Ansible Security automation workshop

Introduction to Ansible security automation for security teams
Housekeeping

- Timing
- Breaks
- Takeaways
What you will learn

Introduction
- Introducing Ansible Automation Platform
- Ansible security automation overview

Section 1
- Exploring the lab environment
- Ansible Automation Platform basics
- Lab exercises

Section 2
- Security personas
- Automation controller basics
- Lab exercises

Section 3
- Wrapping up
Introduction

Topics Covered:

● Why Ansible Automation Platform?
● Ansible security automation overview
Why Ansible Automation Platform?
Automation happens when one person meets a problem they never want to solve again
Many organizations share the same challenge
Too many unintegrated, domain-specific tools
Why the Ansible Automation Platform?

**Powerful**
Orchestrating complex processes at enterprise scale.

**Simple**
Simplifying automation creation and management across multiple domains.

**Agentless**
Easily integrate with hybrid environments.
Automate the deployment and management of automation

Your entire IT footprint

Do this...

- Orchestrate
- Manage configurations
- Deploy applications
- Provision / deprovision
- Deliver continuously
- Secure and comply

On these...

- Firewalls
- Load balancers
- Applications
- Containers
- Virtualization platforms
- Servers
- Clouds
- Storage
- Network devices
- And more ...
Break down silos

Different teams a single platform

Consistent governance

Cloud

Edge

Datacenter

Line of business

Devs/DevOps

IT ops

SecOps

Network ops
What is Ansible security automation?
What Is Ansible security automation?

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.
What Ansible security automation covers

- SIEM
- IDPS
- ENTERPRISE FIREWALLS
- PAM
- ENDPOINT PROTECTION PLATFORMS
- SECURE EMAIL GATEWAYS
- THREAT INTELLIGENCE PLATFORMS
- SECURE WEB GATEWAYS
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.
What Does It Do?

Investigation Enrichment
Enabling programmatic access to log configurations such as destination, verbosity, etc.

Threat Hunting
Automate alerts, correlation searches and signature manipulation to preemptively identify threats

Incident Response
Creating new security policies to grant access, block or quarantine a machine
Ansible Security Ecosystem

Security Information & Events Management

Enterprise Firewalls

Intrusion Detection & Prevention Systems

Privileged Access Management

Endpoint Protection
Section 1

Introduction to Ansible security automation basics
Exercise 1.1

Topics Covered:

- Exploring the lab environment
- What are automation execution environments?
- Automation content navigator (ansible-navigator)
- Workshop inventory overview
Security workshop architecture

Exercise 1.1

CONTROL HOST

ATTACKER

TARGET/IDPS SNORT

CHECKPOINT NGFW

SSH

REST

SYSLOG

RDP

SIEM IBM QRadar
What are Automation execution environments?
Automation execution environments

Components needed for automation, packaged in a cloud-native way

Execution Environments = Collections + Libraries + Ansible Core

Universal Base Image
Build, create, publish

Development cycle of an automation execution environment
Develop, test, run

How to develop, test and run containerized Ansible content
Workshop Automation execution environment
Ansible Security Roles and Automation content collections already available

Do I need to do anything? No

- Everything has been set up for you and ready to use.
- All Content Collections and Roles are included.
- `ansible-navigator` configuration preconfigured

What’s included?

- We will use `ansible-navigator` to explore the `security_ee` content
Automation content navigator
(ansible-navigator)
ansible-navigator

Using the latest ansible-navigator command

What is ansible-navigator?

ansible-navigator command line utility and text-based user interface (TUI) for running and developing Ansible automation content.

It replaces the previous command used to run playbooks “ansible-playbook”.

```
$ ansible-navigator run playbook.yml
```
# ansible-navigator

Mapping to previous Ansible commands

<table>
<thead>
<tr>
<th>ansible command</th>
<th>ansible-navigator command</th>
</tr>
</thead>
<tbody>
<tr>
<td>ansible-config</td>
<td>ansible-navigator config</td>
</tr>
<tr>
<td>ansible-doc</td>
<td>ansible-navigator doc</td>
</tr>
<tr>
<td>ansible-inventory</td>
<td>ansible-navigator inventory</td>
</tr>
<tr>
<td>ansible-playbook</td>
<td>ansible-navigator run</td>
</tr>
</tbody>
</table>
How do I use ansible-navigator?

Hello ansible-navigator

How do I use ansible-navigator?

As previously mentioned, it replaces the ansible-playbook command.

As such it brings two methods of running playbooks:

- Direct command-line interface
- Text-based User Interface (TUI)

# Direct command-line interface method
$ ansible.navigator run playbook.yml -m stdout

# Text-based User Interface method
$ ansible.navigator run playbook.yml
Workshop inventory
Security workshop inventory

The Basics

- Contains all machines of your environment
- Setup up just for you, individually
- Note your individual IP addresses for each machine
- The exercises needs you to replace example IP addresses with your individual ones
- `/home/student<X>/lab_inventory/hosts`
Exercise 1.1

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

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

[siem]
qradar ansible 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 ansible_host=33.44.55.66 ansible_user=ec2-user private_ip=192.168.3.4
Workshop inventory - Variables

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

[control]
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
Exercise Time!

Do Exercise 1.1 in your lab environment

- Follow the steps to access your environment
- Your environment will have unique IP addresses and DNS names. The screenshots are only examples
- Access to machines is done via the VS Code Online editor using the built-in terminal
Exercise 1.2

Topics Covered:

- Ansible Playbook basics
- Managing Check Point Next Generation Firewalls
- Running your first playbook
Ansible Playbook basics
Ansible playbooks

---

- 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
What makes up an Ansible playbook?

- Plays
- Modules
- Plugins
Ansible plays

What am I automating?

What are they?
Top level specification for a group of tasks. Will tell that play which hosts it will execute on and control behavior such as fact gathering or privilege level.

Building blocks for playbooks
Multiple plays can exist within an Ansible playbook that execute on different hosts.

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

The “tools in the toolkit”

What are they?

Parametrized components with internal logic, representing a single step to be done. The modules “do” things in Ansible.

Language

Usually Python, or Powershell for Windows setups. But can be of any language.

Exercise 1.2
Ansible plugins

The “extra bits”

What are they?

Plugins are pieces of code that augment Ansible’s core functionality. Ansible uses a plugin architecture to enable a rich, flexible, and expandable feature set.

Example become plugin:

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

Example filter plugins:

```yaml
{{ some_variable | to_nice_json }}
{{ some_variable | to_nice_yaml }}
```
- 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
---
- 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
Ansible playbooks

---
- 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

A module
Running Playbooks
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
Managing Check Point Firewalls
Managing Check Point Next Generation Firewalls

How do I use access Check Point firewalls?

▸ Accessed via a central management server
▸ Windows management software is called “SmartConsole”

Lab Check Point instances?

▸ Lab SmartConsole instance is installed on your Windows host.
▸ Accessed via generic RDP client or lab-provided RDP-HTML5 client
▸ HTTP REST API used to call Check Point API
First Check Point Management Server Login
Running your first Playbook
Verifying the playbook was successful

Check Point Firewall Policy

Log into SmartConsole

- Check Point Firewall Policy
- Check network objects for added hosts
- Check policies for added policy
Exercise Time!

Do Exercise 1.2 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
- The Check Point credentials differ from the standard workshop details provided.
- Click on the *Workshop access* link on the exercise page for login information.
Exercise 1.3

Topics Covered:

- What is an IDPS?
- Snort basics
- Intro to Ansible Roles
- Running a playbook interacting with Snort
What is an IDPS?
Intrusion Detection and Prevention Systems

What do they do?

- Monitors systems and networks
- Generates logs for malicious activity or policy violations
- Attempts to stop incident
- Logs are collected centrally, typically to a SIEM
- Typically used by security operations
Snort basics
Snort
Open Source Intrusion Detection and Prevention System

What does Snort do?
- Real time traffic analysis and packet logging on IP networks
- Content search and matching
- Service running on possible targets

Your Snort workshop instance
- Snort is installed on a RHEL instance
- RHEL instance accessed via SSH
- Ansible uses SSH connection to automate Snort
Snort Rules
Open Source Intrusion Detection and Prevention System

What are Snort rules?

▸ Rules determine what traffic is collected
▸ Rules define next step for collected traffic

Want more details on Snort rules?

▸ Snort rule infographic
Intro to Ansible Roles
Ansible security automation roles

Reusable automation actions

What are Ansible roles?

- Group your tasks and variables of your automation in a reusable structure.
- Write roles once, and share them with others who have similar challenges in front of them.

Workshop Ansible Security Roles

- Already included in security EE
- ids_config, ids_rule, log_manager

```yaml
tasks:
  - name: import ids_config role
    include_role:
      name: "ansible_security.ids_config"
```
Exercise Time!

Do Exercise 1.3 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
- Click on the Workshop access link on the exercise page for login information.
Exercise 1.4

Topics Covered:

● Intro to Automation Content Collections
● What is a SIEM?
● Introducing QRadar
● Automating your QRadar instance
Intro to Ansible
Content Collections
Ansible Content Collections

Simplified and consistent content delivery

What are they?
Collections are a data structure containing automation content:
- Modules, playbooks, roles, plugins, docs, tests

Workshop Ansible security collections
- Already included in security EE
- ibm.qradar, ansible.security, check_point.mgmt and more
Using Collections

---

- name: Install NGINX Plus
  hosts: all
  tasks:
  - name: Install NGINX
    include_role:
      name: nginxinc.nginx
      vars:
        nginx_type: plus

- name: Install NGINX App Protect
  include_role:
    name: nginxinc.nginx_app_protect
  vars:
    nginx_app_protect_setup_license: false
    nginx_app_protect_remove_license: false
    nginx_app_protect_install_signatures: false

deploy-nginx.yml
Exercise 1.4

Certified Content Collections

90+ certified platforms

Infrastructure  Cloud  Network  Security

Red Hat  AWS  ARISTA  Check Point
NetApp  Google  CISCO  CYBERARK
IBM  Microsoft  F5  FORTINET
SIEM Overview
What is a SIEM?

Security Information and Event Management

What do they do?

- Supports threat detection, security incident management
- Real-time analysis of security alerts
- Aggregates activity from multiple sources
- Mostly used by security analysts
Introducing QRadar
IBM QRadar SIEM
Security and Information and Event Management

What does it do?
- Collects, analyses and stores network data in real time.
- Provides real-time information and monitoring
- Creates alerts and offenses, and responses to network threats.

QRadar Workshop Instance
- Uses `ibm.qradar` collection
- Ansible connects using QRadar HTTP API
IBM QRadar SIEM

Address most important security challenges

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

Prioritized Threats

Automated Investigations

Proactive Hunting
- Insider Threats
- External threats
- Cloud risks
- Vulnerabilities
- Critical data
Automating your QRadar instance
IBM QRadar Interface
Exercise 1.4

Verify Changes in the UI
Exercise Time!

Do Exercise 1.4 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
- QRadar credentials differ from the standard workshop username and password
- Click on the Workshop access link on the exercise page for login information.
Section 2

Security personas

Ansible Security Automation Use-Cases
Security Personas
Overview
Security Personas Overview

Security Personas

Security Operator
- Toolset
  - Firewalls, PAM, IDPS
- Tasks
  - Manage, configure security devices
  - Escalate security events to analyst
- Challenges
  - Attacks more frequent and sophisticated

Security Analyst
- Toolset
  - SIEM, SOAR
- Tasks
  - Analyze and coordinate remediation
- Challenges
  - Attacks more frequent and sophisticated

Chief Information Security Officer
- Toolset
- Resources
  - Conferences, papers, analyst reports
- Tasks
  - Oversee security operations
  - Direct and manage security strategy
- Challenges
  - Multiple, siloed security teams

CONFIDENTIAL designator
Exercise 2.1

Topics Covered:

● What is investigation enrichment?
● Lab scenario overview
What is investigation enrichment?
Collate, investigate and build context for security anomalies

What is investigation enrichment?

- Process of adding contextual information to security events
- Vital for effective security response
- Typically performed by security analyst
- Events redirected to a SIEM
- Ansible security automation enables programmatic redirection

Lab Scenario

- You, the security analyst, is informed of a security anomaly
- You need to gather events from devices and investigate the event
Investigation Enrichment
Scenario overview
Investigation Enrichment

Attacker launches web attack

1. Launches attack

Analyst

SIEM

Firewall

IDPS

Ansible Automation Platform

Attacker
Investigation Enrichment

Anomaly detected

1. Launches attack

2. Anomalies detected

Analyst

SIEM

SIEM

Attacker

Firewall

IDPS

Ansible Automation Platform

Ansible Automation Platform

Red Hat
Exercise 2.1

Investigation Enrichment

Analyst needs more data points to assess anomaly

1. Launches attack
2. Anomalies detected
3. More data points needed

Analyst

SIEM

Ansible Automation Platform

Firewall

IDPS
Investigation Enrichment

Playbook launched to redirect logs to SIEM for analysis

1. Launches attack

2. Anomalies detected

3. More data points needed

4. Playbook launched to redirect logs

Analyst

SIEM

Firewall

IDPS

Attacker

Ansible Automation Platform
Investigation Enrichment

Ansible security automation redirects events to SIEM

1. **Launches attack**
   - **Attacker**

2. **Anomalies detected**
   - **Firewall**
   - **IDPS**

3. **More data points needed**
   - **Analyst**

4. **Playbook launched to redirect logs**
   - **SIEM**

5. **Logs redirected to SIEM**
Investigation Enrichment

IDPS signature added using Ansible security automation

1. Launches attack
2. Anomalies detected
3. More data points needed
4. Playbook launched to redirect logs
5. Logs redirected to SIEM
6. Playbook launched to add IDPS signature
Exercise 2.1

Investigation Enrichment

IDPS signature generates an offense in the SIEM

1. Launches attack
   - Attacker

2. Anomalies detected
   - IDPS
   - Firewall

3. More data points needed
   - Analyst
   - SIEM

4. Playbook launched to redirect logs
   - Ansible Automation Platform

5. Logs redirected to SIEM
   - SIEM

6. Playbook launched to add IDPS signature
   - Ansible Automation Platform

7. IDPS signature generates an offense
   - Red Hat Ansible Automation Platform
Exercise Time!

Do Exercise 2.1 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
Exercise 2.2

Topics Covered:

● Introduction to automation controller
● What is threat hunting?
● Lab scenario overview
Introduction to Automation controller
Introduction to Automation controller

Red Hat
Ansible Automation Platform

Content creators
Operators
Domain experts
Users

Fueled by an open source community
What is Automation controller?

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

Automation controller provides

- Role-based access control
- Push-button deployment access
- All automations are centrally logged
- Powerful workflows match your IT processes
Anatomy of an Automation Job

- Playbook
- Git / Subversion
- Project
Anatomy of an Automation Job

- Playbook
- Git / Subversion
- Credential
- Project
Anatomy of an Automation Job

- Playbook
- Git / Subversion
- Credential
- Project
- Inventory

Logos: CyberArk, HashiCorp Vault, Netbox, Infoblox, DEVICE42, GitHub, Red Hat Ansible Automation Platform
Anatomy of an Automation Job

Playbook

Git / Subversion

Credential

Project

Inventory

Automation controller

Logo: CyberArk

Logo: HashiCorp

Logo: Vault

Logo: netbox

Logo: Infoblox

Logo: DEVICE42

Logo: GitHub

Logo: Red Hat

Ansible Automation Platform
Automation controller features
Inventories

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

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

Credentials are utilized by Automation Controller 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.
A project is a logical collection of Ansible Playbooks, represented in Ansible Automation Controller.

You can manage Ansible Playbooks and playbook directories by placing them in a source code management system supported by automation controller, including Git, Subversion, and Mercurial.
Job Templates

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

A **Job Template** requires:

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

Job Templates can be found and created by clicking the Templates button under the Resources section on the left menu.
Executing an existing Job Template

Job Templates can be launched by clicking the **rocketship button** for the corresponding Job Template.
Creating a new Job Template (1/2)

New Job Templates can be created by clicking the **Add button**
Creating a new Job Template (2/2)

This **New Job Template** window is where the inventory, project and credential are assigned. The red asterisk * means the field is required.
Surveys

Controller 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 Automation controller survey is a simple question-and-answer form that allows users to customize their job runs. Combine that with controller’s role-based access control, and you can build simple, easy self-service for your users.
Creating a Survey (1/2)

Once a Job Template is saved, the Survey menu will have an **Add Button**

Click the button to open the Add Survey window.

![Add Survey window](image-url)
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.
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.
What is threat hunting?
Threat hunting
Proactively defend your environment

What is threat hunting?

▸ Proactive tasks such as triage, identifying new threats.
▸ Updates from security bulletins and signature manipulation
▸ Correlation of events to create new alerts
▸ Typically performed by security operators and analysts
▸ Requires multiple tools.

Lab Scenario

▸ You’re the security operator in detect a firewall policy violation
▸ Then, we are the security analyst who needs to correlate events and investigate
Threat hunting
Scenario overview
Threat Hunting
DDOS attack started

1. Launches DDOS

Analyst
opsids
opsfirewall

Attacker

SIEM

Automation controller

Firewall
IDPS
Threat Hunting

Firewall policy violated

1. Launches DDOS

2. Firewall violations detected
Threat Hunting

opsfirewall redirects logs to SIEM using controller

1. Attacker launches DDOS
2. Firewall violations detected
3. Redirect firewall logs to SIEM
Threat Hunting

analyst updates QRadar log sources to accept firewall logs

1. Launches DDOS
2. Firewall violations detected
3. Redirect firewall logs to SIEM
4. Update SIEM log sources
Threat Hunting

opsids adds IDPS signature using controller survey

1. Attacker launches DDoS
2. Firewall violations detected
3. Redirect firewall logs to SIEM
4. Analyst updates SIEM log sources
5. opsids adds IDPS signature using controller survey
Threat Hunting

IDPS signature generates an offense in the SIEM

1. Attacker launches DDoS
2. Firewall violations detected
3. Redirect firewall logs to SIEM
4. Update SIEM log sources
5. Add IDPS signature using survey
6. Determines event is false positive
Exercise Time!

Do Exercise 2.2 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
- Click on the **Workshop access** link on the exercise page for login information.
Exercise 2.3

Topics Covered:

- What is incident response?
- Lab scenario overview
Incident response
Mitigating the damage of a security attack or breach

What is incident response?
- RemEDIATE A cyber attack or security breach
- Mitigate the risk caused by the security event
- Involves multiple stakeholders
- Organization must have incident response plan
- Requires multiple tools.

Lab Scenario
- As SecOps, we identify events generated on IDPS.
- Events need to be escalated. Logs must be redirects to SIEM
- As the analyst, we will inspect and create a remediation plan
Incident Response
Scenario overview
Incident Response
Attacker launches SQL injection attack

1. Attacker launches SQL injection attack
Incident Response

IDPS alerts identified by SecOps

1. Attacker launches SQL Injection attack
2. IDPS generates alerts identified by SecOps
Incident Response

SecOps forwards events to SIEM

1. Attacker launches SQL Injection attack
2. IDPS alerts generated
3. Redirect IDPS logs to SIEM

Exercise 2.3
Incident Response

Analyst investigates offense generated on the SIEM

1. Attacker launches SQL Injection attack

2. SecOps receives IDPS alerts generated

3. Analyst redirects IDPS logs to SIEM

4. Analyst investigates the offense
Incident Response

Analyst determines this is a cyber attack and launches remediation

1. Attacker launches SQL Injection attack
2. IDPS alerts generated
3. Redirect IDPS logs to SIEM
4. Analyst investigates offense
5. Updates firewall policy to stop attack

SIEM

Ansible Automation Platform

SecOps

Analyst

Firewall

IDPS
Incident Response

SQL Injection attack remediated

1. Attacker launches a SQL Injection attack.
2. IDPS alerts generated.
3. Redirect IDPS logs to SIEM.
4. Analyst investigates offense.
5. Updates firewall policy to stop attack.
6. Attack remediated.

Ansible Automation Platform

Analyst

SecOps

Attacker

SIEM

IDPS

Firewall
Exercise Time!

Do Exercise 2.3 in your lab environment

- Follow the steps in the exercises
- Remember to use the IP addresses assigned to you
- Click on the Workshop access link on the exercise page for login information.
Section 3

Wrapping up
Resources

- Free Ansible Automation Platform 2 overview course
- Ansible security automation (Overview)
- Simplify your security operations center (E-Book)
- Red Hat Ansible Automation Platform blog
- Start your Ansible Automation Platform trial

What you get with this trial
A single, self-supported 60-day subscription for Red Hat® Ansible® Automation Platform for Red Hat Enterprise Linux®
Access to Red Hat’s award-winning Customer Portal, including documentation, helpful videos, discussions, and more

red.ht/ansible_trial
Chat with us

- Slack
Bookmark the Github organization

- GitHub organization
- Examples, samples and demos
- Webinar playbooks
Where to go next

Learn more
- Workshops
- Documents
- Youtube
- Twitter

Get started
- Trial subscription
- cloud.redhat.com

Get serious
- Red Hat Automation Adoption Journey
- Red Hat Training
- Red Hat Consulting
Thank you