Ansible Linux Automation Workshop

Introduction to Ansible for Red Hat Enterprise Linux Automation for System Administrators and Operators
What you will learn

▸ Overview of public cloud provisioning
▸ Converting shell commands into Ansible Commands.
▸ Retrieving information from hosts
▸ Deploying applications at scale
▸ Self-service IT via surveys
▸ Overview of System Roles for Red Hat Enterprise Linux
▸ Overview of Red Hat Insights integration
Introduction

Topics Covered:

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

- Network ops
- SecOps
- Devs/DevOps
- IT ops
Why the Ansible Automation Platform?

Powerful
Orchestrate complex processes at enterprise scale.

Simple
Simplify automation creation and management across multiple domains.

Agentless
Easily integrate with hybrid environments.
**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

Line of business

Devs/DevOps  IT ops  SecOps  Network ops

Edge  Cloud  Datacenter
What makes a platform?

Red Hat Ansible Automation Platform

Content creators
Operators
Domain experts
Users

On-premises
- Automation controller
- Automation hub
- Automation services catalog
- Insights for Ansible Automation Platform

Ansible Cloud Services
- Ansible content domains
  - Infrastructure
    - Linux
    - Windows
  - Cloud
  - Network
  - Security
- Ansible command line

Fueled by an open source community
Red Hat named a Leader in The Forrester Wave™

Infrastructure Automation Platforms, Q3 2020

- Received highest possible score in the criteria of:
  - Deployment functionality
  - Product Vision
  - Partner Ecosystem
  - Supporting products and services
  - Community support
  - Planned product enhancements

▸ “Ansible continues to grow quickly, particularly among enterprises that are automating networks. The solution excels at providing a variety of deployment options and acting as a service broker to a wide array of other automation tools."

▸ “Red Hat’s solution is a good fit for customers that want a holistic automation platform that integrates with a wide array of other vendors’ infrastructure.”

Source:

DISCLAIMER: The Forrester Wave™ is copyrighted by Forrester Research, Inc. Forrester and Forrester Wave™ are trademarks of Forrester Research, Inc. The Forrester Wave™ is a graphical representation of Forrester’s call on a market and is plotted using a detailed spreadsheet with exposed scores, weightings, and comments. Forrester does not endorse any vendor, product, or service depicted in the Forrester Wave™. Red Hat has licensed the use of the Forrester Wave™ from Forrester Research, Inc., and neither Red Hat nor Forrester owns any rights to the Forrester Wave™ trademark or logo.
Ansible automates technologies you use
Time to automate is measured in minutes

<table>
<thead>
<tr>
<th>Cloud</th>
<th>Virt &amp; Container</th>
<th>Windows</th>
<th>Network</th>
<th>Security</th>
<th>Monitoring</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Dynatrace</td>
</tr>
<tr>
<td>Azure</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Datadog</td>
</tr>
<tr>
<td>Digital Ocean</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>LogicMonitor</td>
</tr>
<tr>
<td>Google</td>
<td>Docker</td>
<td></td>
<td>ACLs</td>
<td></td>
<td>F5</td>
</tr>
<tr>
<td>OpenStack</td>
<td>VMware</td>
<td></td>
<td>Files</td>
<td></td>
<td>Fortinet</td>
</tr>
<tr>
<td>Rackspace</td>
<td>RHV</td>
<td></td>
<td>Packages</td>
<td></td>
<td>Juniper</td>
</tr>
<tr>
<td>+more</td>
<td>OpenStack</td>
<td></td>
<td>IIS</td>
<td></td>
<td>IBM</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>RegEdits</td>
<td>Checkpoint</td>
<td>Palo Alto</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Shares</td>
<td>Cisco</td>
<td>Snort</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Services</td>
<td>Arista</td>
<td>+more</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Configs</td>
<td>Aruba</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Users</td>
<td>Cumulus</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Domains</td>
<td>Bigswitch</td>
<td></td>
</tr>
<tr>
<td>Operating Systems</td>
<td>Storage</td>
<td>Netapp</td>
<td></td>
<td>Cisco</td>
<td></td>
</tr>
<tr>
<td>RHEL</td>
<td></td>
<td></td>
<td></td>
<td>Dell</td>
<td></td>
</tr>
<tr>
<td>Linux</td>
<td></td>
<td></td>
<td></td>
<td>Extreme</td>
<td></td>
</tr>
<tr>
<td>Windows</td>
<td></td>
<td></td>
<td></td>
<td>F5</td>
<td></td>
</tr>
<tr>
<td>+more</td>
<td></td>
<td></td>
<td></td>
<td>Lenovo</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>MikroTik</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Juniper</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OpenSwitch</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>+more</td>
<td></td>
</tr>
<tr>
<td>+more</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Operating Systems: RHEL, Linux, Windows, +more

Virt & Container: Docker, VMware, RHV, OpenStack, OpenShift, +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

Storage: Netapp, Red Hat Storage, Infinidat, +more

Operating Systems: RHEL, Linux, Windows, +more
Cloud

Topics Covered:

- Understanding the Ansible Infrastructure
- Check the prerequisites
The lab environment today

- **Drink our own champagne.** Provisioned by, configured by, and managed by Red Hat Ansible Automation Platform. [https://github.com/ansible/workshops](https://github.com/ansible/workshops)

- **Learn with the real thing**
  Every student will have their own fully licensed Red Hat Ansible Tower control node. No emulators or simulators here.

- **Red Hat Enterprise Linux**
  All four nodes are enterprise Linux, showcasing real life use-cases to help spark ideas for what you can automate today.
How does it work?

Provision
- Resources: Subnets, gateways, security groups, SSH keys
- Instances: RHEL, Cisco, Arista, Checkpoint, Windows, etc
- Inventory: Load and sort newly created instances for further automation

Configure
- Ansible environment: Install Ansible Tower, SSH config, user accounts, etc
- Code Server: Configure in-browser text editor and terminal
- DNS: Configure DNS names for all control nodes

Manage
- Login Website: Dynamically create login webpage for students
- Instructor Inventory: Provide inventory and login information and master key
- Log Information: Record student count and instructor for statistics
Exercise 1

Topics Covered:

- Understanding the Ansible Infrastructure
- Check the prerequisites
Create

The automation lifecycle

Content creators

Build

Ansible content experience

Discover

Red Hat cloud / on-premises

Automation hub

Trust

Ansible content domains

Domain experts

Infrastructure

Linux

Windows

Cloud

Network

Security
---
- 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.

```yaml
- name: latest index.html file ...
  template:
    src: files/index.html
    dest: /var/www/html/
```
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 }}
```
Ansible Inventory

The systems that a playbook runs against

What are they?

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
Ansible roles

Reusable automation actions

What are they?

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.

```yaml
---
- name: install and start apache
  hosts: web
  roles:
  - common
  - webservers
```
Collections

Simplified and consistent content delivery

What are they?

Collections are a data structure containing automation content:

- Modules
- Playbooks
- Roles
- Plugins
- Docs
- Tests
---
- name: Install NGINX
  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
Why the Red Hat Ansible Automation Platform?

90+
certified platforms
How Ansible Automation Works

Module code is executed locally on the control node.

Network Devices / API Endpoints

Local Execution

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

Remote Execution

Linux / Windows Hosts
Verify Lab Access

- Follow the steps in to access environment
- Use the IP provided to you, the script only has example IP
- Which editor do you use on command line?
  If you don’t know, we have a short intro
Lab Time
Complete exercise 1-setup now in your lab environment
Exercise 2

Topics Covered:

● Ansible inventories
● Accessing Ansible docs
● Modules and getting help
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
Ansible Inventory

The Basics

An example of a static Ansible inventory including systems with IP addresses as well as fully qualified domain name (FQDN)

[myservers]
10.42.0.2
10.42.0.6
10.42.0.7
10.42.0.8
10.42.0.100
host.example.com
[app1srv]
appserver01 ansible_host=10.42.0.2
appserver02 ansible_host=10.42.0.3

[web]
node-[1:30] ansible_host=10.42.0.[31:60]

[web:vars]
apache_listen_port=8080
apache_root_path=/var/www/mywebdocs/

[all:vars]
ansible_user=kev
ansible_ssh_private_key_file=/home/kev/.ssh/id_rsa
[app1srv]
appserver01 ansible_host=10.42.0.2
appserver02 ansible_host=10.42.0.3

[web]
node-[1:30] ansible_host=10.42.0.[31:60]

[web:vars]
apache_listen_port=8080
apache_root_path=/var/www/mywebdocs/

[all:vars]
ansible_user=ender
ansible_ssh_private_key_file=/home/ender/.ssh/id_rsa
[nashville]
  bnaapp01
  bnaapp02

[atlanta]
  atlapp03
  atlapp04

[south:children]
  atlanta
  nashville
  hsvapp05
Accessing the Ansible docs

With the use of the latest command utility ansible-navigator, one can trigger access to all the modules available to them as well as details on specific modules.

A formal introduction to ansible-navigator and how it can be used to run playbooks in the following exercise.
Accessing the Ansible docs

Aside from listing a full list of all the modules, you can use ansible-navigator to provide details about a specific module.

In this example, we are getting information about the user module.

```bash
$ ansible-navigator doc user -m stdout

> ANSIBLE.BUILTIN.USER
(/usr/lib/python3.8/site-packages/ansible/modules/user.py)

Manage user accounts and user attributes. For Windows targets, use the [ansible.windows.win_user] module instead.
```
Bash vs. Ansible

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

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

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

- **name**: Add mssql-tools to $PATH
  - **lineinfile**:
    - **path**: "{{ item }}"
    - **line**: `export PATH="$PATH:/opt/mssql-tools/bin"
  - **loop**:
    - ~/.bash_profile
    - ~/.bashrc
Lab Time
Complete exercise **2-thebasics** now in your lab environment
Exercise 3

Topics Covered:

- Playbooks basics
- Running a playbook
- 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
---
- 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 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
A playbook run

Where it all starts

- A playbook is interpreted and run against one or multiple hosts - task by task. The order of the tasks defines the execution.

- In each task, the module does the actual work.
Running an Ansible Playbook

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

Bye ansible-playbook, 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
**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>
# ansible-navigator

## Common subcommands

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
<th>CLI Example</th>
<th>Colon command within TUI</th>
</tr>
</thead>
<tbody>
<tr>
<td>collections</td>
<td>Explore available collections</td>
<td>ansible-navigator collections --help</td>
<td>:collections</td>
</tr>
<tr>
<td>config</td>
<td>Explore the current ansible configuration</td>
<td>ansible-navigator config --help</td>
<td>:config</td>
</tr>
<tr>
<td>doc</td>
<td>Review documentation for a module or plugin</td>
<td>ansible-navigator doc --help</td>
<td>:doc</td>
</tr>
<tr>
<td>images</td>
<td>Explore execution environment images</td>
<td>ansible-navigator images --help</td>
<td>:images</td>
</tr>
<tr>
<td>inventory</td>
<td>Explore and inventory</td>
<td>ansible-navigator inventory --help</td>
<td>:inventory</td>
</tr>
<tr>
<td>replay</td>
<td>Explore a previous run using a playbook artifact</td>
<td>ansible-navigator replay --help</td>
<td>:replay</td>
</tr>
<tr>
<td>run</td>
<td>Run a playbook</td>
<td>ansible-navigator run --help</td>
<td>:run</td>
</tr>
<tr>
<td>welcome</td>
<td>Start at the welcome page</td>
<td>ansible-navigator welcome --help</td>
<td>:welcome</td>
</tr>
</tbody>
</table>
Lab Time
Complete exercise **3-playbooks** now in your lab environment
Exercise 4

Topics Covered:

- Working with variables
- What are facts?
---

- name: variable playbook test
  hosts: localhost

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

  tasks:
    - name: print out var_three
      debug:
        msg: "{{ var_three }}"
---
- name: variable playbook test
  hosts: localhost

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

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

ansible is awesome
Ansible Facts

- Just like variables, really...
- ... but: coming from the host itself!
- Check them out with the setup module

```yaml
tasks:
  - name: Collect all facts of host
    setup:
      gather_subset:
        - 'all'
```
---
- name: facts playbook
  hosts: localhost

  tasks:
  - name: Collect all facts of host
    setup:
      gather_subset:
        - 'all'

$ansible-navigator run playbook.yml
<table>
<thead>
<tr>
<th>PLAY NAME</th>
<th>OK</th>
<th>CHANGED</th>
<th>UNREACHABLE</th>
<th>FAILED</th>
<th>SKIPPED</th>
<th>IGNORED</th>
<th>IN PROGRESS</th>
<th>TASK COUNT</th>
<th>PROGRESS</th>
</tr>
</thead>
<tbody>
<tr>
<td>facts playbook</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>COMPLETE</td>
</tr>
</tbody>
</table>

RESULT

<table>
<thead>
<tr>
<th>HOST</th>
<th>NUMBER</th>
<th>CHANGED</th>
<th>TASK</th>
<th>TASK ACTION</th>
<th>DURATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>localhost</td>
<td>0</td>
<td>False</td>
<td>Gathering Facts</td>
<td>gather_facts</td>
<td>1s</td>
</tr>
<tr>
<td>localhost</td>
<td>1</td>
<td>False</td>
<td>Collect all facts of host</td>
<td>setup</td>
<td>1s</td>
</tr>
</tbody>
</table>

PLAY [facts playbook:1]

```
ansible_facts:
ansible_all_ipv4_addresses:
- 10.0.2.100
ansible_all_ipv6_addresses:
- fe80::1caa:f0ff:fe15:23c4
```
$ tree ansible-files/

```
  ├── deploy_index_html.yml
  ├── files
  │   ├── dev_web.html
  │   └── prod_web.html
  │
  │
  ├── group_vars
  │   └── web.yml
  │
  │
  └── host_vars
      └── node2.yml
```
Ansible Inventory - Managing Variables In Files

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

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

- name: copy web.html
  copy:
    src: "{{ stage }}_web.html"
    dest: /var/www/html/index.html
Lab Time
Complete exercise `4-variables` now in your lab environment
Exercise 5

Topics Covered:

• Surveys
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 Ansible 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.
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.
Lab Time
Complete exercise 5-surveys now in your lab environment
Exercise 6

Topics Covered:

- Red Hat Enterprise Linux System Roles
Automation Hub and Ansible Galaxy

Physical Site → Ansible Content
Roles & Collections → Physical Site
Linux System Roles Collection

- Consistent user interface to provide settings to a given subsystem that is abstract from any particular implementation

Examples

kdump  network  selinux  timesync
---

- name: example system roles playbook
  hosts: web

  tasks:

  - name: Configure Firewall
    include_role:
      name: linux-system-roles.firewall

  - name: Configure Timesync
    include_role:
      name: redhat.rhel_system_roles.timesync

  timesync role is referenced from the RHEL System Roles Collection
Lab Time
Complete exercise 6-system-roles now in your lab environment
Exercise 7

Topics Covered:

- Red Hat Insights intro
- Insights integration
What is Red Hat Insights?
Helping you better manage your hybrid and cloud environments

Predicting risks
Recommending actions
Analyzing costs
What does Red Hat Insights do?

A cloud analytics platform that helps you better manage your hybrid and cloud environments

- **Gathers** configuration and utilization data from your Red Hat® products
- **Analyzes** the data based on Red Hat knowledge and expertise
- **Generates** and prioritizes insights for you to take action
How does Red Hat Insights help me?

Use Red Hat’s expertise and knowledge to evaluate your systems

- Configuration review to make sure systems are setup correctly
- Centralized view of all CVEs, patches, and compliance risks
- Easily identify interoperability issues from the hypervisor or cloud, through the OS, and through the application stack
- Identify drift to make sure systems are the same
- Know how many subscriptions you are using in seconds
Red Hat Insights for Red Hat Ansible Automation Platform

For all your hybrid-cloud challenges

Red Hat Insights

Efficient hybrid-cloud scale

Business
- Track and optimize spending
  - Savings planner
  - Automation calculator

Operations
- Improve stability and performance
  - Advisor
  - Drift
  - Automation analytics

Security
- Reduce risk
  - Policies

Red Hat Ansible Automation Platform
Savings Planner

Create a plan that details:

- How long manual work takes
- How often manual work is performed
- How many hosts are impacted
- List of tasks to be automated

Results in estimated time and cost savings of automation.
Savings planner: Pre-plan your automation savings

Statistics

Total savings
$1,696,415.00

- Savings from automating this plan
- Costs from creating, maintaining and running the automation
- Cumulative savings over time

Automation formula

Manual cost for template = (time for a manual run on one host in hours * (sum of all hosts across all job runs)) * cost per hour
Automation cost for template = cost of automation per hour * sum of total elapsed hours for a template
Savings = Sum of (manual cost - automation cost) across all templates
Automation Calculator

Measures the success of your automation

- See most to least save tasks in terms of $$
- Determine ROI of your automation
Automation calculator: Calculate your ROI from automation using the data gathered by Red Hat Insights analytics.

Total savings: $2,191,846.45

Calculate your automation:
- Manual cost of automation (e.g., average salary of mid-level Software Engineer)
  - $50/hr
- Automated process cost
  - $20/hr

Enter the time it takes to run the following templates manually:
- clean long-running aws instances (sean & yanzai)
  - 60 minutes
Automation Analytics

Provides a detailed view of automation activity across your organization

- Health Notifications
- Organization Statistics
**Automation analytics:** Gather information about how automation is being used in your environment

- **Clusters**
  - Job
  - Filter by job type
  - Post 30 days

**Job status**

- Jobs across all clusters

**Top workflows**

<table>
<thead>
<tr>
<th>Workflow</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS - workshop uptime turnoff workflow</td>
<td>715</td>
</tr>
<tr>
<td>AWS - stop untagged instances</td>
<td>714</td>
</tr>
</tbody>
</table>

**Top templates**

<table>
<thead>
<tr>
<th>Template</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS - stop long running instances</td>
<td>4267</td>
</tr>
<tr>
<td>AWS - stop untagged instances</td>
<td>3571</td>
</tr>
</tbody>
</table>

**Top modules**

<table>
<thead>
<tr>
<th>Module</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>ec2</td>
<td>3579</td>
</tr>
<tr>
<td>debug</td>
<td></td>
</tr>
</tbody>
</table>
Advisor

Analyzes your automation controller deployments for issues with:

- Availability
- Performance
- Stability
- Security
## Advisor: Availability, performance, stability, and security risk analysis

### Advisor recommendations

<table>
<thead>
<tr>
<th>Name</th>
<th>Added</th>
<th>Category</th>
<th>Total risk</th>
<th>Risk of change</th>
<th>Systems</th>
<th>Ansible</th>
</tr>
</thead>
<tbody>
<tr>
<td>New Ansible Engine packages are inaccessible when dedicated Ansible repo is not enabled</td>
<td>3 years ago</td>
<td>Availability</td>
<td>Moderate</td>
<td>Very Low</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Ansible Tower is out-of-scope for Ansible support due to running with an unsupported version of PostgreSQL database</td>
<td>2 months ago</td>
<td>Availability</td>
<td>Moderate</td>
<td>Moderate</td>
<td>2</td>
<td>No</td>
</tr>
<tr>
<td>Filesystems get filled up when keeping runtime directories in Ansible Tower node</td>
<td>2 months ago</td>
<td>Availability</td>
<td>Moderate</td>
<td>Moderate</td>
<td>1</td>
<td>No</td>
</tr>
</tbody>
</table>

Files get filled up because the temporary runtime directories keep consuming disk space when "AWX_CLEANUP_PATHS="False" is configured.

Knowledgebase article [Red Hat Insights for Red Hat Ansible Automation Platform](#)

### Total risk

The total risk of this remediation is **moderate**, based on the combination of likelihood and impact to remediate.

- **Medium likelihood**
- **Medium impact**

### Risk of change
Drift

Lets you created baselines and compare systems to identify:

- Differences from a baseline
- Differences from other systems
- Differences from historical system profiles
Drift: Create baselines and compare systems to monitor for differences as systems drift

Red Hat Insights for Red Hat Ansible Automation Platform
Policies

Lets you create your own internal policies based on facts collected by Insights

- Quickly identify situations that exist that shouldn’t
Policies: Define and monitor against your own policies to identify misalignment

Tower Upgrade

Enabled

Description
Notify if a system is running the ansible-tower service. If so, this system should be targeted for upgrade to automation controller

Last updated 18 Aug 2021 | Created 18 Aug 2021

Conditions
facts.enabled_services contains ['ansible-tower']

Trigger actions
⚠️ Send a notification

Recent trigger history

<table>
<thead>
<tr>
<th>Date</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>18 Aug 2021 19:16:07 UTC</td>
<td>twr383.demorehat.com</td>
</tr>
</tbody>
</table>
Reports

Provides executive summaries of automation across the organization

- Defined in collaboration with existing customers
  - Hosts changed by job template
  - Changes made by job template
  - Job template run rate
- Track automation KPIs and identify discrepancies
**Reports:** Provide executive summaries of automation across the organization

**Changes made by job template**

The total count of changes made by each job template in a specified time window. You can use this report to ensure the correct number of changes are made per hostname, as well as see which job templates are doing the most changes to your infrastructure.
Red Hat Insights
Included with your Red Hat Enterprise Linux subscription

Assesses
customer’s Red Hat environments

Remediates
findings with prescriptive remediation steps or an Ansible playbook

Insights
rule contributions directly from Red Hat subject matter experts

Identifying risks for Availability, performance, stability and security
Insights plans with Ansible playbooks

Solve common issues through Ansible Automation

- Systems reboot:
  - 6: No reboot
  - 0: Reboot required
  - Auto reboot

- Playbook details:
  - Created by: John Spinks
  - Created: a minute ago
  - Last modified by: John Spinks

- Dnsmasq with listening processes vulnerable to remote code execution via crafted DNS requests (CVE-2017-14491)
  - Update dnsmasq package and restart related service(s)

- Systems
  - ic3.example.com
  - ic4.example.com
  - ic6.example.com
  - ic7.example.com
ANSIBLE & INSIGHTS

While Insights includes Ansible playbooks for risks, Insights alone can’t perform remediation of the risks.

Insights
- Insights provides Ansible Playbooks for resolving many common risks.
- Dynamically generates Ansible Playbooks for risk remediation
- Playbooks can be downloaded and run via `ansible-playbook` or Satellite

Insights connected to Ansible Controller
- View identified risks in the Tower inventory
- Execute generated Ansible Playbook as a Tower job
- Use Tower for enterprise risk remediation
Where to go next

Learn more
- Workshops
- Documents
- Youtube
- Twitter

Get started
- Evals
- cloud.redhat.com

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