hat
Ansible Automation
Platform

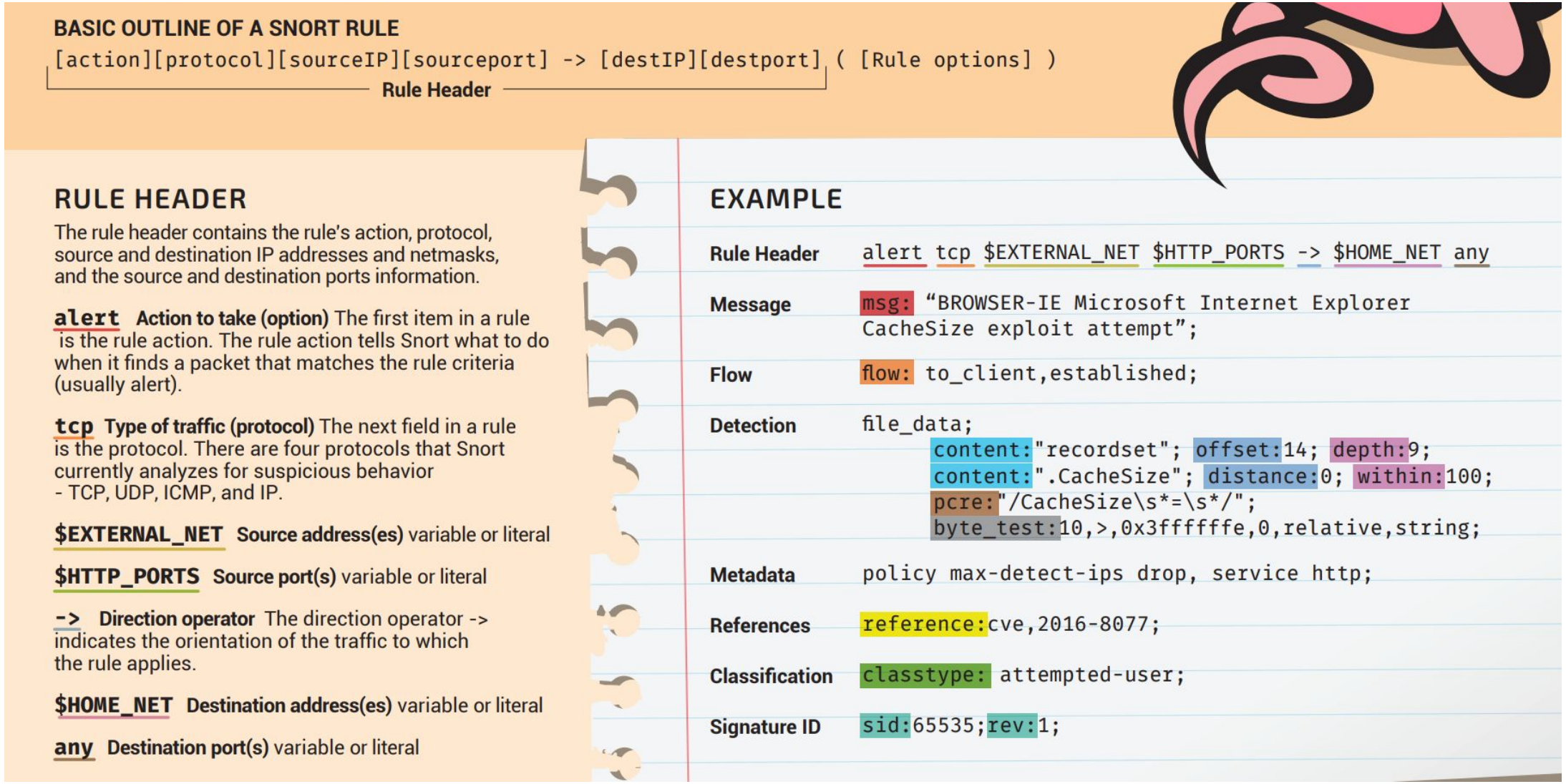
# RECAP OF DAY 1

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






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

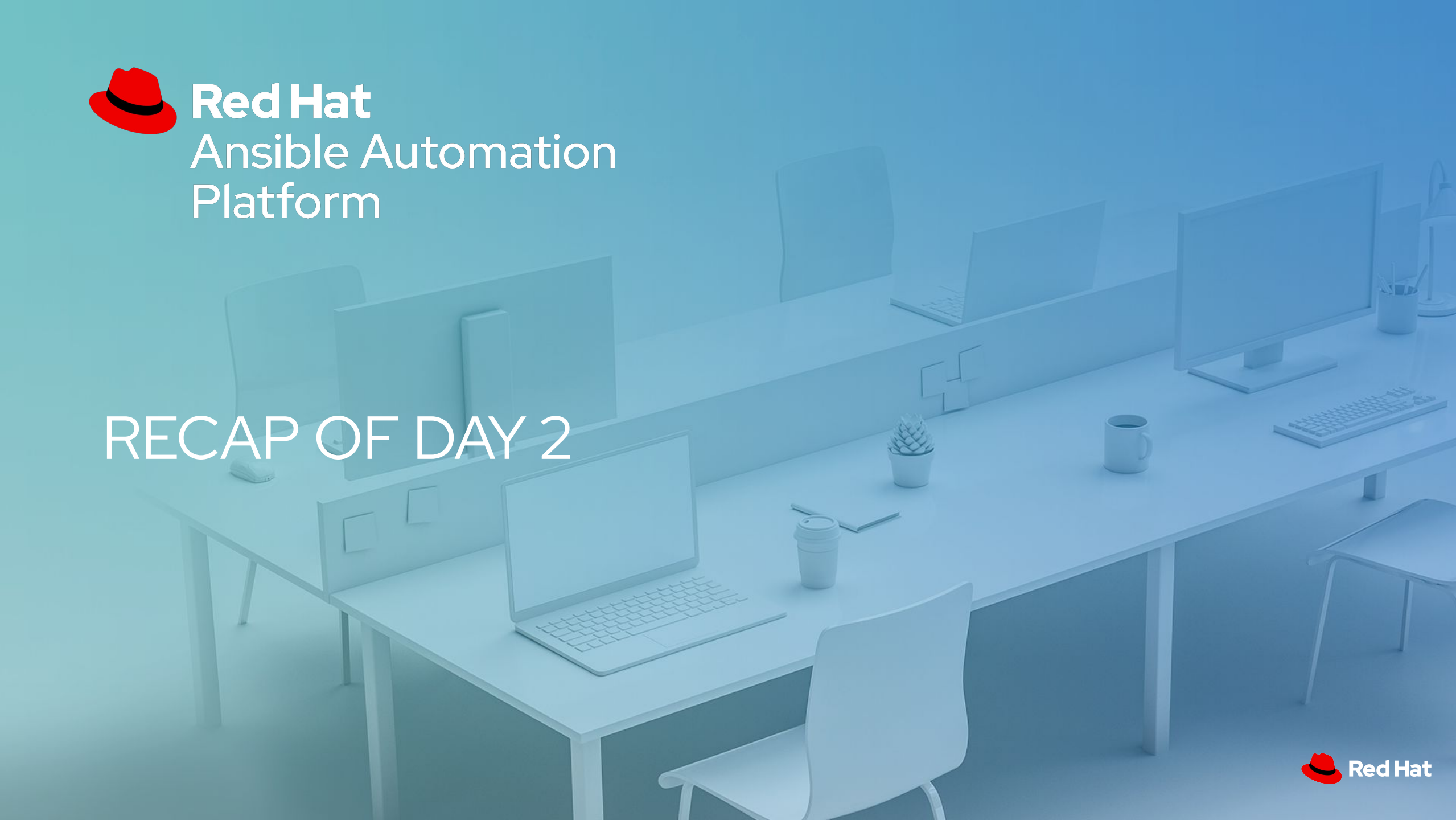
```
$ ansible-galaxy install ansible_security.acl_manager
```


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

# Day 3:

Topics Covered:

- Understanding QRadar
- Collections
- Exercise 1.4, 2.1



Red Hat
Ansible Automation
Platform

# RECAP OF DAY 2

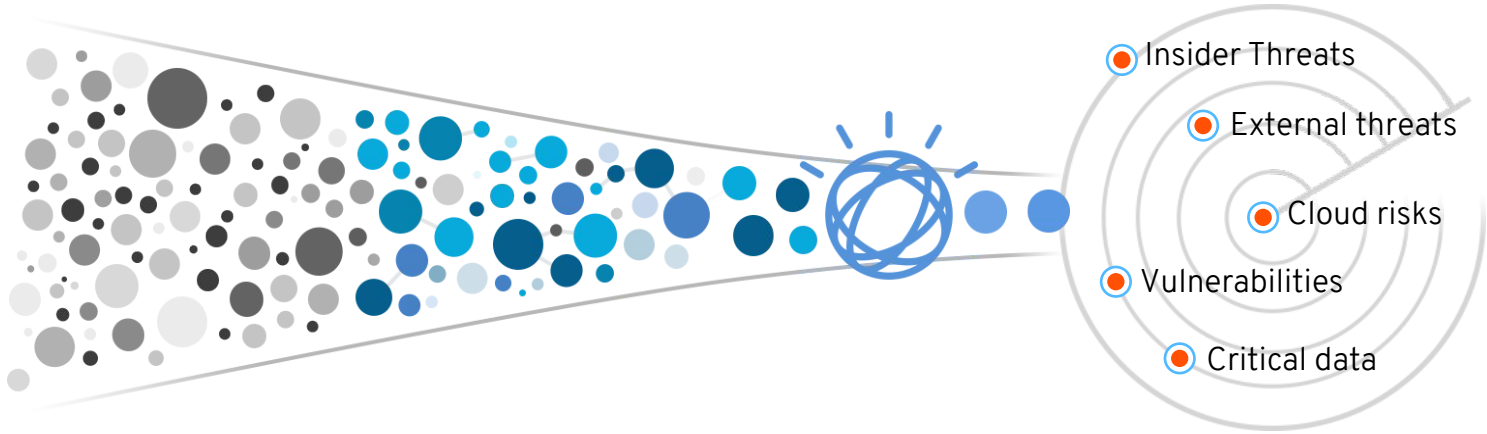
Red Hat

# 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



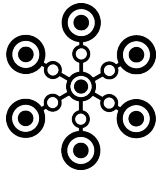



Red Hat

# IBM QRadar: Automate Intelligence

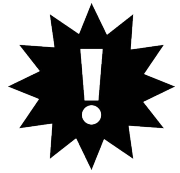
## Detect

Known and unknown threats



## Connect

Related activity in multi-stage attacks



## Prioritize

Business critical events

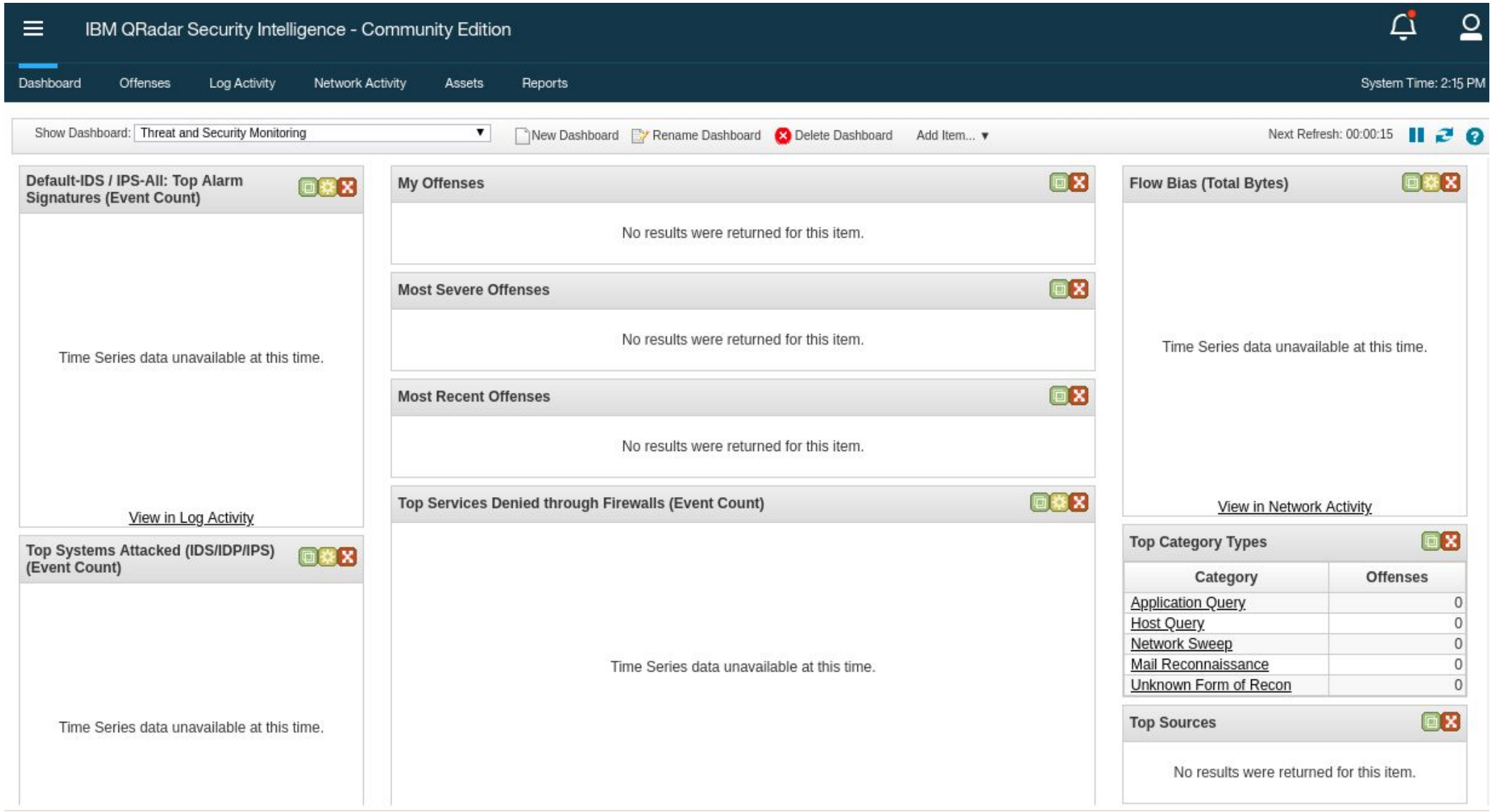
## Investigate

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

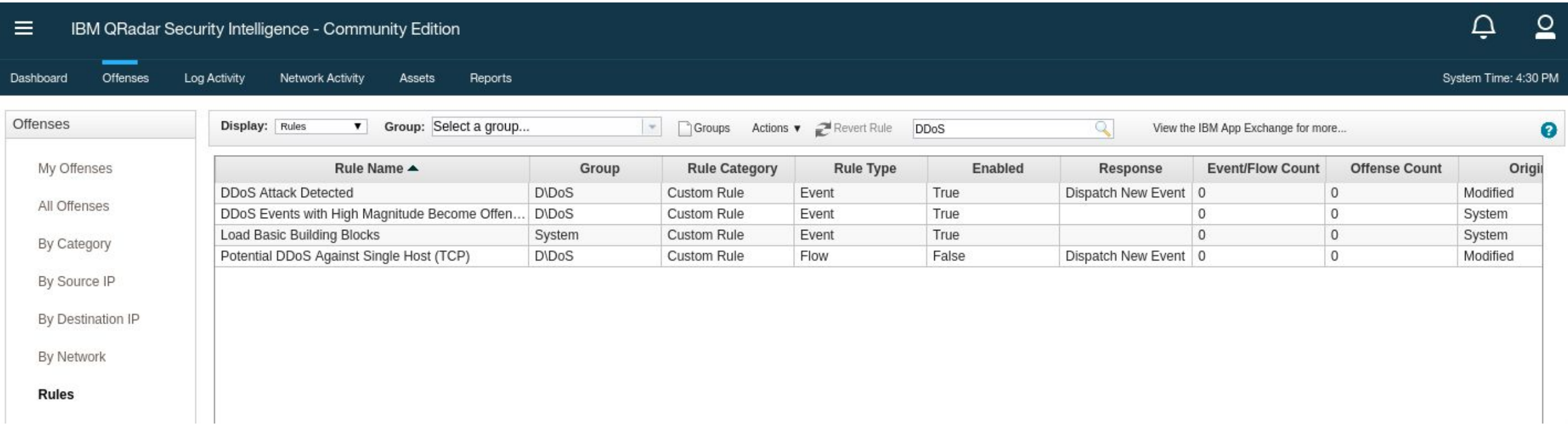
# QRadar






Red Hat

# Verification In The UI



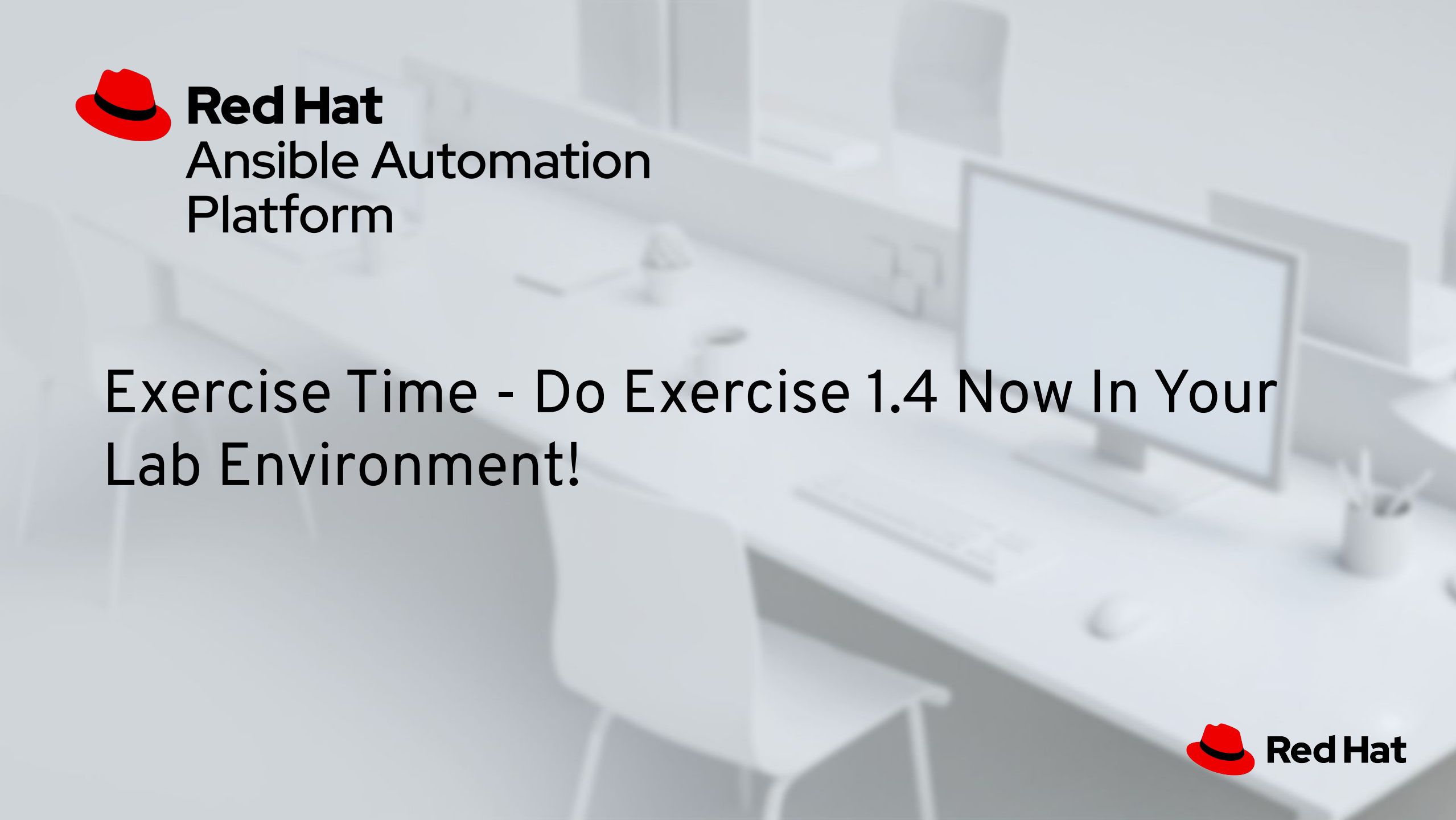



Red Hat

# Collections

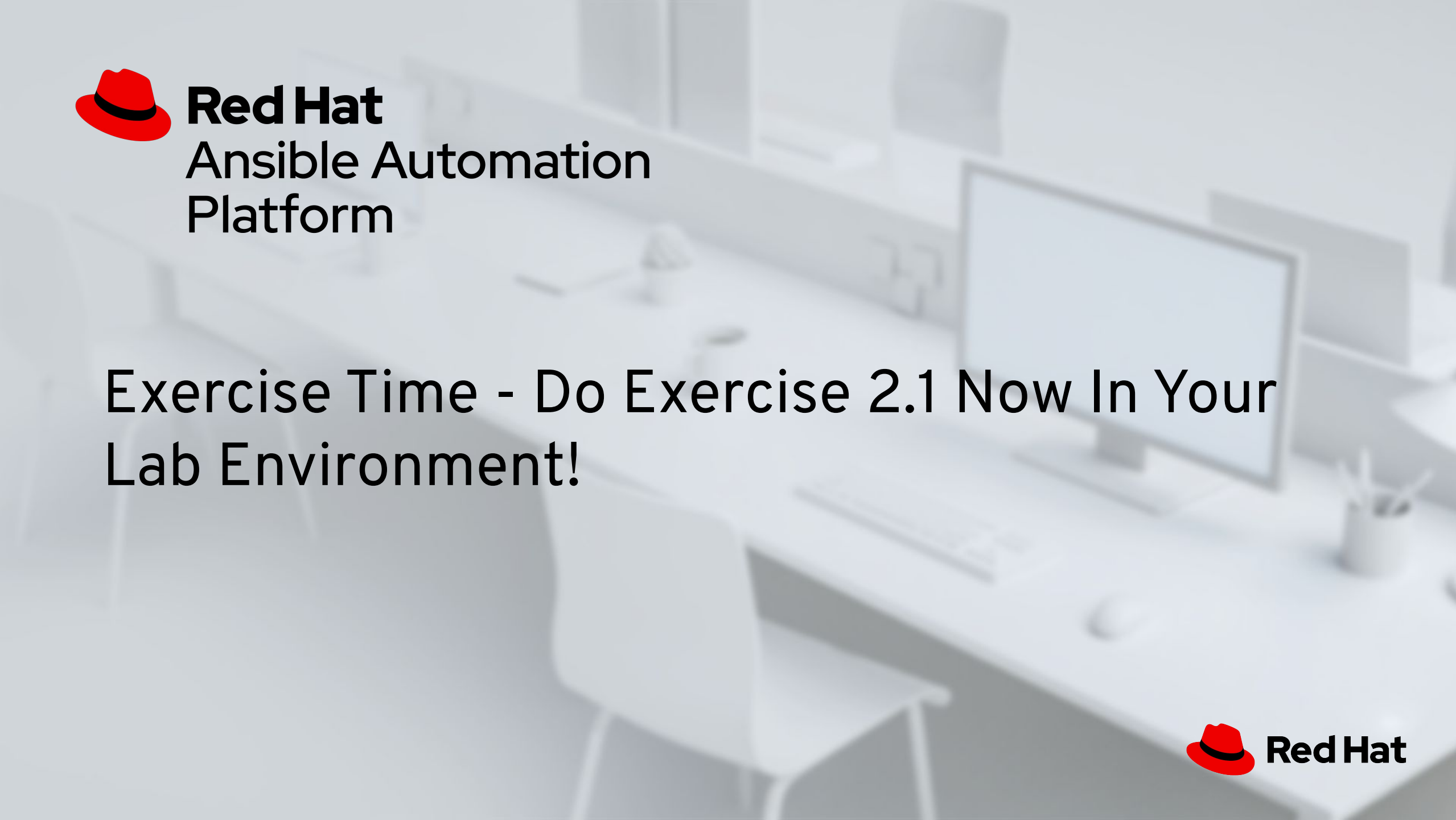
- Ansible content to interact with QRadar: provided as collections
- Like roles, but even more powerful
- Can also contain modules, connection plugins and so on

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


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

# 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!

Red Hat

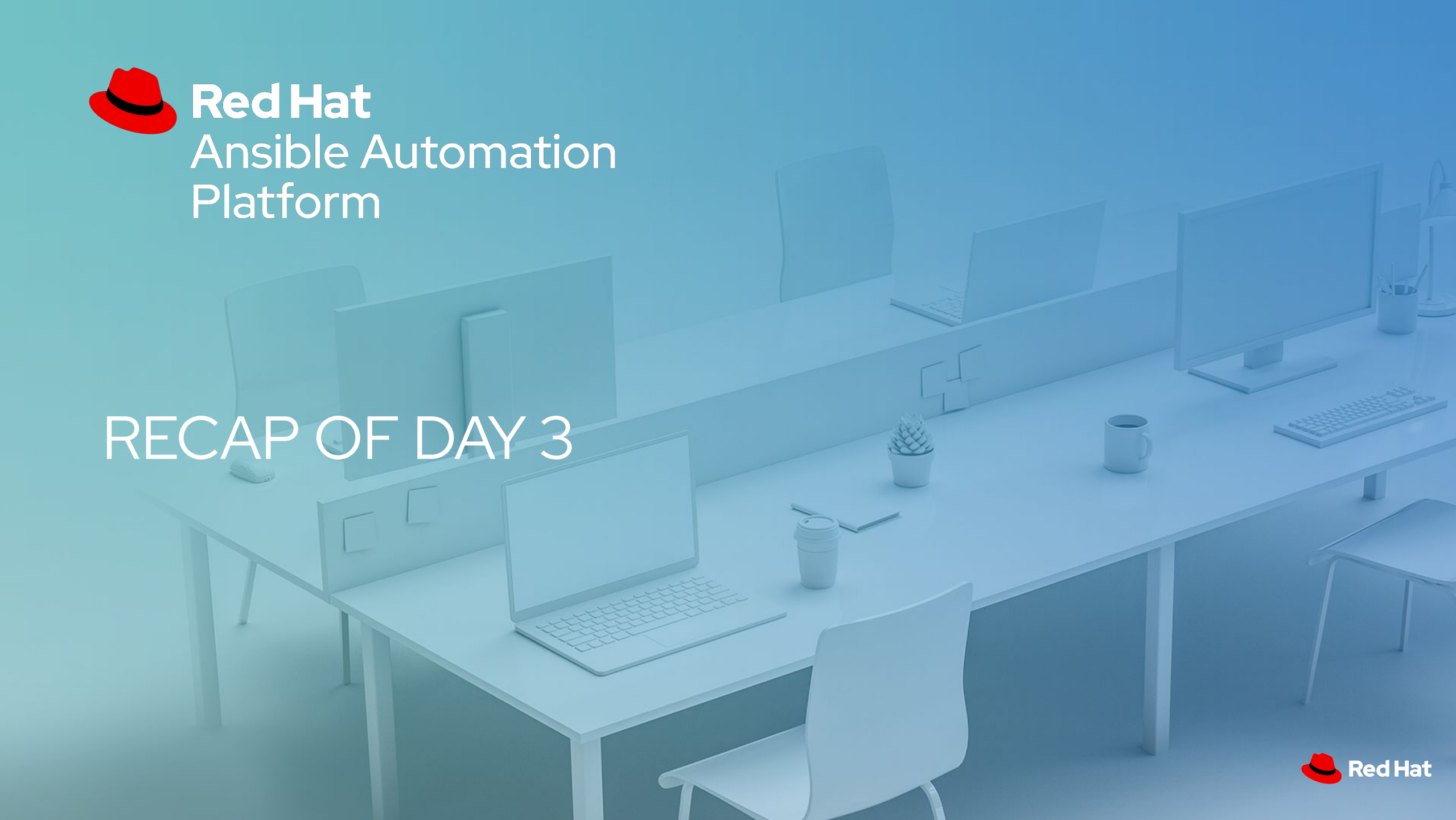
Red Hat
Ansible Automation
Platform

# Day 4:

Topics Covered:

- Threat hunting
- How Tower helps bringing together the automation of different teams
- Exercise 2.2





Red Hat
Ansible Automation
Platform

# RECAP OF DAY 3

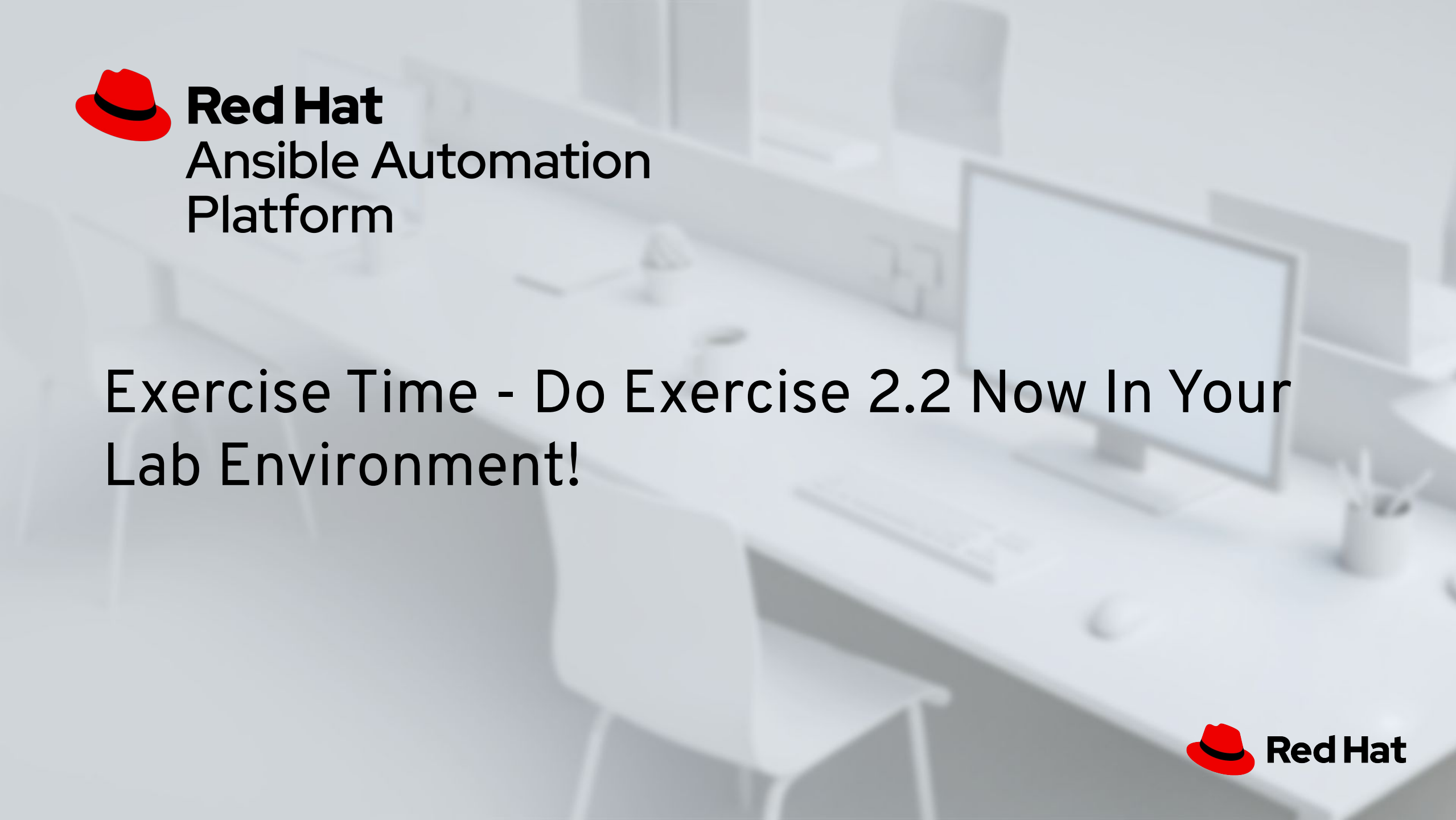
Red Hat

# 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


Red Hat
Ansible Automation
Platform
Exercise Time - Do Exercise 2.2 Now In Your Lab Environment!
Red Hat

# Day 5:

Topics Covered:

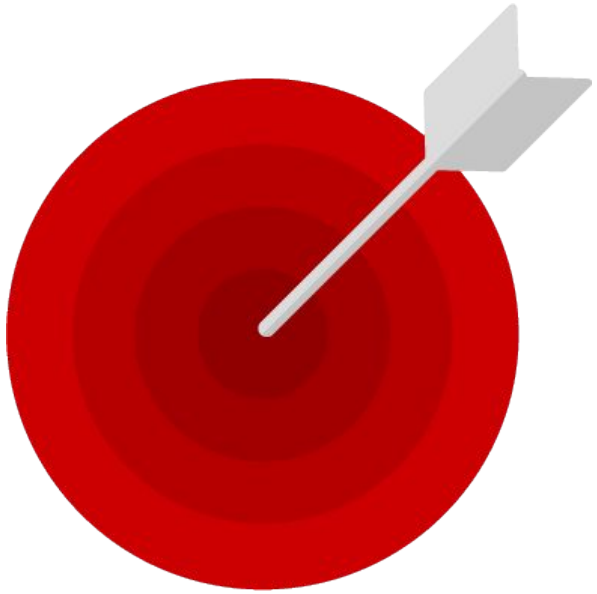
- Incident response
- Exercise 2.3, 2.4
- Recap of Workshop, Wrap it all up
- Discuss Next Steps:
  - Learning Subscription
  - Ansible Trial

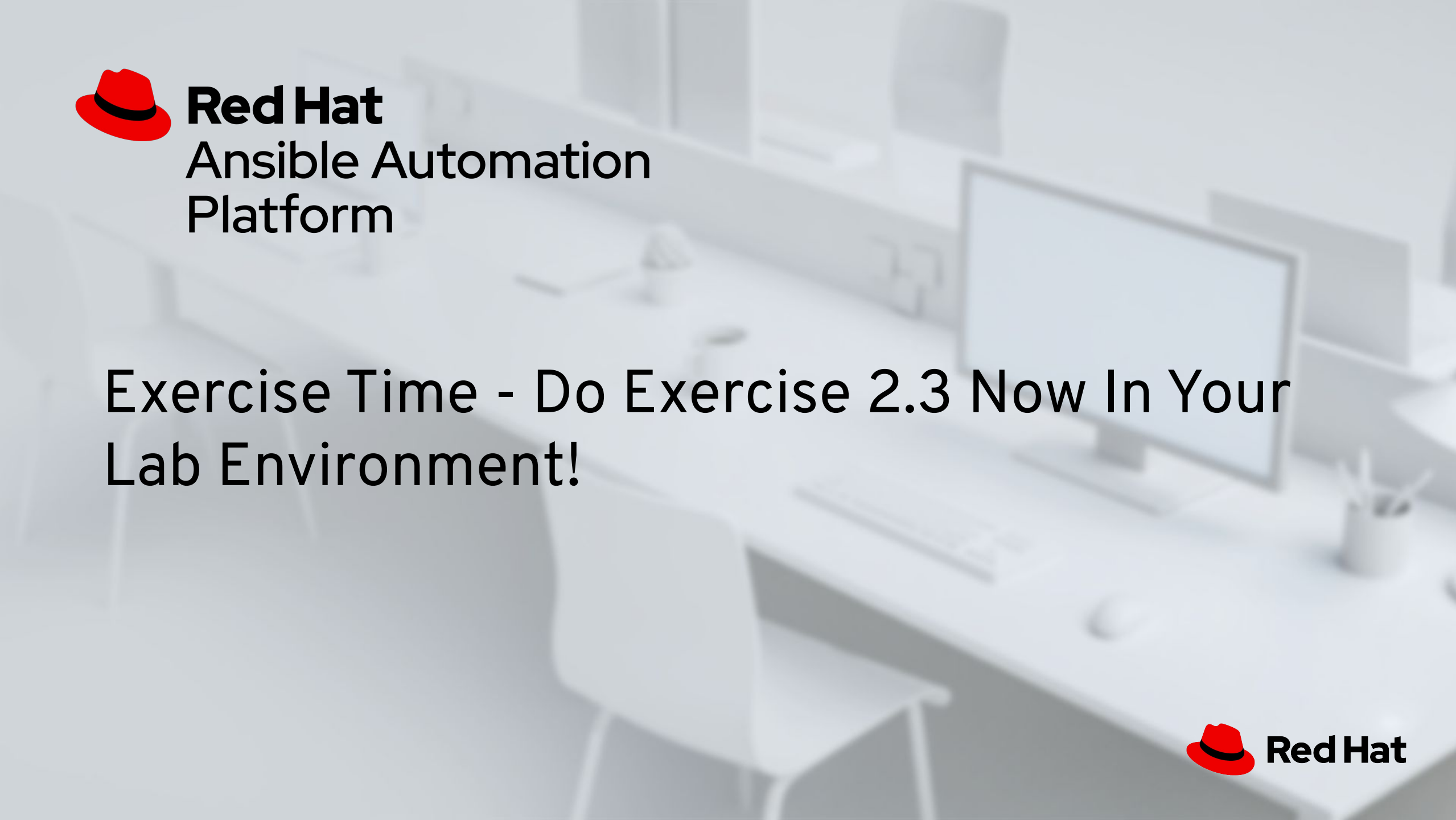


Red Hat
Ansible Automation
Platform

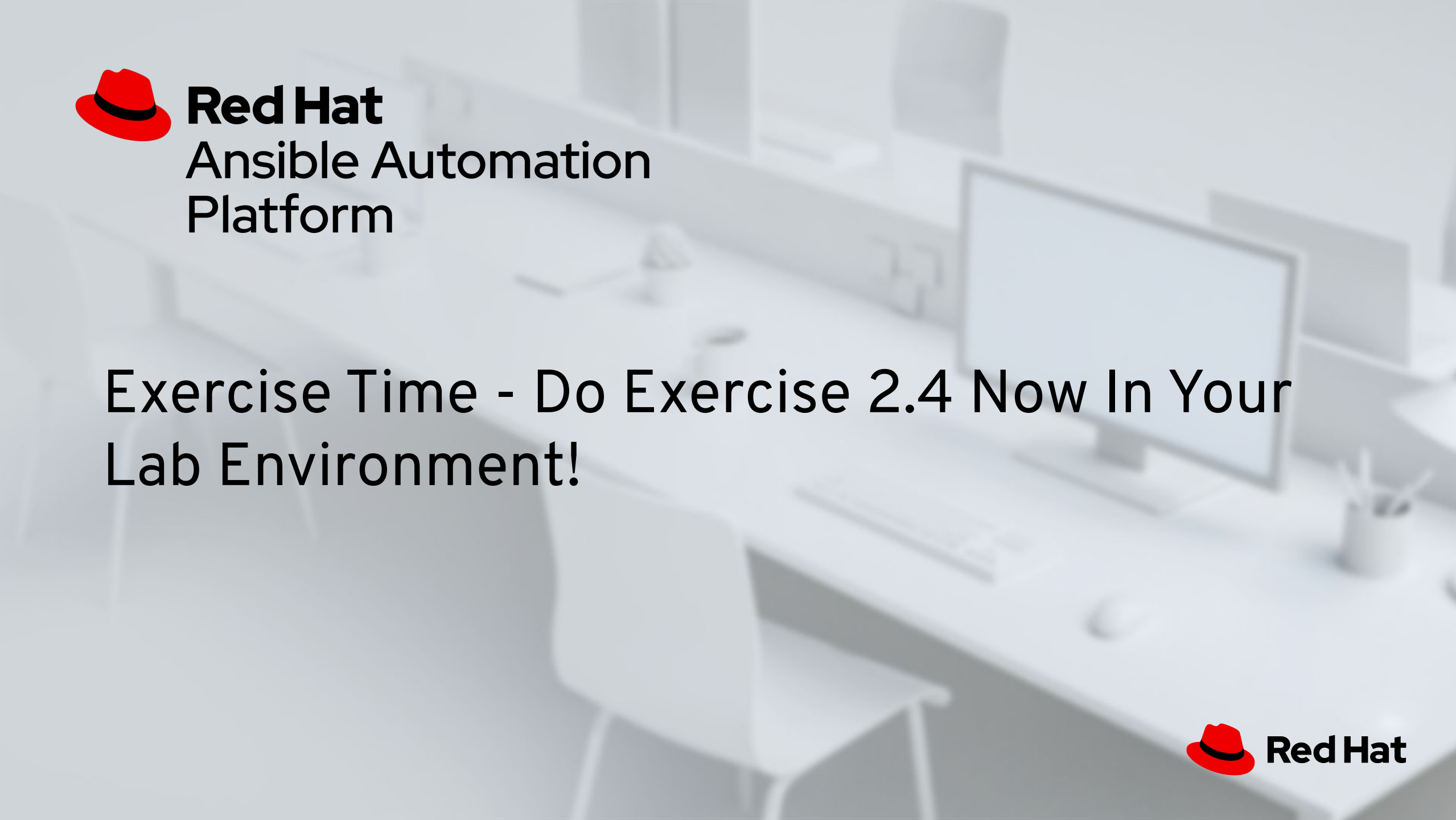
# Persona & Situation

- 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!


Red Hat
Ansible Automation
Platform
Exercise Time - Do Exercise 2.4 Now In Your Lab Environment!
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!

1. Take the survey
2. Learning Subscription
3. Ansible Trial Subscription

# 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

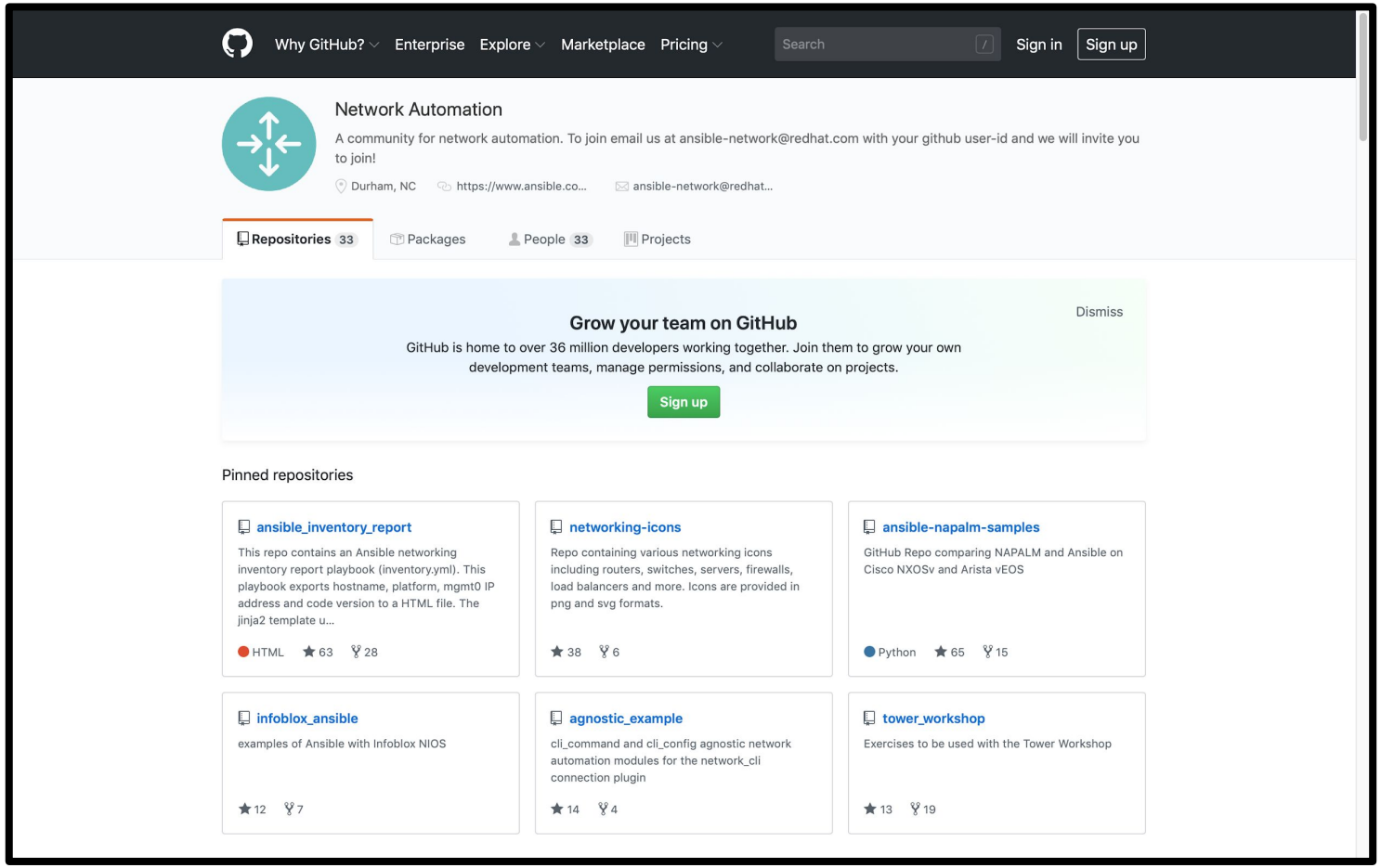
# Chat with us

- **Slack**
  https://ansiblenetwork.slack.com
  Join by clicking here http://bit.ly/ansibleslack

- **IRC**
  #ansible-network on freenode
  http://webchat.freenode.net/?channels=ansible-network

# Bookmark the Github organization

- Examples, samples and demos
- Run network topologies right on your laptop




Red Hat

# Thank you


linkedin.com/company/red-hat


youtube.com/AnsibleAutomation


facebook.com/ansibleautomation


twitter.com/ansible


github.com/ansible



Red Hat