Red Hat
Ansible Automation
Platform

ANSIBLE AUTOMATES

How to leverage Ansible Security Automation if you are a Dev or a SecOp

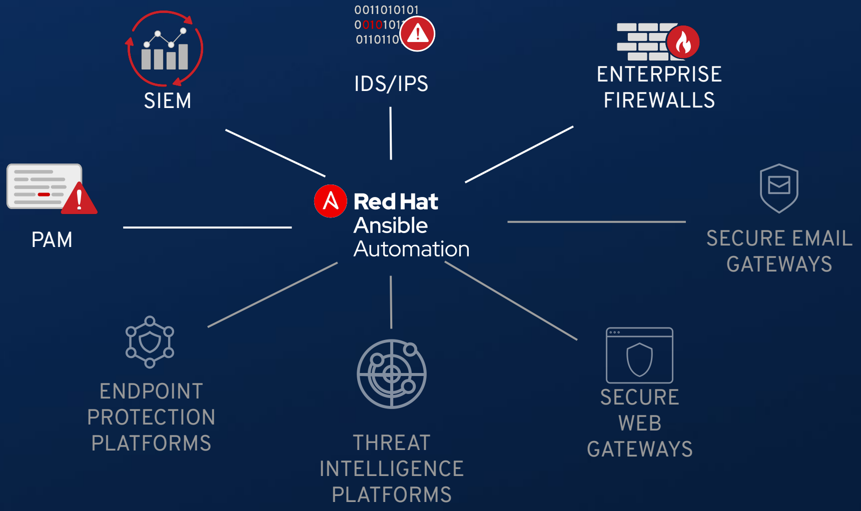
Chris Tjon
Senior Solutions Architect

Red Hat

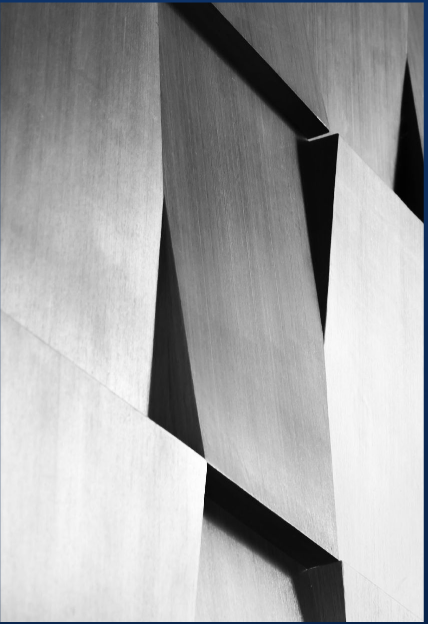
ANSIBLE SECURITY AUTOMATION

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.



WHY ANSIBLE SECURITY AUTOMATION?



“For one, security teams are overwhelmed. **The average security team typically examines less than 5% of the alerts flowing into them every day** (and in many cases, much less than that). "

Venturebeat

57% of respondents said the
time to resolve an incident has increased

65% reported the
severity of attacks has increased

Ponemon Institute

63% of respondents say their leaders understand that **automation, machine learning, artificial intelligence and orchestration** strengthens cyber resilience.

Ponemon Institute

Source:
https://venturebeat.com/2017/12/16/the-lesson-behind-2017s-biggest-enterprise-security-story/
The Third Annual Study on the Cyber Resilient Organization - Ponemon Institute (Sponsored by IBM)

WHAT TYPES OF DEVICES?
WHO ARE OUR PARTNERS?

Security Information & Events Management

splunk>

IBM

Enterprise Firewalls

Check Point
SOFTWARE TECHNOLOGIES LTD

CISCO

f5

FORTINET

Intrusion Detection & Prevention Systems

SNORT

Check Point
SOFTWARE TECHNOLOGIES LTD

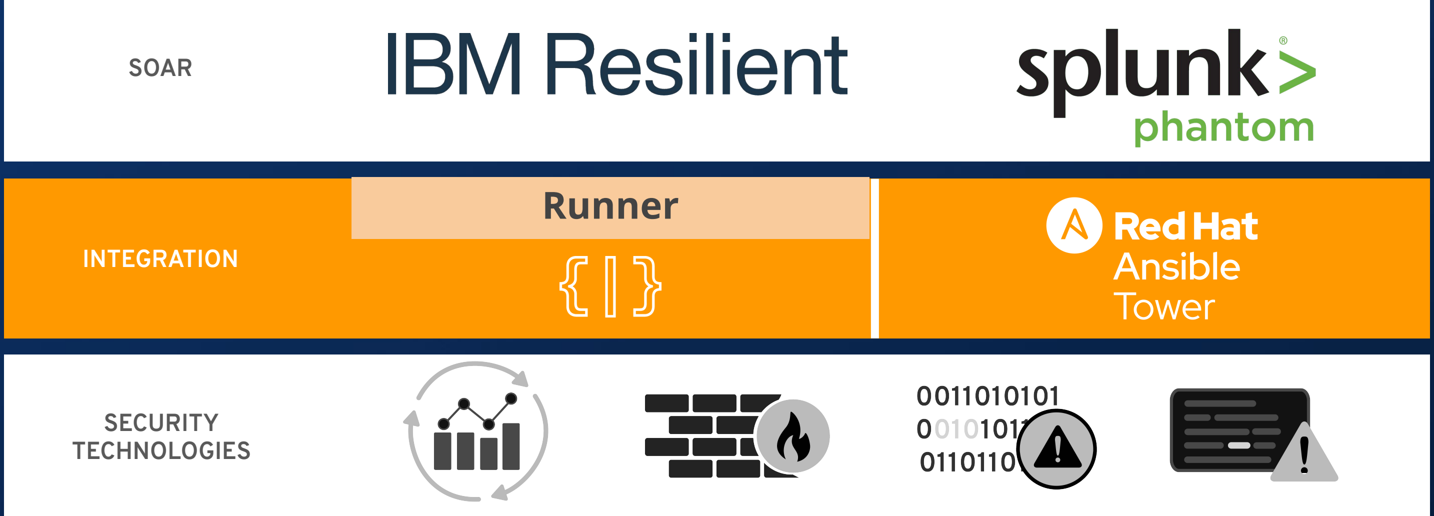
FORTINET

Privileged Access Management

CYBERARK

Syncope

ANSIBLE INTEGRATION WITH SOAR



SOAR

IBM Resilient

splunk> phantom

INTEGRATION

Runner

Red Hat Ansible Tower

SECURITY TECHNOLOGIES

WHICH **SOC** ACTIVITIES CAN BENEFIT THE MOST FROM AUTOMATION?

Triage Of Suspicious Activities

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

Threat Hunting

Automating alerts, correlation searches and signature manipulation

Incident Response

Creating new security policies to whitelist, blacklist or quarantine a machine

WHAT **DEVOPS** ACTIVITIES CAN BENEFIT THE MOST FROM AUTOMATION?

Deployment

Ensure Code Deployment Commit Has Firewall Rules, IDS Signatures, Passes Validation

Baselining

Update relevant security tools to understand the application behaviour

Integration

Interact with the broader corporate infrastructure

FIREWALL MANAGEMENT

BLACKLIST THE ATTACKER IP ON CHECK POINT NGFW

INCIDENT RESPONSE

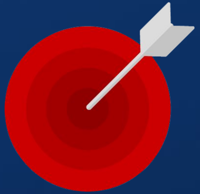
Creating new security policies to whitelist, blacklist or quarantine a machine

```
- hosts: checkpoint
  connection: httpapi
  tasks:
  - name: Create blacklist IP
    include_role:
      name: acl_manager
      tasks_from: blacklist_ip
    vars:
        source_ip: “{{ attacker_ip }}”
        destination_ip: “{{ target_ip }}”
        ansible_network_os: checkpoint
```

Check Point®
SOFTWARE TECHNOLOGIES LTD

ref: https://github.com/ansible-security/acl_manager

BLACKLIST THE ATTACKER URL ON CISCO FTD



INCIDENT RESPONSE

Creating new security policies to whitelist, blacklist or quarantine a machine

```
- hosts: ftd
  connection: httpapi
  tasks:
  - name: Create blacklist URL
    include_role:
      name: acl_manager
      tasks_from: blacklist_url
    vars:
      blacklist_url_type: url
      blacklist_name: ”attacker_url”
      blacklist_url_description: “Attacker url
description”
      blacklist_url: www.attacker.com
      ansible_network_os: cisco_ftd
```

ref: https://github.com/ansible-security/acl_manager

BRING IT INTO DEV WORKFLOWS WITH CI

DEPLOYMENT

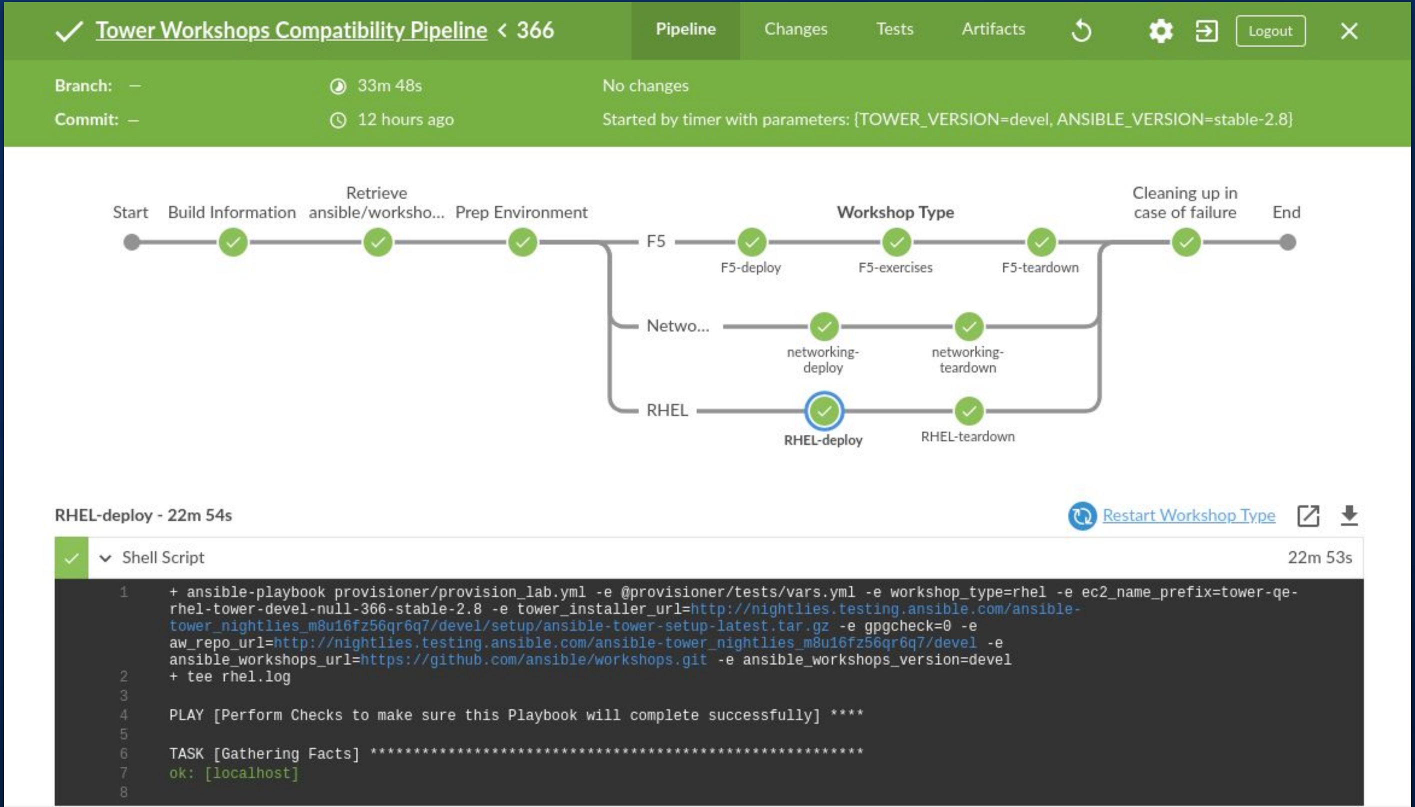
Ensure CI Security Environment Setup on Fresh Deployment Code Commit Has Firewall Rules, IDS Signatures, Passes Validation

```
- hosts: checkpoint
  connection: httpapi
  tasks:
  - name: Grant Access to App Floating IP
    include_role:
      name: acl_manager
      tasks_from: whitelist_ip
    vars:
        source_ip: *
        destination_ip: “{{ app_float_ip }}”
        ansible_network_os: checkpoint
```

Check Point®
SOFTWARE TECHNOLOGIES LTD

ref: https://github.com/ansible-security/acl_manager

BRING IT INTO DEV WORKFLOWS WITH CI



Branch: –

Commit: –

33m 48s

12 hours ago

No changes

Started by timer with parameters: {TOWER_VERSION=devel, ANSIBLE_VERSION=stable-2.8}

RHEL-deploy - 22m 54s

✓ Shell Script 22m 53s

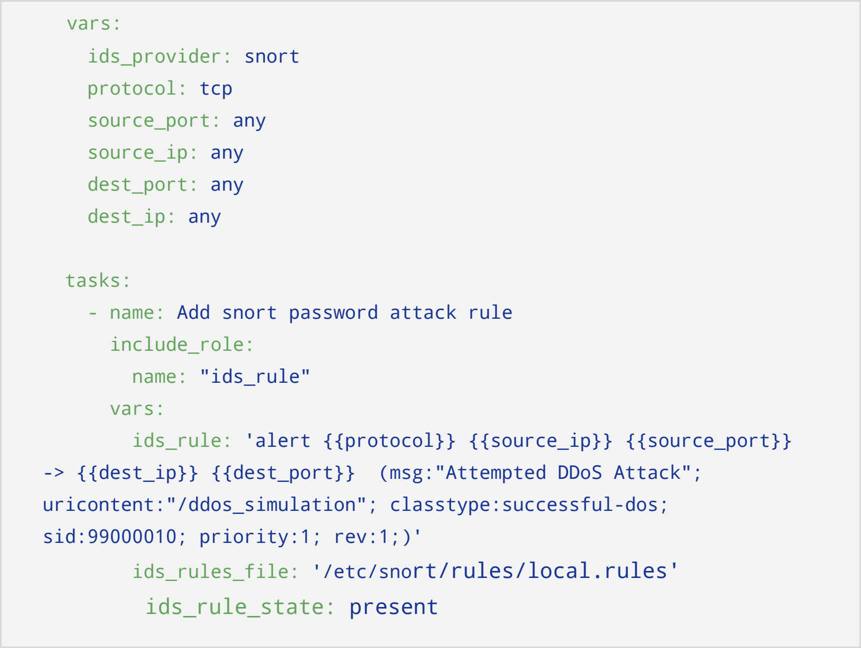
```
+ ansible-playbook provisioner/provision_lab.yml -e @provisioner/tests/vars.yml -e workshop_type=rhel -e ec2_name_prefix=tower-qe-
rhel-tower-devel-null-366-stable-2.8 -e tower_installer_url=http://nightlies.testing.ansible.com/ansible-
tower_nightlies_m8u16fz56qr6q7/devel/setup/ansible-tower-setup-latest.tar.gz -e gpgcheck=0 -e
aw_repo_url=http://nightlies.testing.ansible.com/ansible-tower_nightlies_m8u16fz56qr6q7/devel -e
ansible_workshops_url=https://github.com/ansible/workshops.git -e ansible_workshops_version=devel
+ tee rhel.log

PLAY [Perform Checks to make sure this Playbook will complete successfully] ****

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

```

INTRUSION DETECTION/PREVENTION SYSTEMS MANAGEMENT

IMPLEMENTING A NEW SIGNATURE ON SNORT IDS

THREAT HUNTING

Automating alerts, correlation searches and signature manipulation

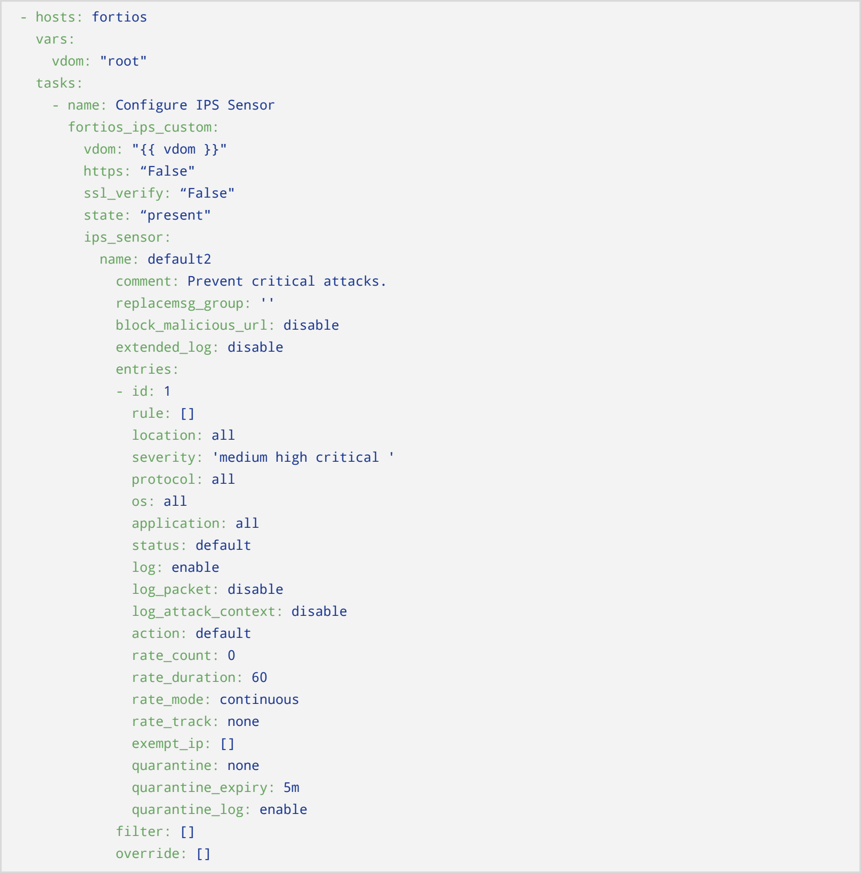


```
  vars:
    ids_provider: snort
    protocol: tcp
    source_port: any
    source_ip: any
    dest_port: any
    dest_ip: any

  tasks:
    - name: Add snort password attack rule
      include_role:
        name: "ids_rule"
      vars:
        ids_rule: 'alert {{protocol}} {{source_ip}} {{source_port}}
-> {{dest_ip}} {{dest_port}}  (msg:"Attempted DDoS Attack";
uricontent:"/ddos_simulation"; classtype:successful-dos;
sid:99000010; priority:1; rev:1;)'
        ids_rules_file: '/etc/snort/rules/local.rules'
         ids_rule_state: present
```

IMPLEMENTING A NEW IPS SENSOR ON FORTINET FORTIOS

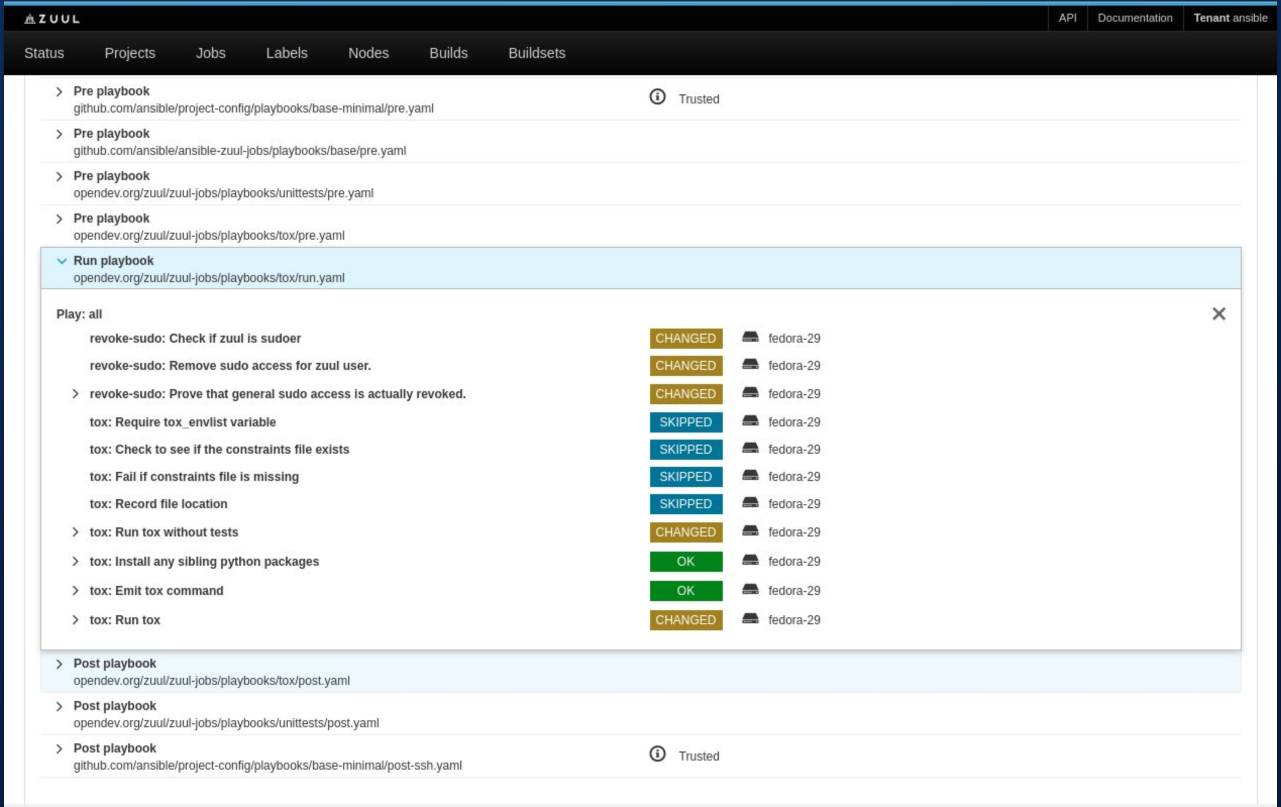
BASELINING

Update relevant security tools to understand the application behaviour



```
- hosts: fortios
  vars:
    vdom: "root"
  tasks:
    - name: Configure IPS Sensor
      fortios_ips_custom:
        vdom: "{{ vdom }}"
        https: “False"
        ssl_verify: “False"
        state: “present"
        ips_sensor:
          name: default2
            comment: Prevent critical attacks.
            replacemsg_group: ''
            block_malicious_url: disable
            extended_log: disable
            entries:
            - id: 1
              rule: []
              location: all
              severity: 'medium high critical '
              protocol: all
              os: all
              application: all
              status: default
              log: enable
              log_packet: disable
              log_attack_context: disable
              action: default
              rate_count: 0
              rate_duration: 60
              rate_mode: continuous
              rate_track: none
              exempt_ip: []
              quarantine: none
              quarantine_expiry: 5m
              quarantine_log: enable
            filter: []
            override: []
```

DEVSECOPS REAL WORLD SCENARIO – ZUUL CI



Pre playbook
github.com/ansible/project-config/playbooks/base-minimal/pre.yaml — Trusted

Pre playbook
github.com/ansible/ansible-zuul-jobs/playbooks/base/pre.yaml

Pre playbook
opendev.org/zuul/zuul-jobs/playbooks/unittests/pre.yaml

Pre playbook
opendev.org/zuul/zuul-jobs/playbooks/tox/pre.yaml

Run playbook
opendev.org/zuul/zuul-jobs/playbooks/tox/run.yaml

Play: all

revoke-sudo: Check if zuul is sudoer	CHANGED	fedora-29
revoke-sudo: Remove sudo access for zuul user.	CHANGED	fedora-29
revoke-sudo: Prove that general sudo access is actually revoked.	CHANGED	fedora-29
tox: Require tox_envlist variable	SKIPPED	fedora-29
tox: Check to see if the constraints file exists	SKIPPED	fedora-29
tox: Fail if constraints file is missing	SKIPPED	fedora-29
tox: Record file location	SKIPPED	fedora-29
tox: Run tox without tests	CHANGED	fedora-29
tox: Install any sibling python packages	OK	fedora-29
tox: Emit tox command	OK	fedora-29
tox: Run tox	CHANGED	fedora-29

Post playbook
opendev.org/zuul/zuul-jobs/playbooks/tox/post.yaml

Post playbook
opendev.org/zuul/zuul-jobs/playbooks/unittests/post.yaml

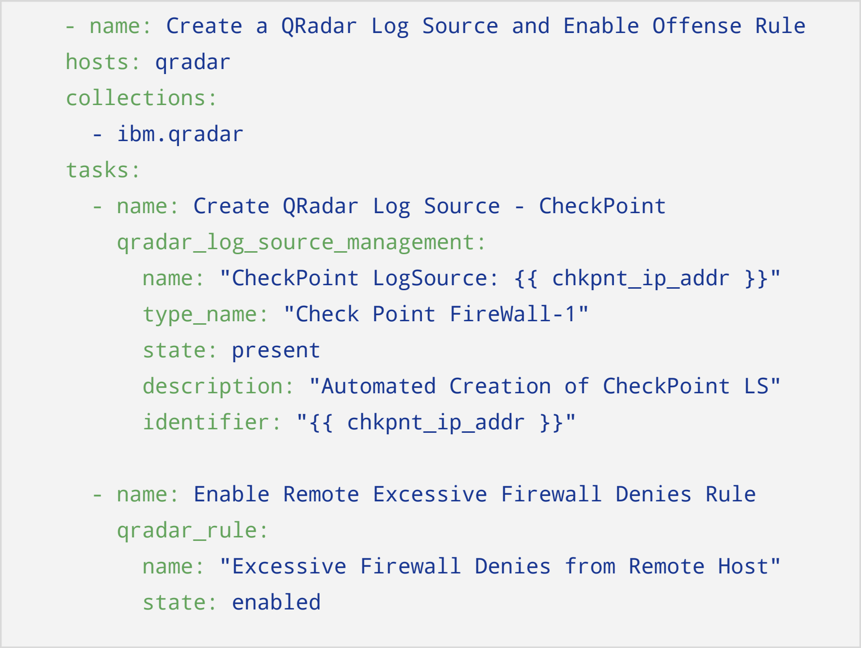
Post playbook
github.com/ansible/project-config/playbooks/base-minimal/post-ssh.yaml — Trusted

SECURITY INFORMATION & EVENT MANAGEMENT (SIEM)

ADD LOG SOURCE AND ENABLE SIEM RULE TO GENERATE OFFENSES

TRIAGE OF SUSPICIOUS ACTIVITIES

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



```yaml
- name: Create a QRadar Log Source and Enable Offense Rule
hosts: qradar
collections:
  - ibm.qradar
tasks:
  - name: Create QRadar Log Source - CheckPoint
    qradar_log_source_management:
      name: "CheckPoint LogSource: {{ chkpnt_ip_addr }}"
      type_name: "Check Point FireWall-1"
      state: present
      description: "Automated Creation of CheckPoint LS"
      identifier: "{{ chkpnt_ip_addr }}"

  - name: Enable Remote Excessive Firewall Denies Rule
    qradar_rule:
      name: "Excessive Firewall Denies from Remote Host"
      state: enabled
```

ADD LOG SOURCE AND ENABLE SIEM RULE TO GENERATE OFFENSES

TRIAGE OF SUSPICIOUS ACTIVITIES

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

```
- name: Get info about Qradar Offense - Excessive Offense
  qradar_offense_info:
    name: “Excessive Offense”
  register: offense_info

- name: Assign Actions to Offense
  qradar_offense_action:
    id: offense_info[“offenses”][0][“id”]
    status: "hidden"
    assigned_to: "admin"
    protected: false

- name: Add Note to Offense
  qradar_offense_note:
    id: offense_info[“offenses”][0][“id”]
    note_text: "Run investigate_offense.yml playbook"
```

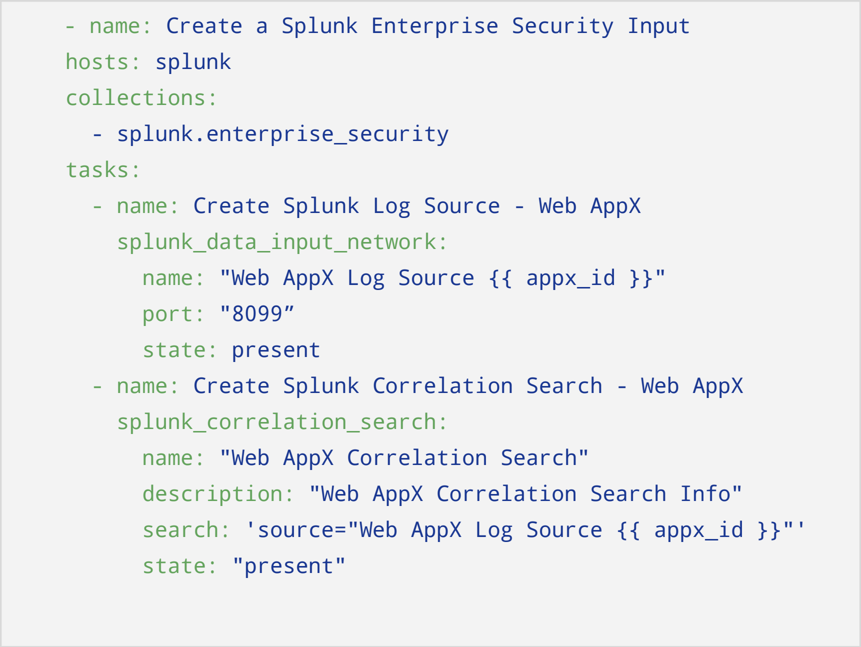
IBM

QRadar

ADD LOG SOURCE AND ENABLE SIEM RULE TO GENERATE OFFENSES

INTEGRATION

Interact with the broader corporate infrastructure



```
- name: Create a Splunk Enterprise Security Input
hosts: splunk
collections:
  - splunk.enterprise_security
tasks:
  - name: Create Splunk Log Source - Web AppX
    splunk_data_input_network:
      name: "Web AppX Log Source {{ appx_id }}"
      port: "8099”
      state: present
  - name: Create Splunk Correlation Search - Web AppX
    splunk_correlation_search:
      name: "Web AppX Correlation Search"
      description: "Web AppX Correlation Search Info"
      search: 'source="Web AppX Log Source {{ appx_id }}"'
      state: "present"
```

SECOPS REAL WORLD SCENARIO

IBM QRadar

Generates an *offense* from an anomaly on the intranet perimeter or outbound traffic from an internal machine.

IBM QRadar

An **investigation is opened** and populated with all relevant data.

CISCO

The IP address is **added to the *blacklist*** on Firepower through FTD.

IBM QRadar

The *offense* criteria are no longer met.

IBM QRadar

The investigation is **populated** with data from the actions taken.

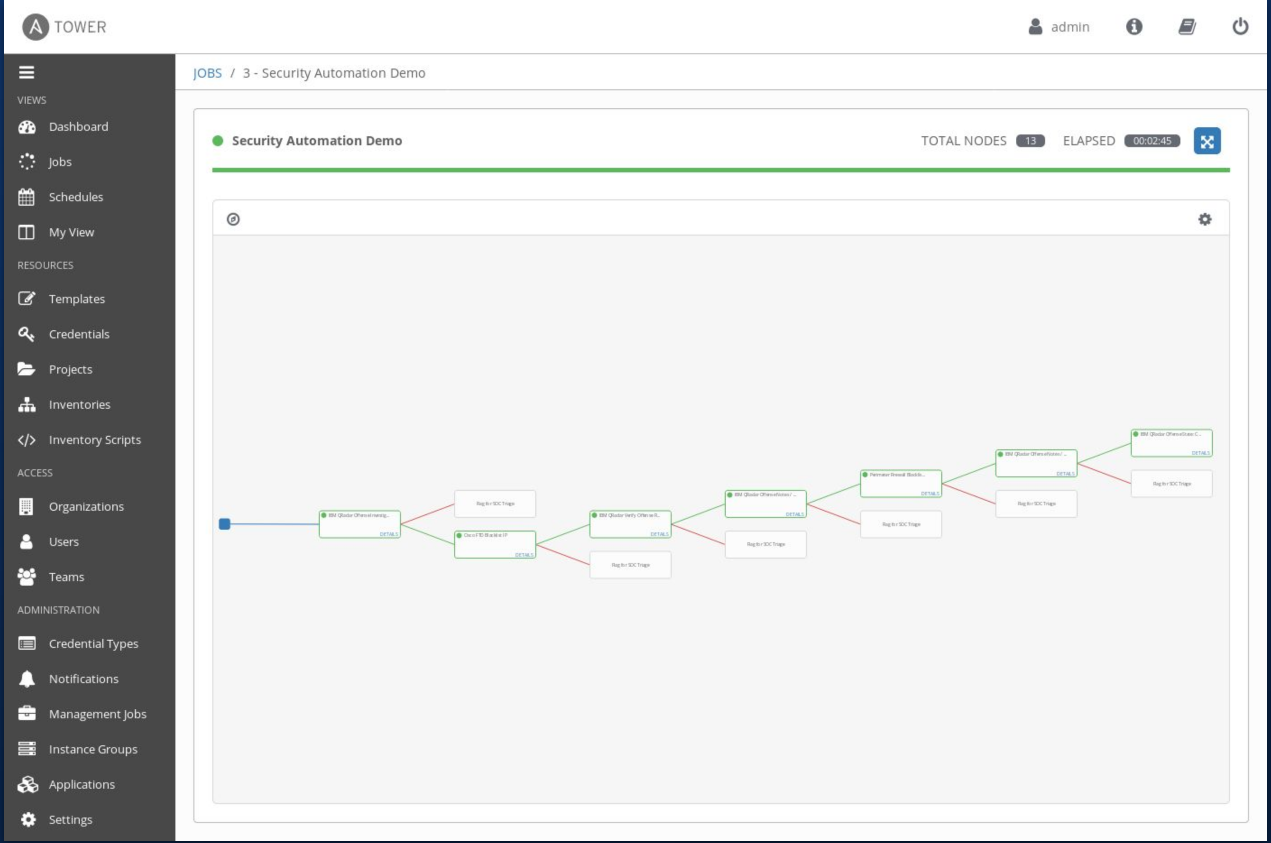
Check Point SOFTWARE TECHNOLOGIES LTD FORTINET

The IP address is **added to the *blacklist*** on the other firewalls in the perimeter.

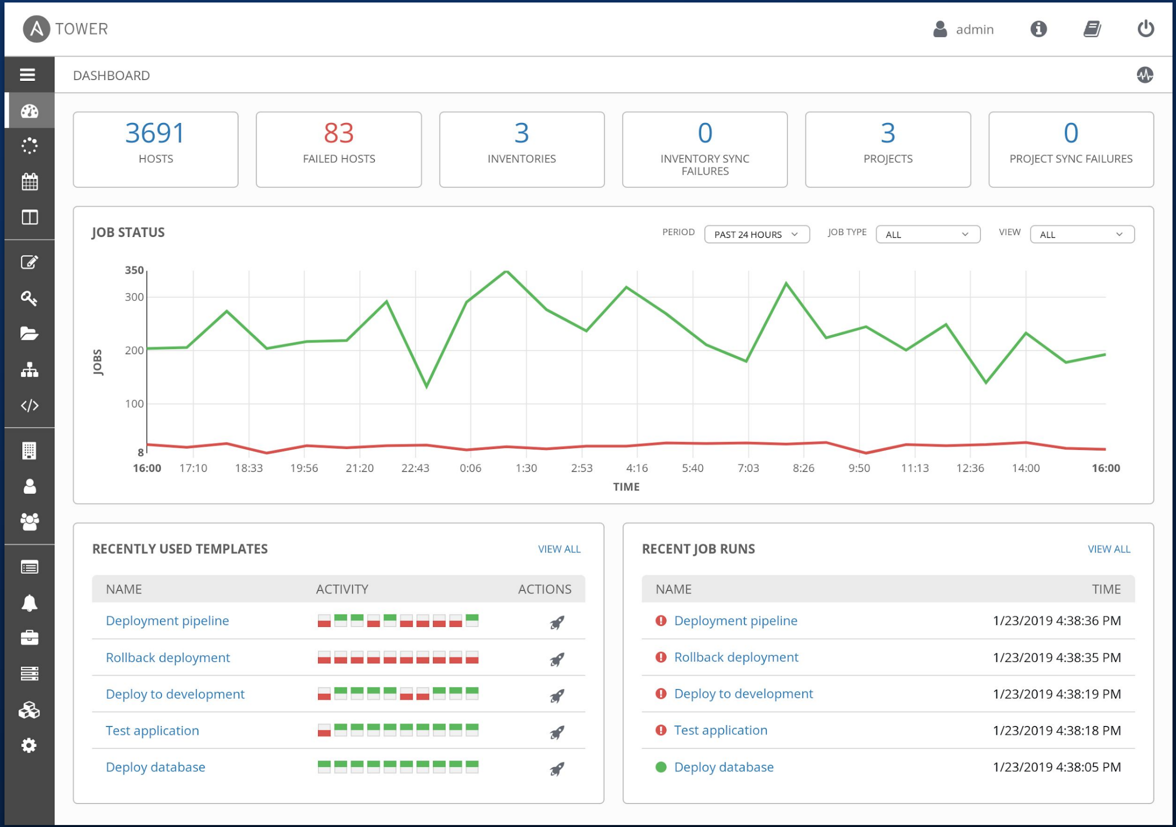
IBM QRadar

The investigation is **populated** with data from the actions taken and then **closed.** The *offense* on QRadar is **closed.**

SECOPS REAL WORLD SCENARIO



DEV REAL WORLD SCENARIO
DEPLOY WITH TOWER



3691	83	3	0	3	0
HOSTS	FAILED HOSTS	INVENTORIES	INVENTORY SYNC FAILURES	PROJECTS	PROJECT SYNC FAILURES

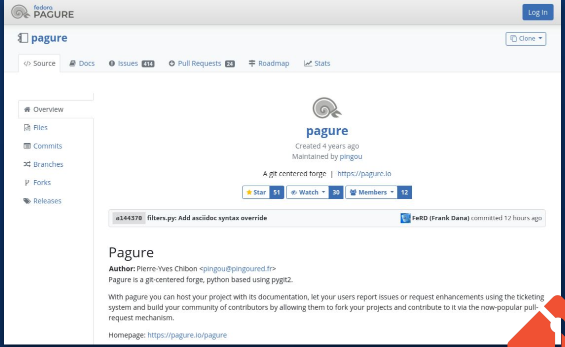
RECENTLY USED TEMPLATES VIEW ALL

NAME	ACTIVITY	ACTIONS
Deployment pipeline	🟥🟩🟩🟥🟩🟥🟥🟥🟥🟩	🚀
Rollback deployment	🟥🟥🟥🟥🟥🟥🟥🟥🟥🟥	🚀
Deploy to development	🟥🟩🟩🟩🟩🟥🟥🟩🟩🟩	🚀
Test application	🟥🟩🟩🟩🟩🟩🟩🟩🟩🟩	🚀
Deploy database	🟩🟩🟩🟩🟩🟩🟩🟩🟩🟩	🚀

RECENT JOB RUNS VIEW ALL

NAME	TIME
❗ Deployment pipeline	1/23/2019 4:38:36 PM
❗ Rollback deployment	1/23/2019 4:38:35 PM
❗ Deploy to development	1/23/2019 4:38:19 PM
❗ Test application	1/23/2019 4:38:18 PM
● Deploy database	1/23/2019 4:38:05 PM

DEVSECOPS



pagure

Created 4 years ago
Maintained by pingou

A git centered forge | https://pagure.io

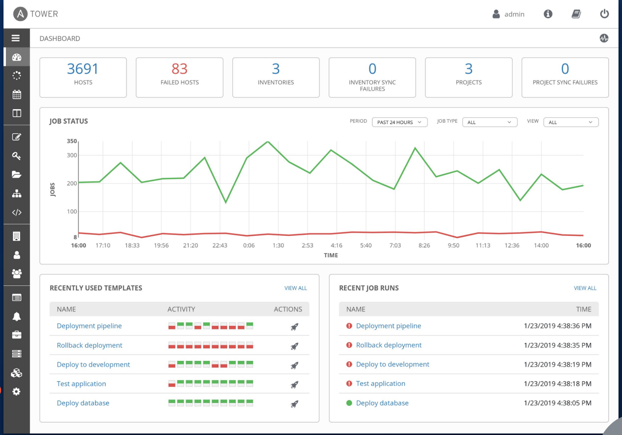
a144370 filters.py: Add asciidoc syntax override — FeRD (Frank Dana) committed 12 hours ago

Pagure

Author: Pierre-Yves Chibon <pingou@pingoured.fr>
Pagure is a git-centered forge, python based using pygit2.

With pagure you can host your project with its documentation, let your users report issues or request enhancements using the ticketing system and build your community of contributors by allowing them to fork your projects and contribute to it via the now-popular pull-request mechanism.

Homepage: https://pagure.io/pagure



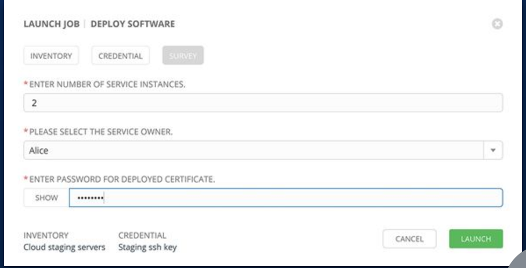
3691 HOSTS	83 FAILED HOSTS	3 INVENTORIES	0 INVENTORY SYNC FAILURES	3 PROJECTS	0 PROJECT SYNC FAILURES

RECENTLY USED TEMPLATES

NAME	ACTIVITY	ACTIONS
Deployment pipeline	🟥🟩🟩🟥🟩🟥🟥🟥🟥🟩	🚀
Rollback deployment	🟥🟥🟥🟥🟥🟥🟥🟥🟥🟥	🚀
Deploy to development	🟥🟩🟩🟩🟩🟥🟥🟩🟩🟩	🚀
Test application	🟥🟩🟩🟩🟩🟩🟩🟩🟩🟩	🚀
Deploy database	🟩🟩🟩🟩🟩🟩🟩🟩🟩🟩	🚀

RECENT JOB RUNS

NAME	TIME
❗ Deployment pipeline	1/23/2019 4:38:36 PM
❗ Rollback deployment	1/23/2019 4:38:35 PM
❗ Deploy to development	1/23/2019 4:38:19 PM
❗ Test application	1/23/2019 4:38:18 PM
● Deploy database	1/23/2019 4:38:05 PM



LAUNCH JOB | DEPLOY SOFTWARE

*ENTER NUMBER OF SERVICE INSTANCES.
2

*PLEASE SELECT THE SERVICE OWNER.
Alice

*ENTER PASSWORD FOR DEPLOYED CERTIFICATE.
••••••••

INVENTORY
Cloud staging servers

CREDENTIAL
Staging ssh key

RELEVANT RESOURCES

Ansible.com:	https://www.ansible.com/use-cases/security-automation
Access:	https://access.redhat.com/articles/4001711
Galaxy:	https://galaxy.ansible.com/ansible_security https://galaxy.ansible.com/ibm/qradar https://galaxy.ansible.com/splunk/enterprise_security https://galaxy.ansible.com/cyberark
GitHub:	https://github.com/ansible-security
IRC:	#ansible-security on irc.freenode.net

THANK YOU!

NARRATIVE

- Ansible security automation intro
    - Ansible security automation history
    - Ansible security automation available platforms/content
- How SecOps will consume ASA vs how Developers will consume the same content
    - SecOps using Ansible for Response and Remediation > Our use cases
    - Developers using Ansible for Deployment > Web App CI/CD
- Example 1: Firewall management
    - SecOps use these modules to blacklist/whitelist an IP/URL as a result of an investigation
    - Devs use these modules to open all the relevant ports on the corporate firewalls when deploying a new application
- Example 2: IDS management
    - SecOps use these modules for threat hunting proactively updating the signatures
    - Devs use these modules to update snort signatures and identify what is and is not valid traffic
- Example 3: SIEM management
    - SecOps use these modules to enable relevant search queries and update investigations
    - Devs use these module to send the relevant logs of the new workloads to the SIEM
- All of that comes together
    - For SecOps to fully automate end to end investigation and remediation processes
    - For Devs to integrate security tools in their CI/CD pipeline
- The future
    - DevSecOps > Ansible security automation will support code/dev oriented security tools and Ansible language can be used as the defacto standard for interactions between SecOps and Dev