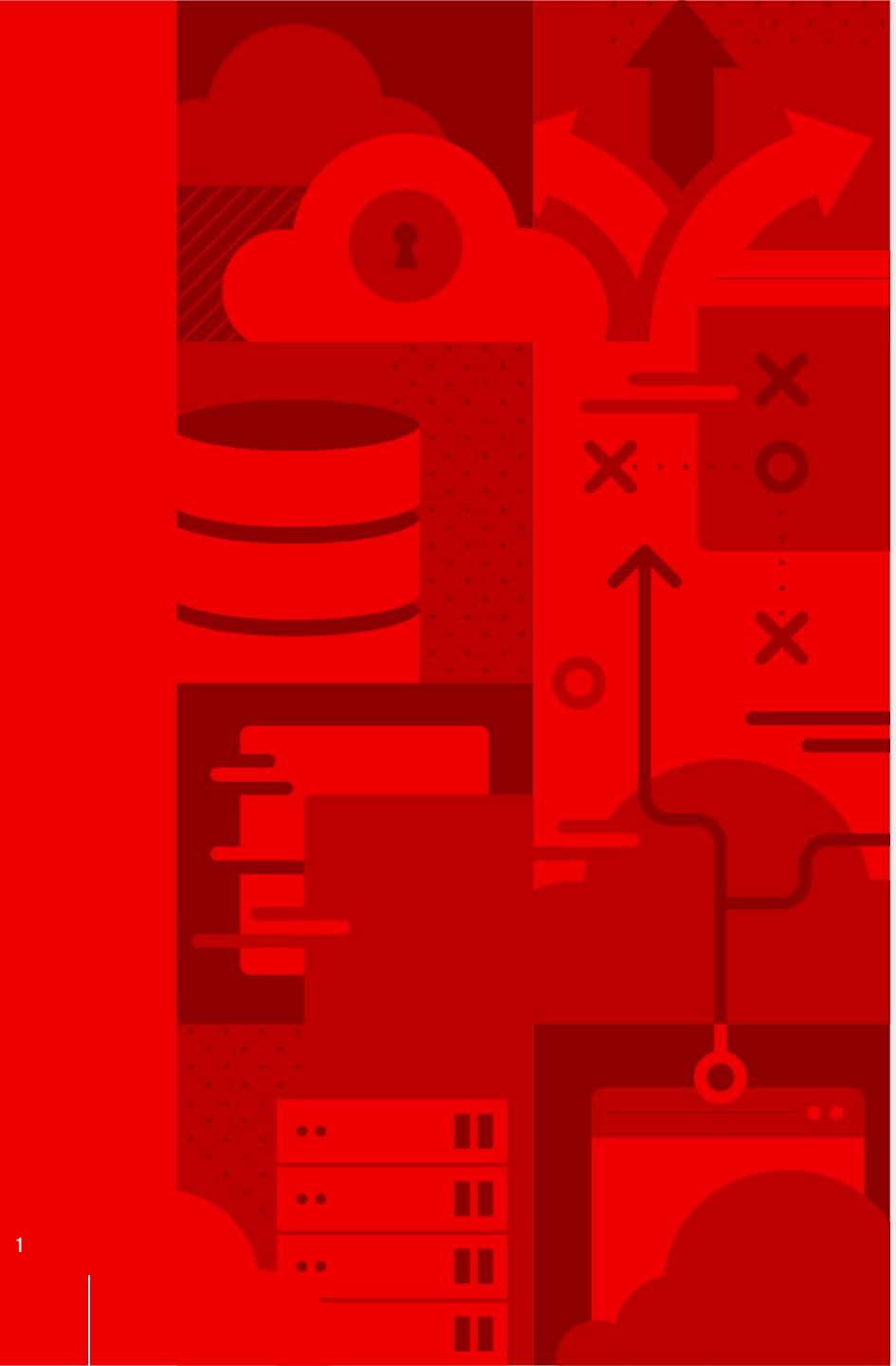


Red Hat Ansible Automation

Overview

Raed Soliman
Solution Architect, Management Tiger Team

Red Hat

Value of Automation

How do you modernize existing I.T. while shifting investment to innovation?

Optimize the IT you have

Integrate apps, data, & processes

Add & manage cloud infrastructure

Build more modern applications

Automate & manage IT

Red Hat

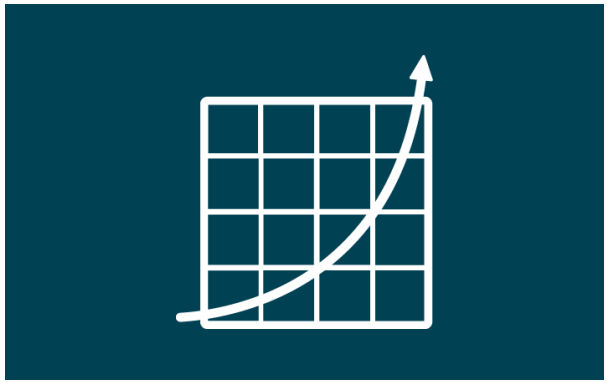
ROI Value of Automation

Operational Efficiency
Reduced lead times, automated provisioning
“It’s all about repeatability and consistency”

Cost Avoidance
Saved $390k in additional hardware costs

System Reconfiguration and Incident Response
Reduced staff time requirements by 94%

Security Updates
Reduced staff time requirements by 80%



<3 months payback

146% ROI

Source: Findings for an Organization interviewed for The Total Economic Impact™ Of Red Hat Ansible Automation, a June 2018 commissioned study conducted by Forrester Consulting on behalf of Red Hat.

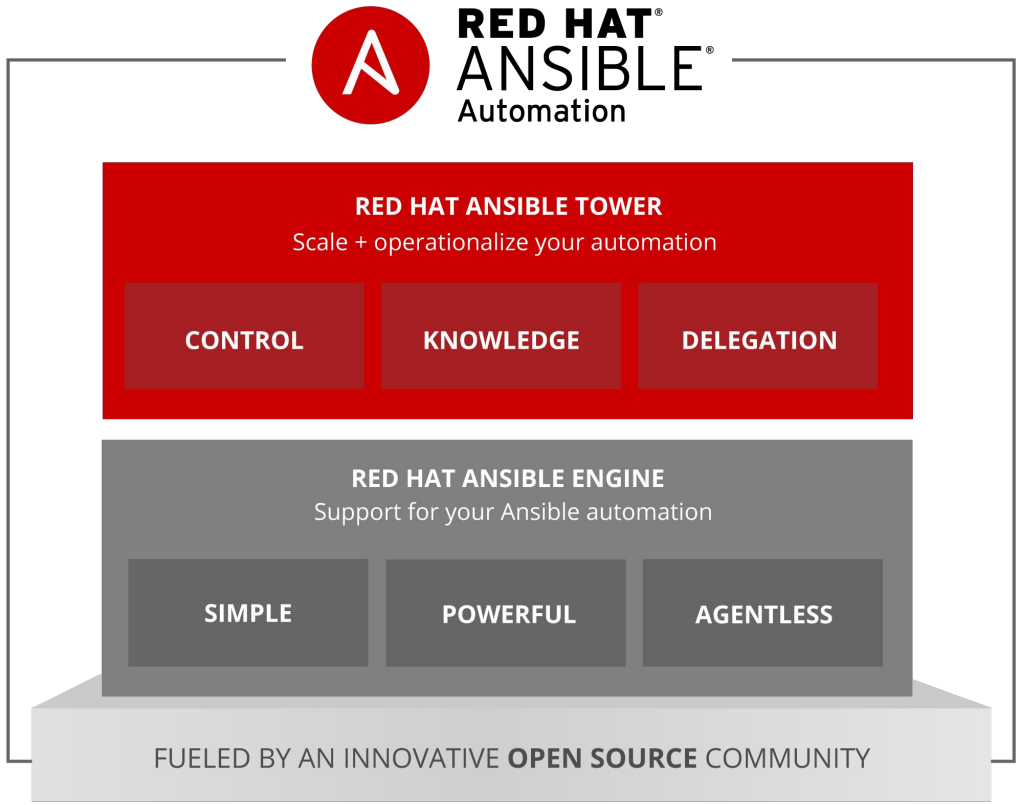
Red Hat

Red Hat Ansible Automation

Ansible is a **simple automation language** that can perfectly describe an IT application infrastructure in **Ansible Playbooks.**

RedHat Ansible Engine is an **automation engine** that runs Ansible Playbooks.

Red Hat Ansible Tower is an **enterprise framework** for controlling, securing and managing your Ansible automation with a **UI & RESTful API.**



Red Hat

The Ansible Way

CROSS PLATFORM

Agentless support for all major OS variants, physical, virtual, cloud and network devices.

HUMAN READABLE

Perfectly describe and document every aspect of your application environment.

PERFECT DESCRIPTION OF APPLICATION

Every change can be made by Playbooks, ensuring everyone is on the same page.

VERSION CONTROLLED

Playbooks are plain-text. Treat them like code in your existing version control.

DYNAMIC INVENTORIES

Capture all the servers 100% of the time, regardless of infrastructure, location, etc.

ORCHESTRATION PLAYS WELL WITH OTHERS

Every change can be made by Playbooks, ensuring everyone is on the same page.

Why Ansible?

SIMPLE

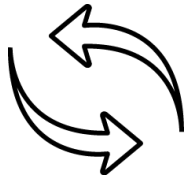
Human readable automation

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

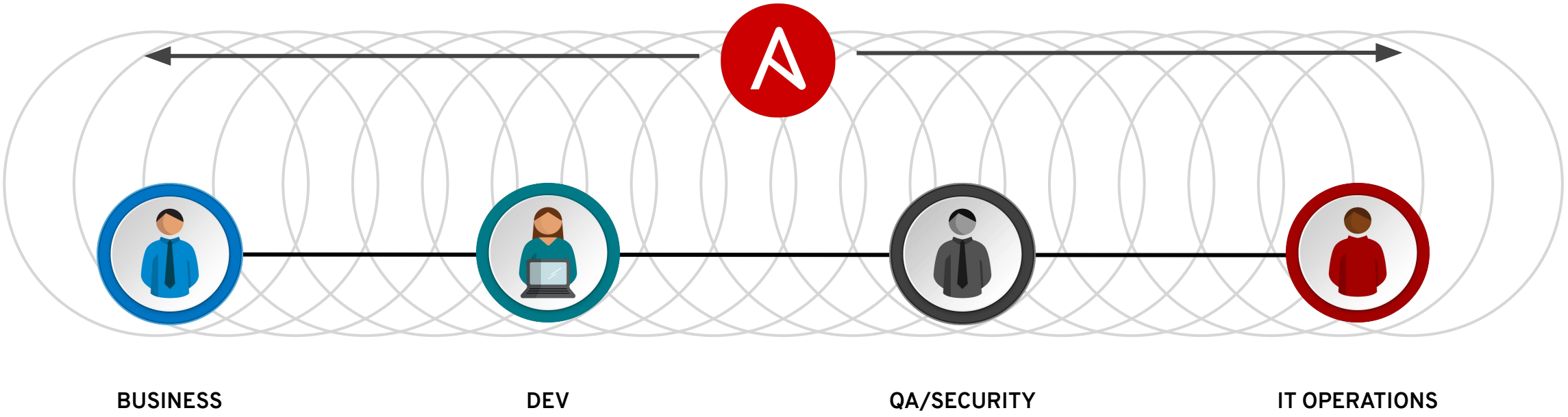
Uses OpenSSH & WinRM

No agents to exploit or update

Get started immediately

More efficient & more secure

Ansible Is The Universal Language



Ansible is the first **automation language** that can be read and written across IT.

Ansible is the only **automation engine** that can automate the entire application lifecycle and continuous delivery pipeline.

Red Hat

What Can Ansible Do?

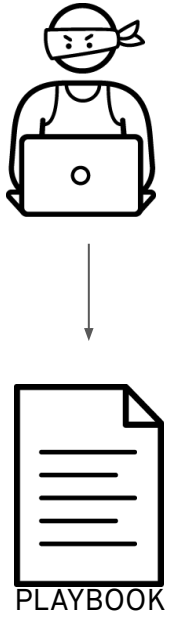
USE ANSIBLE FOR:

- Orchestration
- Configuration Management
- Application Deployment
- Provisioning
- Continuous Delivery
- Security and Compliance

USE ANSIBLE ON:

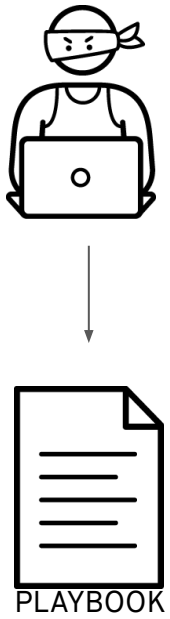
- Servers
- Clouds
- Storage
- Applications
- Containers
- Clouds
- Firewalls
- Load Balancers
- Network Devices
- .... And More

How Ansible Works



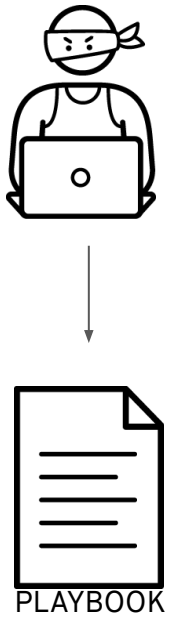
Red Hat

How Ansible Works




```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

How Ansible Works

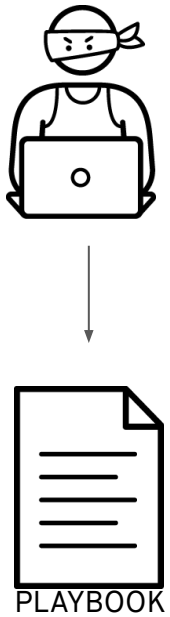



```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

Ansible Playbooks:

- **Written in YAML**
- A collection of Plays
- Each play is a collection of Tasks
- Tasks invoke ansible modules
- Tasks are executed sequentially
- Runs against a defined set of targets

How Ansible Works

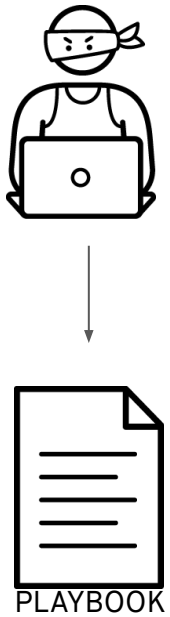



```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

Ansible Playbooks:

- Written in YAML
- **A collection of Plays**
- Each play is a collection of Tasks
- Tasks invoke ansible modules
- Tasks are executed sequentially
- Runs against a defined set of targets

How Ansible Works

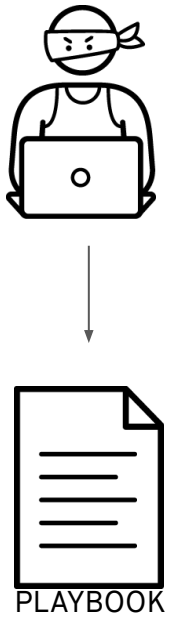



```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

Ansible Playbooks:

- Written in YAML
- A collection of Plays
- **Each play is a collection of Tasks**
- Tasks invoke ansible modules
- Tasks are executed sequentially
- Runs against a defined set of targets

How Ansible Works

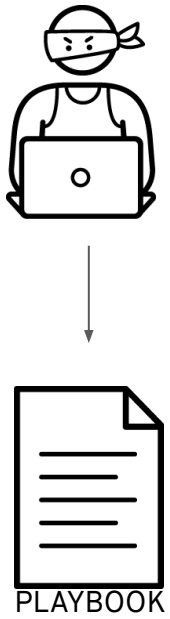



```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

Ansible Playbooks:

- Written in YAML
- A collection of Plays
- Each play is a collection of Tasks
- **Tasks invoke ansible modules**
- Tasks are executed sequentially
- Runs against a defined set of targets

How Ansible Works




```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

1

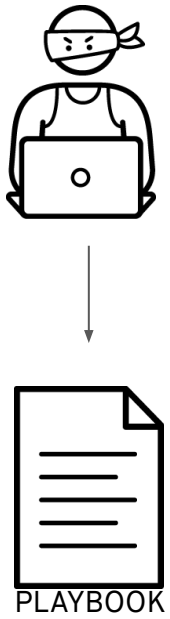
2

3

Ansible Playbooks:

- Written in YAML
- A collection of Plays
- Each play is a collection of Tasks
- Tasks invoke ansible modules
- **Tasks are executed sequentially**
- Runs against a defined set of targets

How Ansible Works

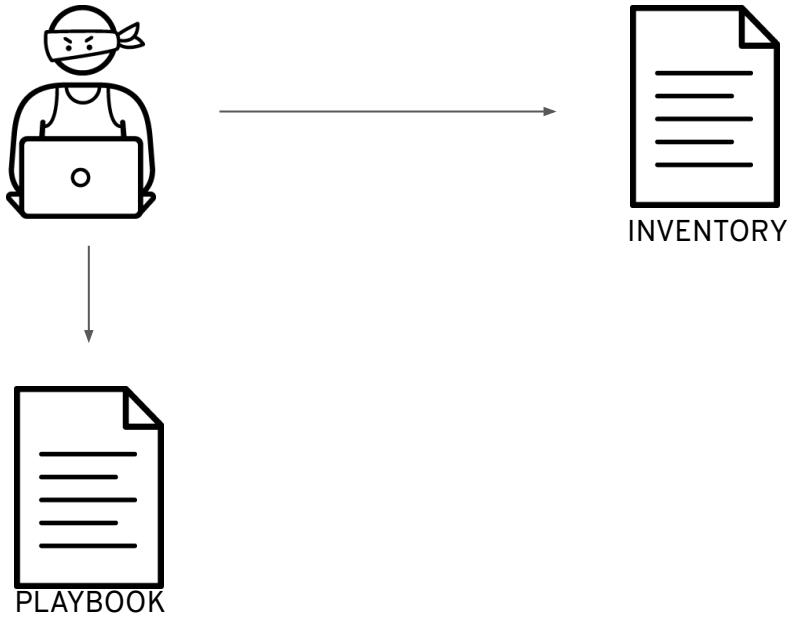



```
---
- name: install and start apache
  hosts: web
  become: yes
  vars:
    http_port: 80

  tasks:
  - name: httpd package is present
    yum:
      name: httpd
      state: latest

  - name: index.html file is present
    copy:
      src: files/index.html
      dest: /var/www/html/

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

Ansible Playbooks:

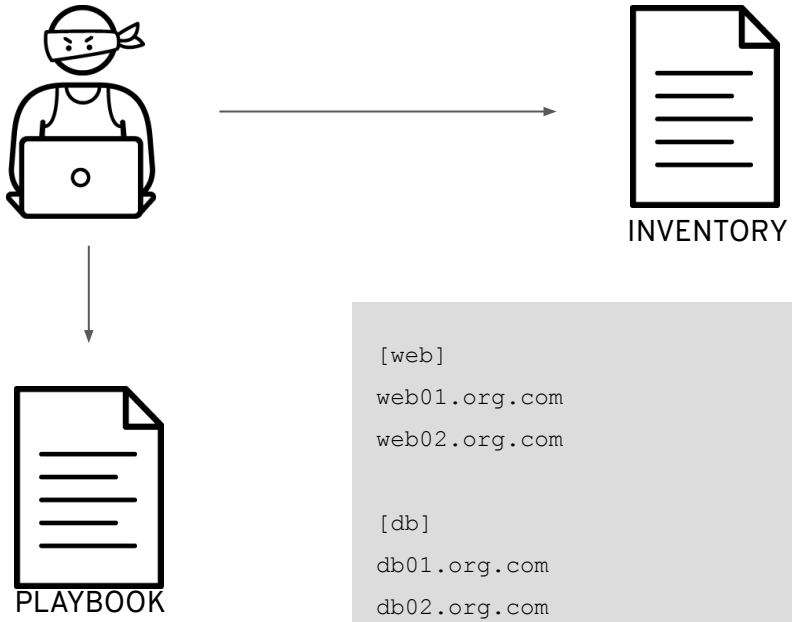
- Written in YAML
- A collection of Plays
- Each play is a collection of Tasks
- Tasks invoke ansible modules
- Tasks are executed sequentially
- **Runs against a defined set of targets**

How Ansible Works



Red Hat

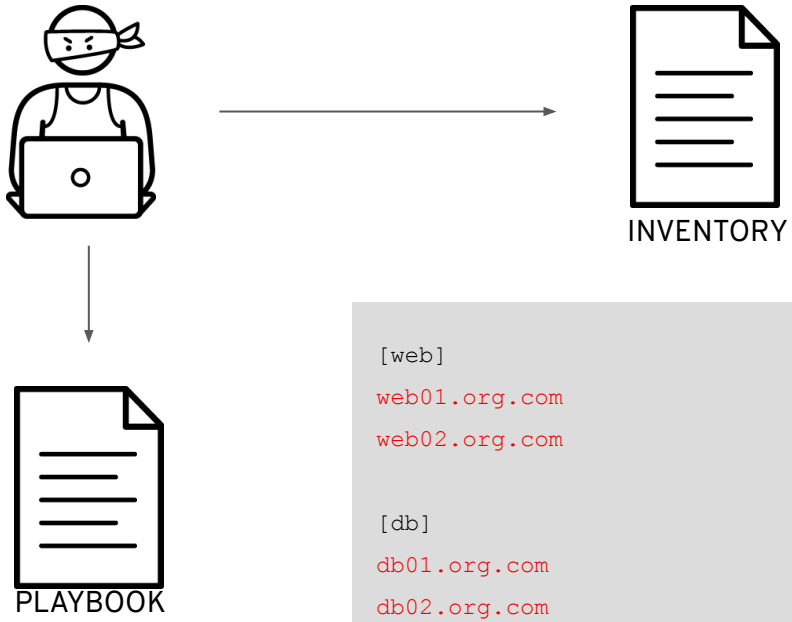
How Ansible Works




```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```


Red Hat

How Ansible Works

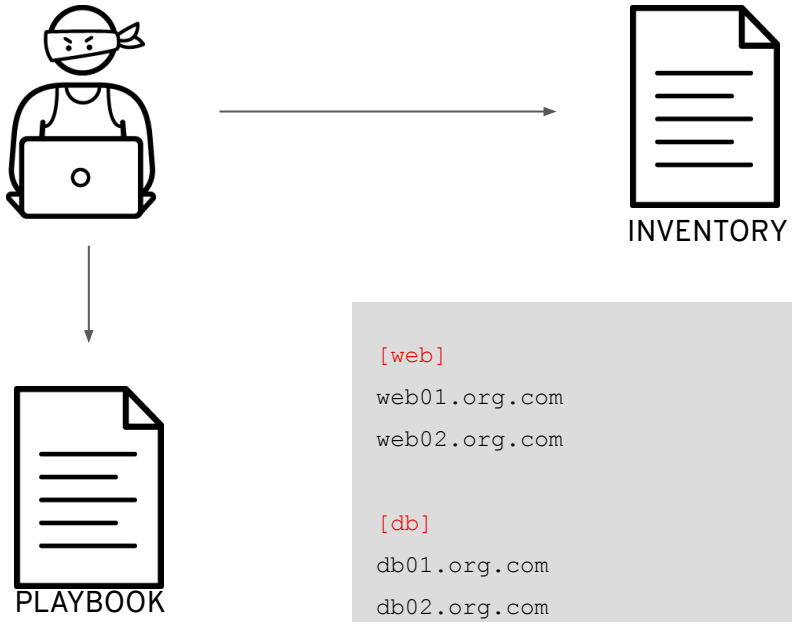



```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```

Ansible Inventory:

- **Defines hosts and host groups**
- Can be defined manually
- Public/Private Cloud:
  - OpenStack, VMware, EC2, Rackspace, GCE, Azure, Spacewalk, Cobbler
- CIs from CMDB

How Ansible Works

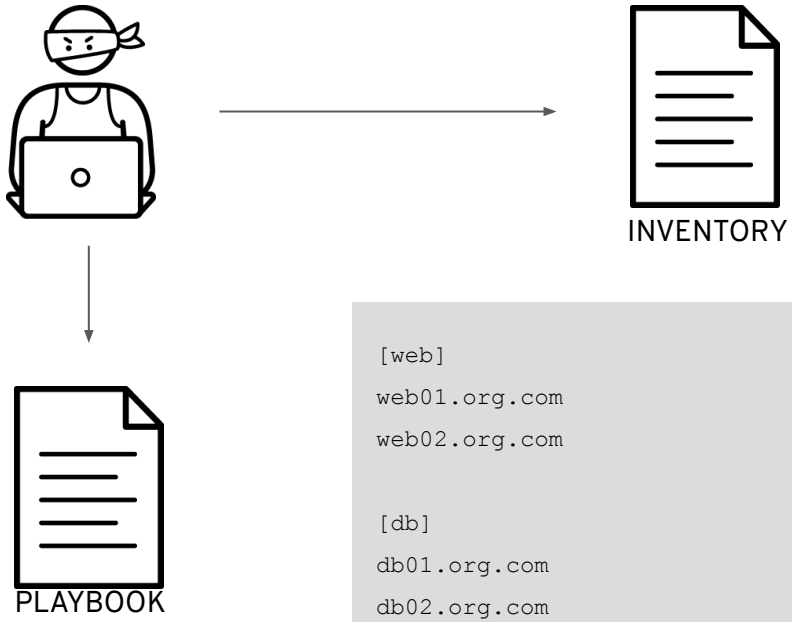



```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```

Ansible Inventory:

- **Defines hosts and host groups**
- Can be defined manually
- Public/Private Cloud:
  - OpenStack, VMware, EC2, Rackspace, GCE, Azure, Spacewalk, Cobbler
- CIs from CMDB

How Ansible Works

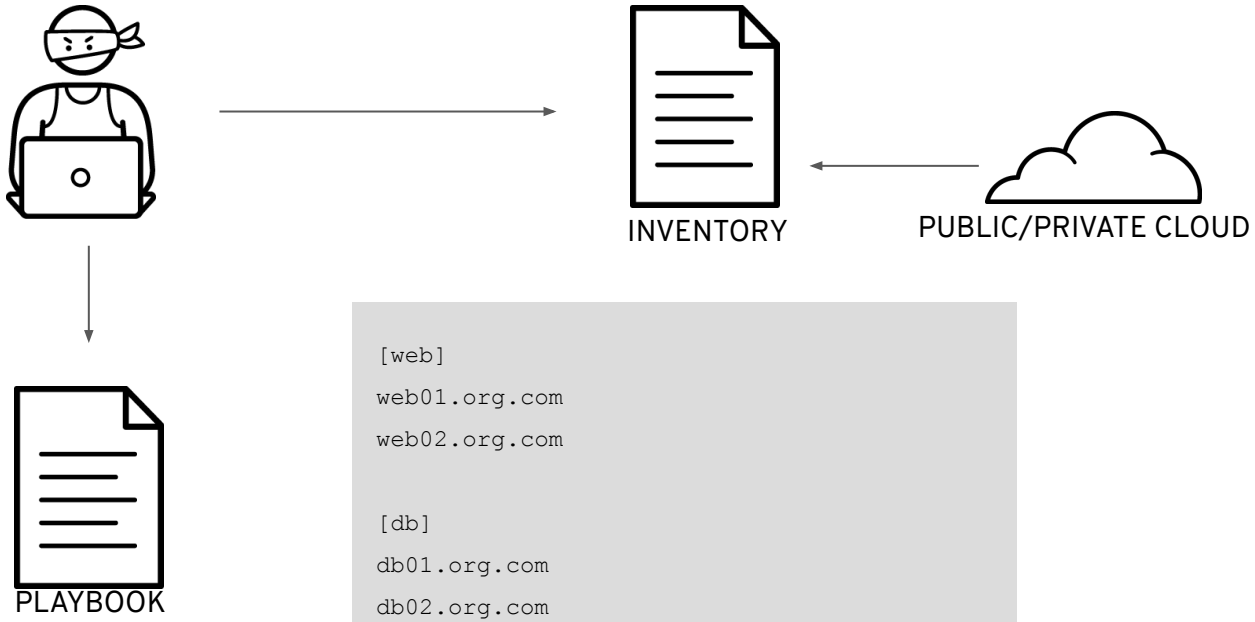



```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```

Ansible Inventory:

- Defines hosts and host groups
- **Can be defined manually**
- Public/Private Cloud:
  - OpenStack, VMware, EC2, Rackspace, GCE, Azure, Spacewalk, Cobbler
- CIs from CMDB

How Ansible Works



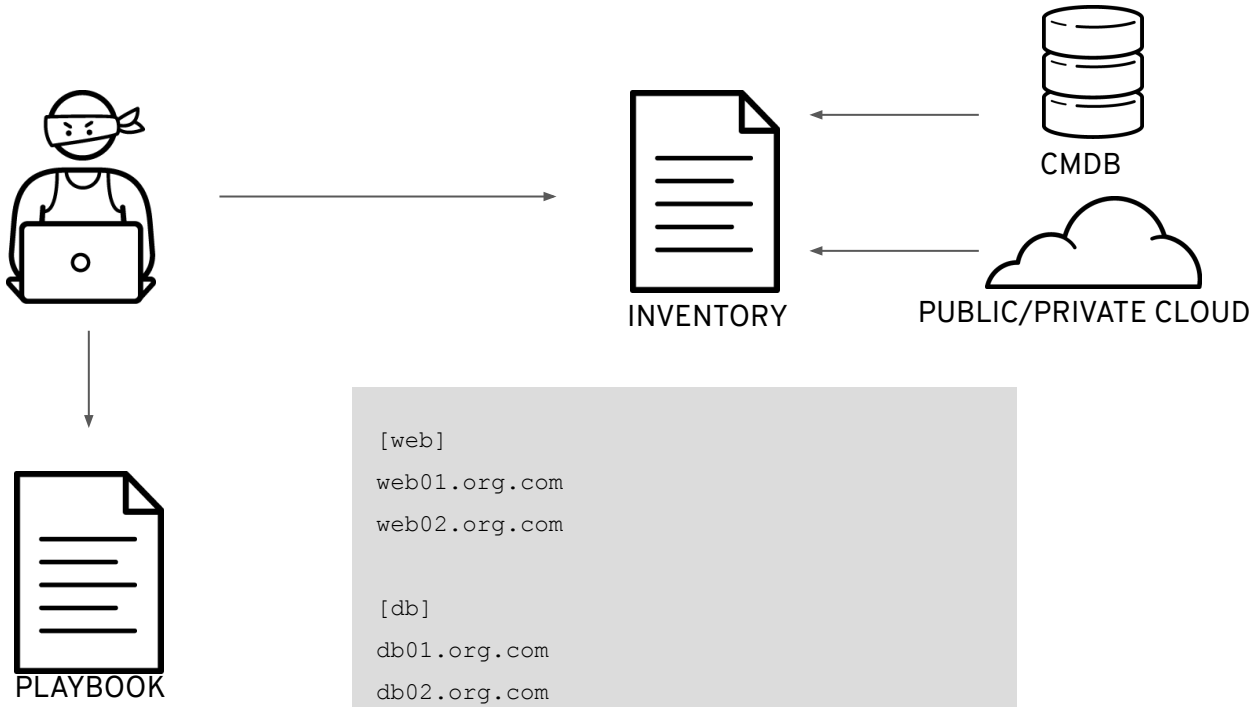

```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```

Ansible Inventory:

- Defines hosts and host groups
- Can be defined manually
- **Public/Private Cloud:**
  - **OpenStack, VMware, EC2, Rackspace, GCE, Azure, Spacewalk, Cobbler**
- CIs from CMDB

Red Hat

How Ansible Works

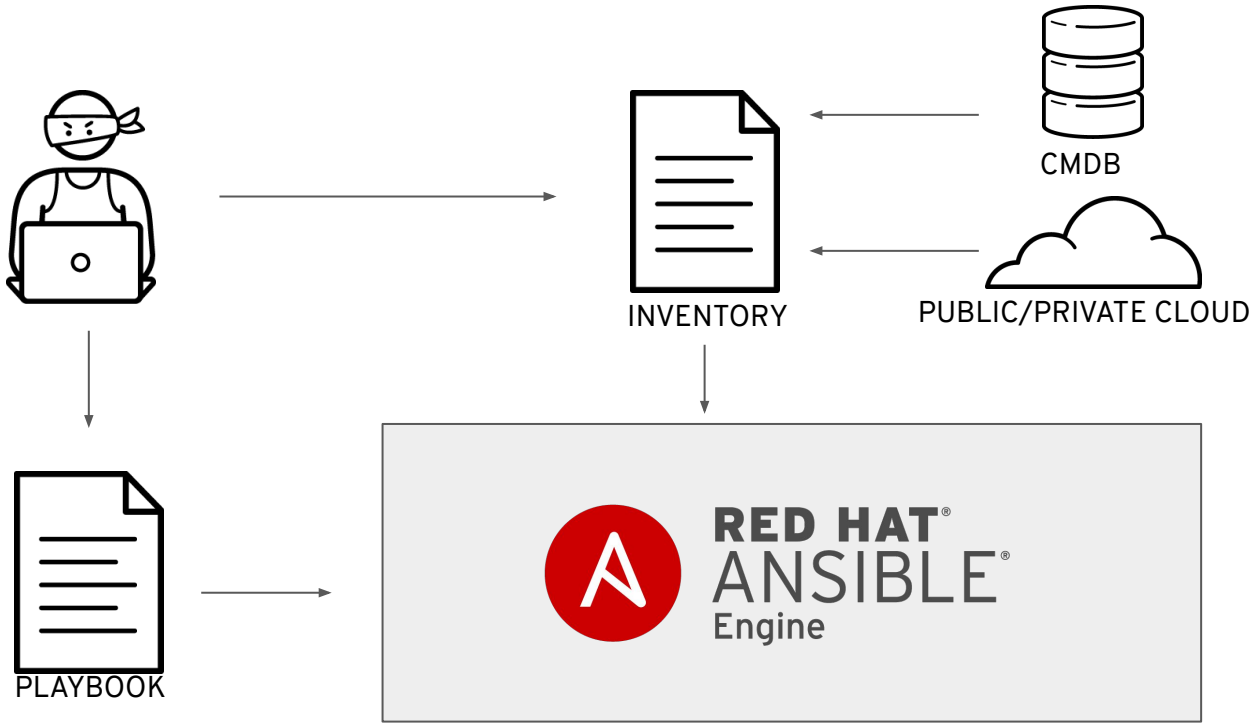



```
[web]
web01.org.com
web02.org.com

[db]
db01.org.com
db02.org.com
134.33.23.345
```

Ansible Inventory:

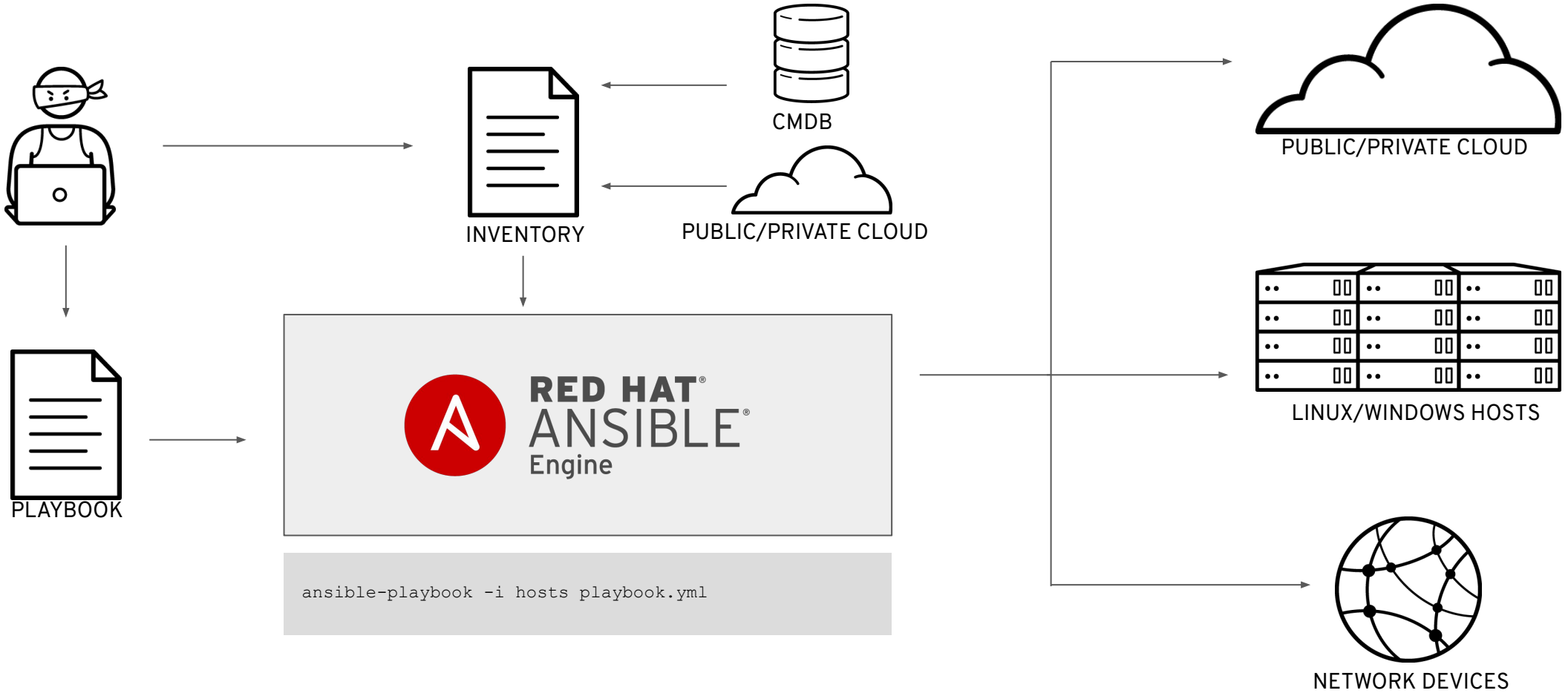
- Defines hosts and host groups
- Can be defined manually
- Public/Private Cloud:
  - OpenStack, VMware, EC2, Rackspace, GCE, Azure, Spacewalk, Cobbler
- **CIs from CMDB**

How Ansible Works



Red Hat

How Ansible Works



Red Hat

Ansible for Windows

ANSIBLE WINDOWS AUTOMATION

Use Ansible to deploy and manage Windows systems and applications.

100+
Windows Modules

1,300+
Powershell DSC resources

ansible.com/windows

Red Hat

Ansible for Windows: Example

```
---
- hosts: new_servers
  tasks:
  - name: ensure common OS updates are current
    win_updates:
    register: update_result

  - name: ensure domain membership
    win_domain_membership:
      dns_domain_name: contoso.corp
      domain_admin_user: '{{ domain_admin_username }}'
      domain_admin_password: '{{ domain_admin_password }}'
      state: domain
    register: domain_result

  - name: reboot and wait for host if updates or domain change require it
    win_reboot:
    when: update_result.reboot_required or domain_result.reboot_required

  - name: ensure local admin account exists
    win_user:
      name: localadmin
      password: '{{ local_admin_password }}'
      groups: Administrators

  - name: ensure common tools are installed
    win_chocolatey:
      name: '{{ item }}'
    with_items: ['sysinternals', 'googlechrome']
```

Ansible for Network Automation

ANSIBLE NETWORK AUTOMATION

50
Network Platforms

700+
Network Modules

12*
Galaxy Network Roles

ansible.com/networking
galaxy.ansible.com/ansible-network

Red Hat

Ansible for Networking: Example

```
---
- name: configure ios interface
  hosts: ios01
  tasks:
    - name: collect device running-config
      ios_command:
        commands: show running-config interface GigabitEthernet0/2
        provider: “{{ cli }}”
      register: config

    - name: administratively enable interface
      ios_config:
        lines: no shutdown
        parents: interface GigabitEthernet0/2
        provider: “{{ cli }}”
      when: ‘”shutdown” in config.stdout[0]‘

    - name: verify operational status
      ios_command:
        commands:
           - show interfaces GigabitEthernet0/2
           - show cdp neighbors GigabitEthernet0/2 detail
        waitfor:
           - result[0] contains ‘line protocol is up’
           - result[1] contains ‘iosxr03’
           - result[1] contains ’10.0.0.42’
        provider: “{{ cli }}”
```

Automate All The Things

CLOUD

AWS
Azure
Digital Ocean
Google
OpenStack
Rackspace
+more

VIRT & CONTAINER

Docker
VMware
RHV
OpenStack
OpenShift
+more

WINDOWS

ACLs
Files
Packages
IIS
Regedits
Shares
Services
Configs
Users
Domains
+more

NETWORK

Arista
A10
Cumulus
Bigswitch
Cisco
Cumulus
Dell
F5
Juniper
Palo Alto
OpenSwitch
+more

DEVOPS

Jira
GitHub
Vagrant
Jenkins
Bamboo
Atlassian
Subversion
Slack
Hipchat
+more

MONITORING

Dynatrace
Airbrake
BigPanda
Datadog
LogicMonitor
Nagios
New Relic
PagerDuty
Sensu
StackDriver
Zabbix
+more

OPERATING SYSTEMS

RHEL and Linux
UNIX
Windows
+more

STORAGE

NetApp
Red Hat Storage
Infinidat
+more

Jumpstart Your Automation

15,000 ROLES AT YOUR DISPOSAL

Reusable Roles and Container Apps that allow you to do more, faster

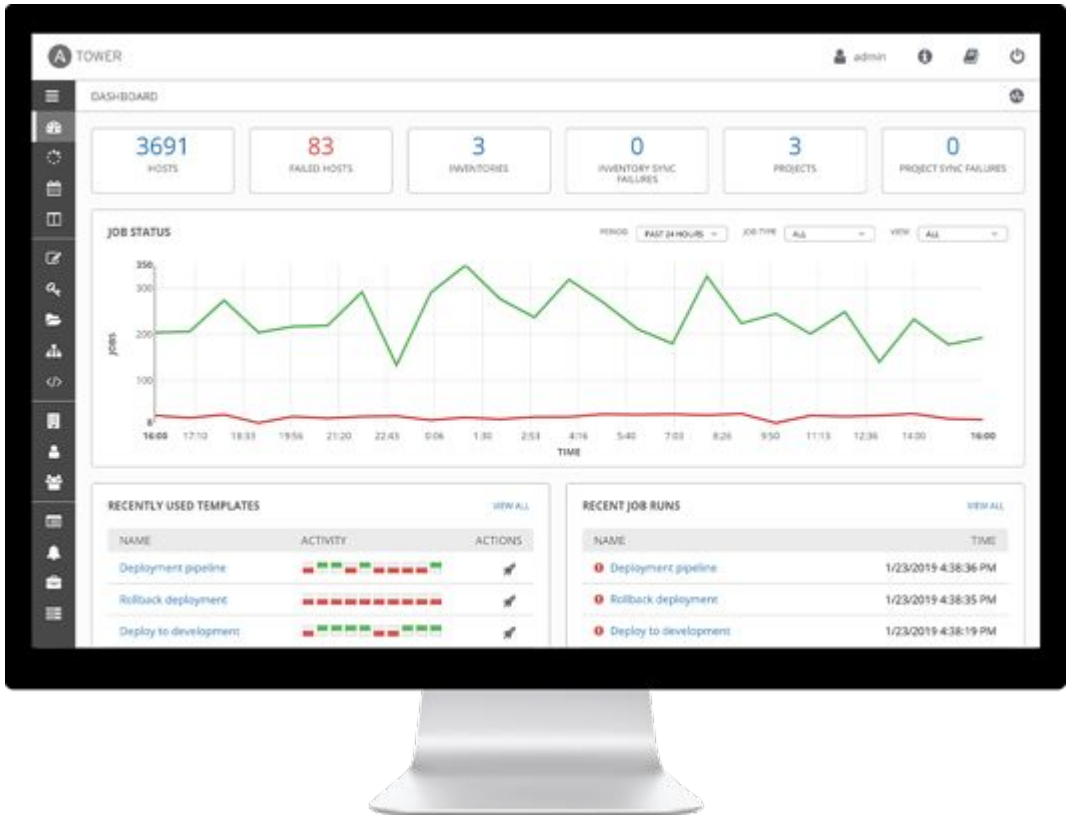
Built into the Ansible CLI and Tower

galaxy.ansible.com

Red Hat

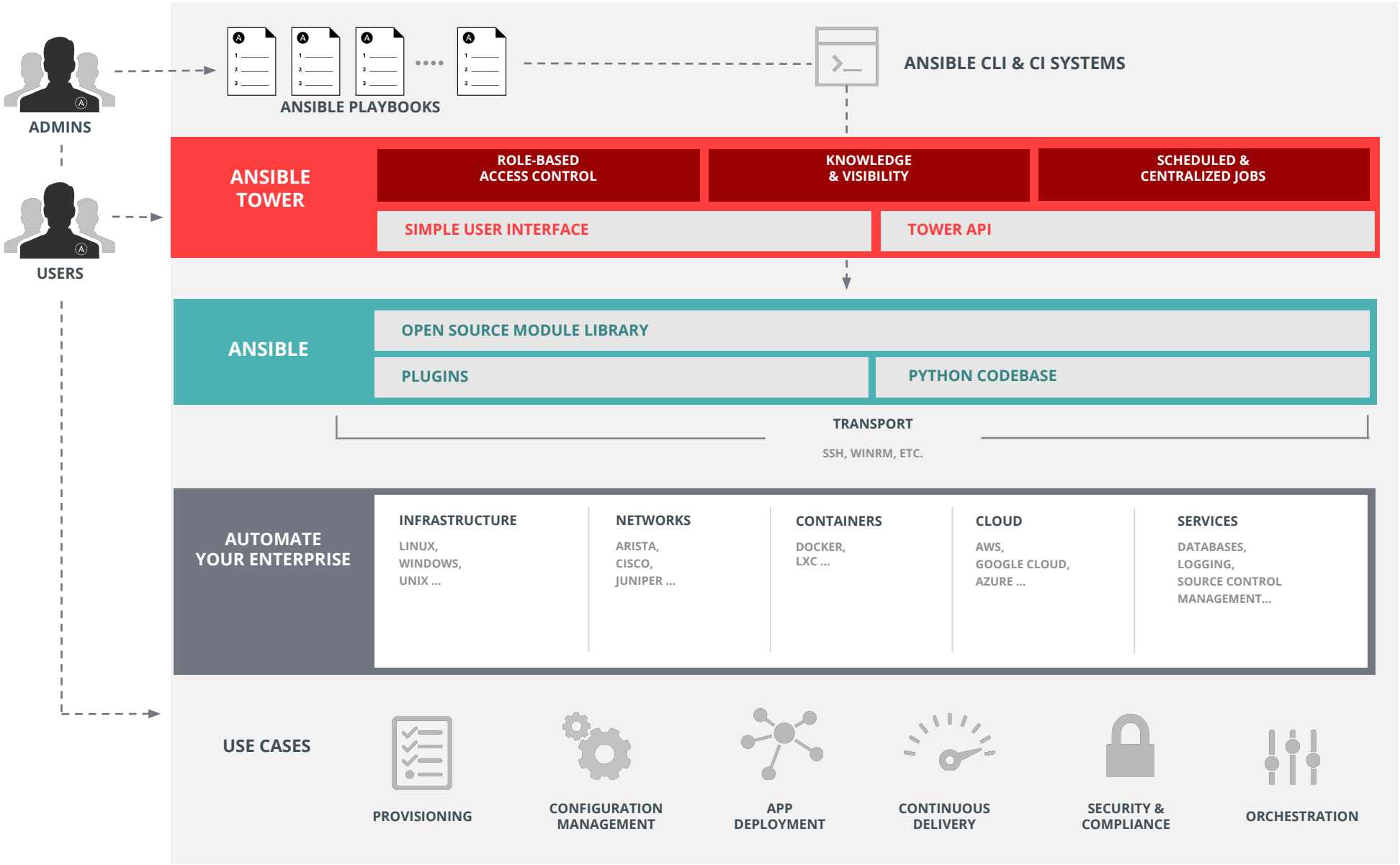
Ansible Tower

- A visual dashboard
- Role-based access control
- Job scheduling
- Graphical inventory management
- Multiplaybook workflows
- External logging integrations
- Real-time job status updates
- Support for external credential vaults
- REST API and CLI
- Self Service I.T.
- .....



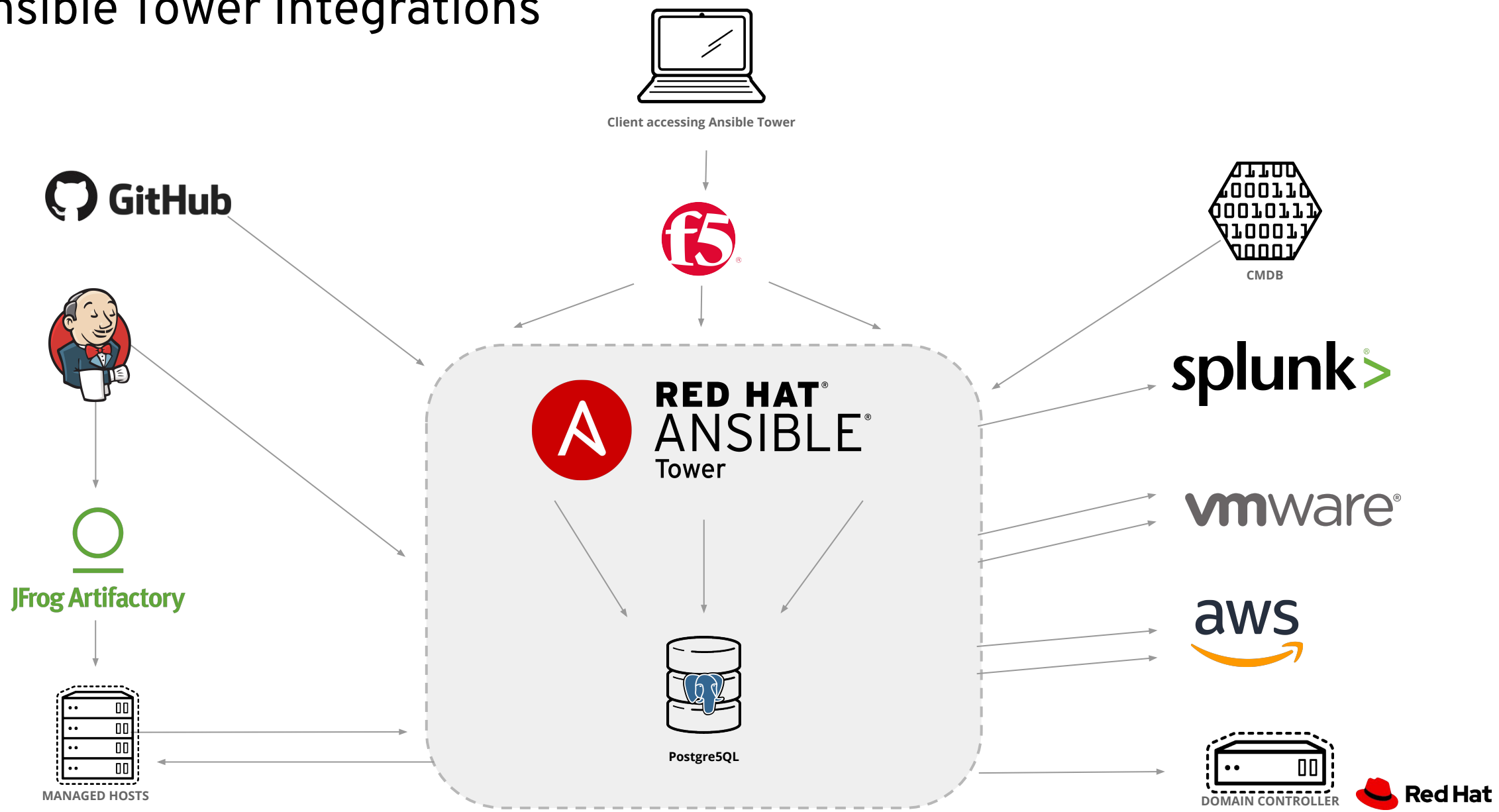
Red Hat

Ansible Tower

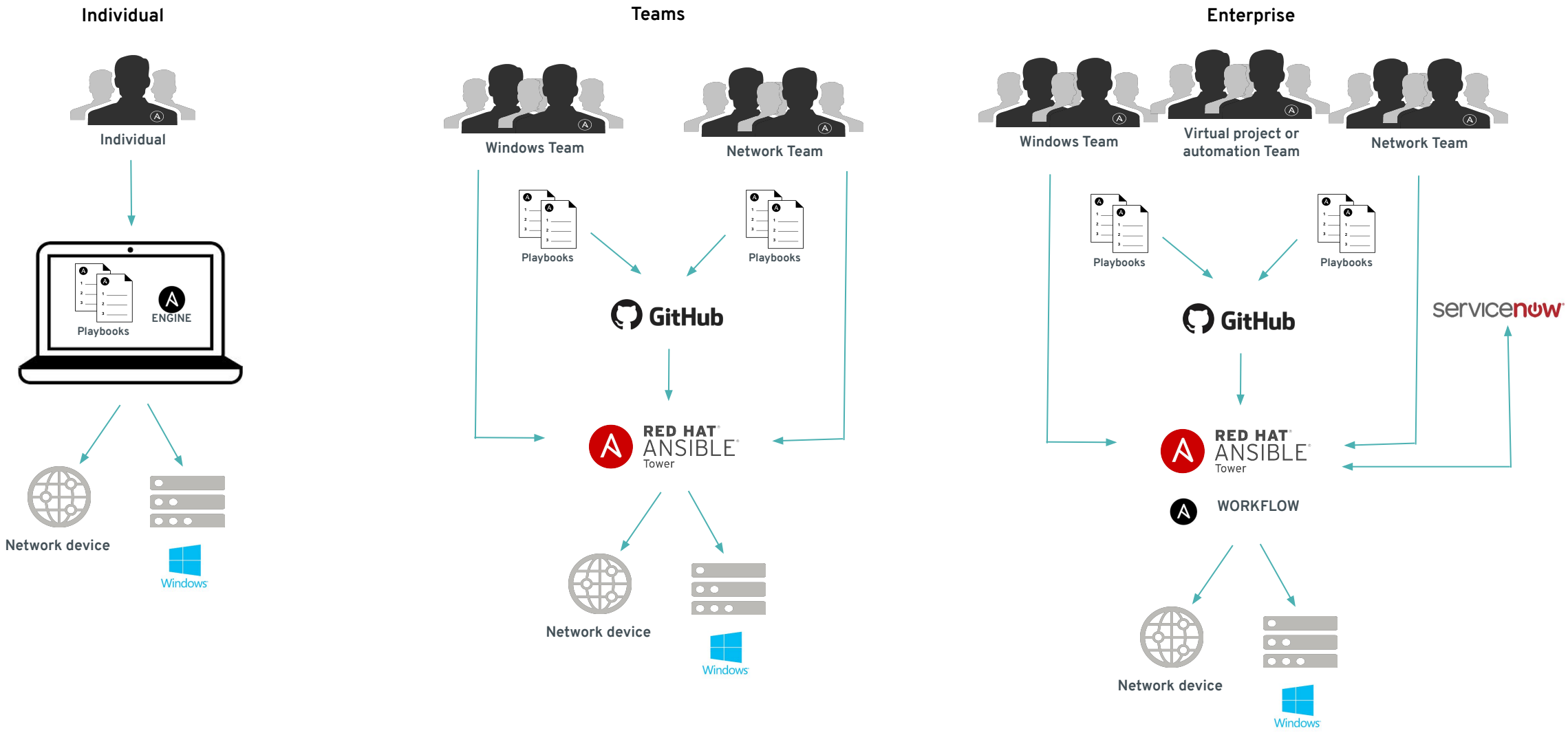


Red Hat

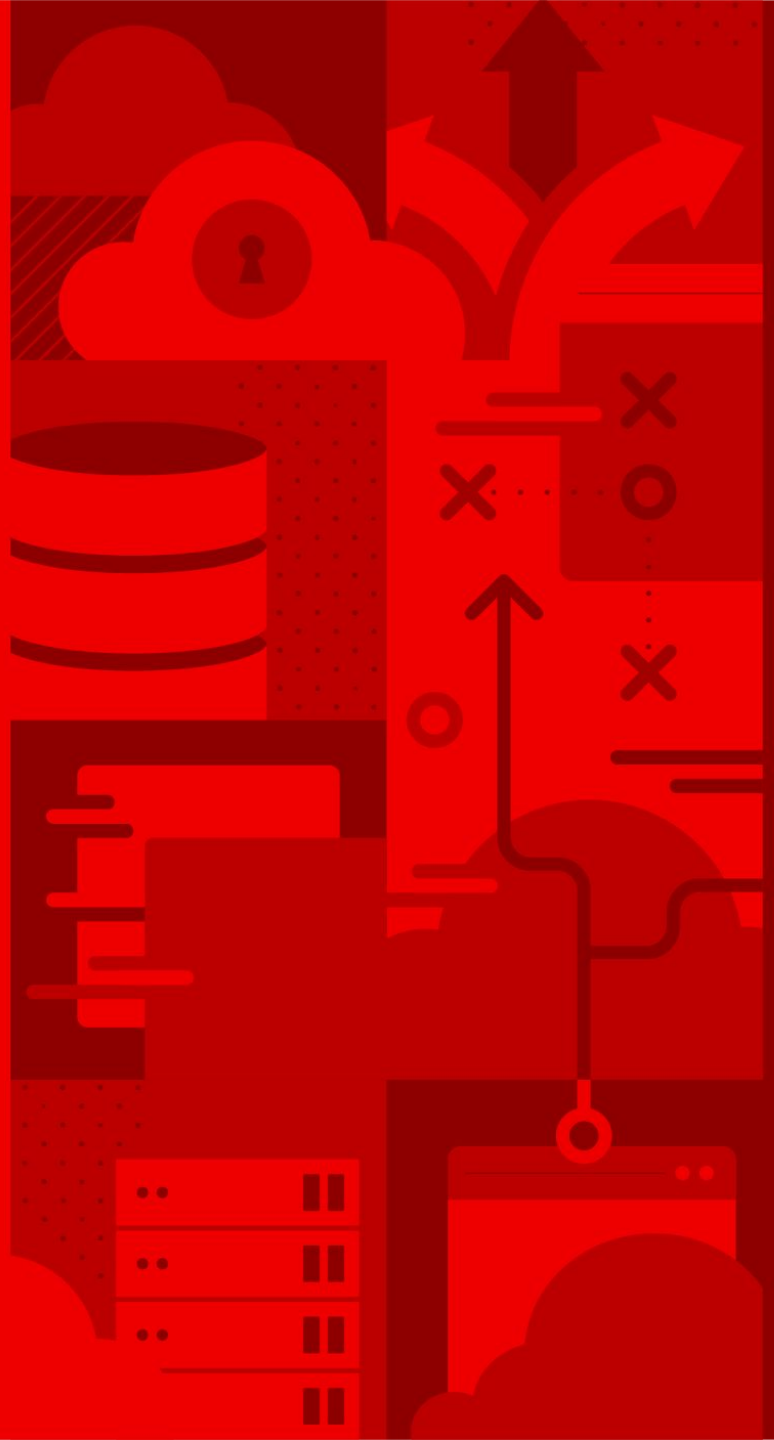
Ansible Tower Integrations



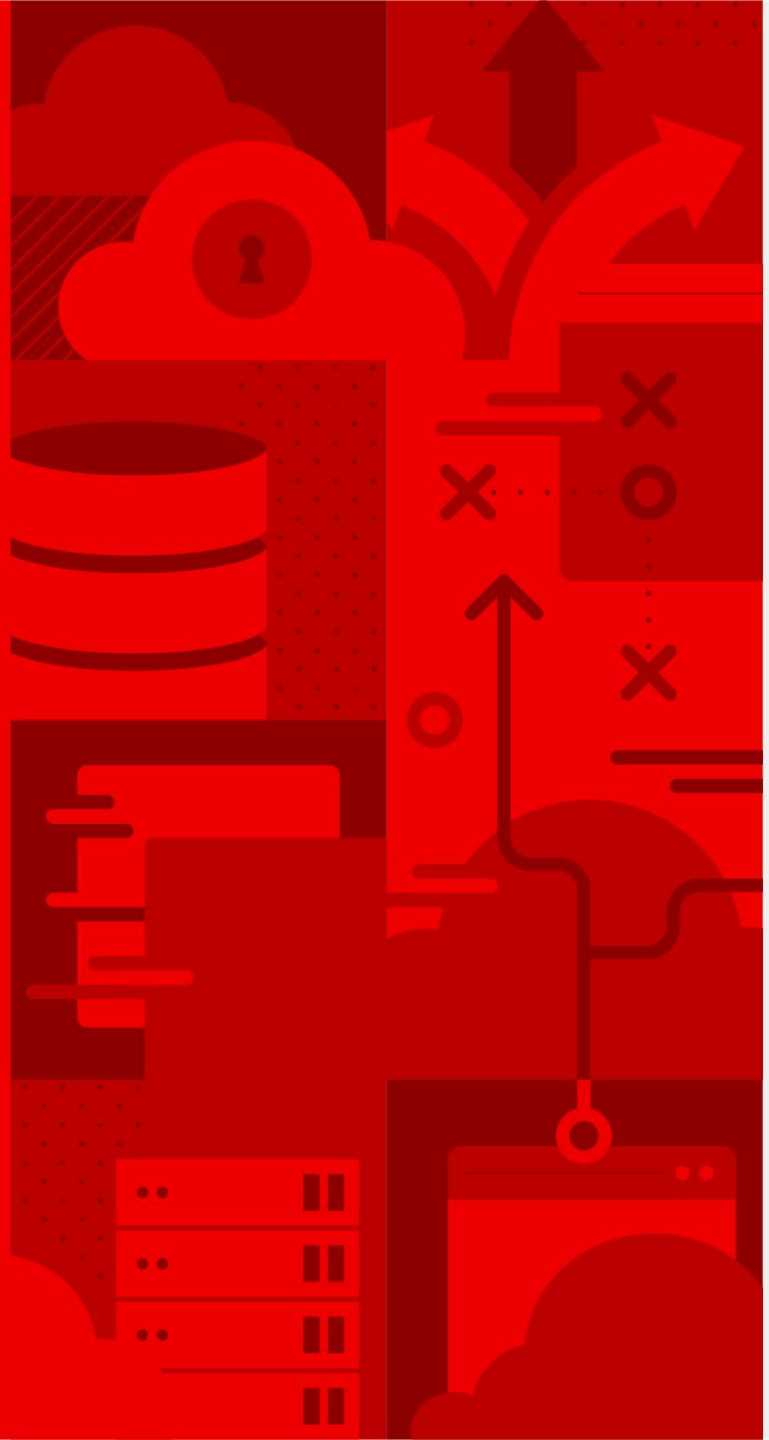
Ansible in the Enterprise



Red Hat



- Try Ansible yourself: ansible.com/resources/get-started
- Check out the documentation: docs.ansible.com
- Ansible Galaxy: galaxy.ansible.com
- Try Ansible Tower: ansible.com/products/tower/trial



Thank you

Red Hat is the world's leading provider of enterprise open source software solutions. Award-winning support, training, and consulting services make Red Hat a trusted adviser to the Fortune 500.

linkedin.com/company/red-hat

facebook.com/redhatinc

youtube.com/user/RedHatVideos

twitter.com/RedHat

Red Hat