Red Hat Satellite

Provisioning in an Enterprise Setting

You can do with this, or you can do with that

Josh Swanson
Platform Specialist Solution Architect

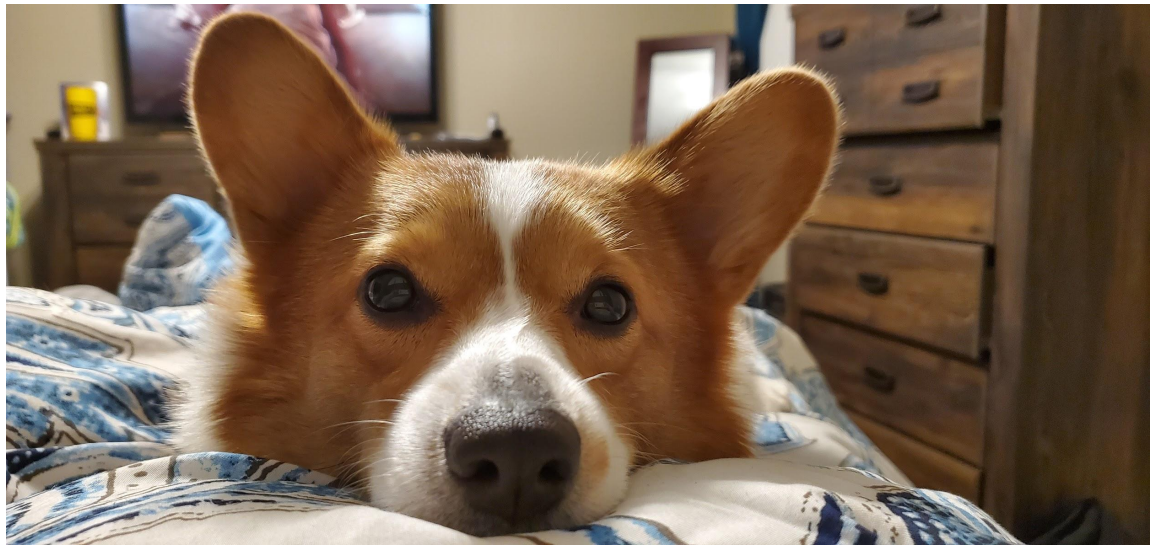
Pete Scurek
Solution Architect

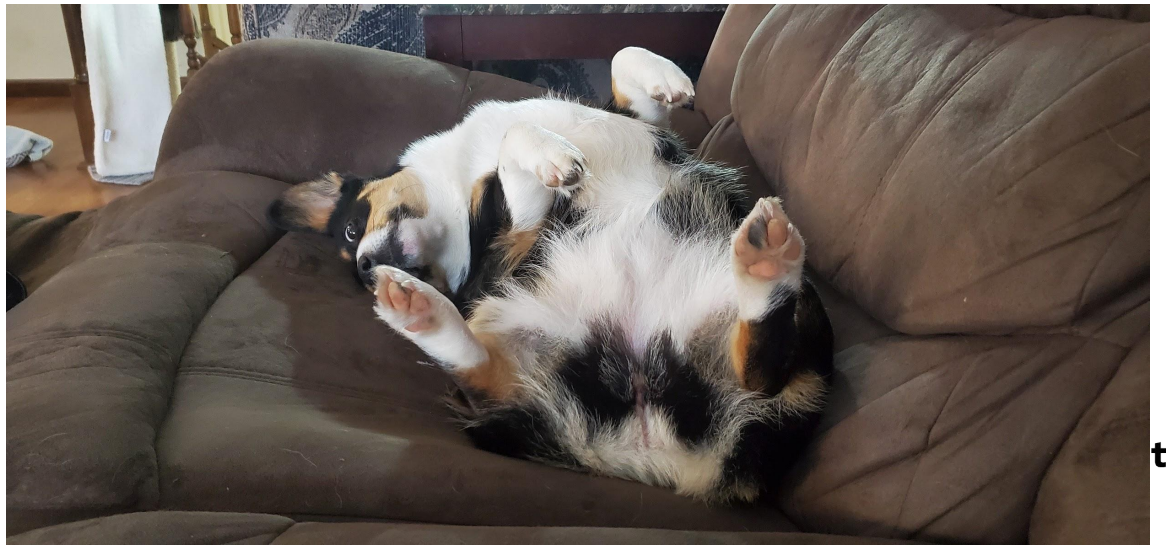
Red Hat

Intros
Agenda
Housekeeping

Josh Swanson

joshswanson@redhat.com





Pete Scurek

pscurek@redhat.com

Agenda

Let’s Break Down the Provisioning Process

Level-set:
- What is Red Hat Satellite?
- What is Red Hat Ansible Automation Platform?
- What is a standard operating environment?
- What is the provisioning process?

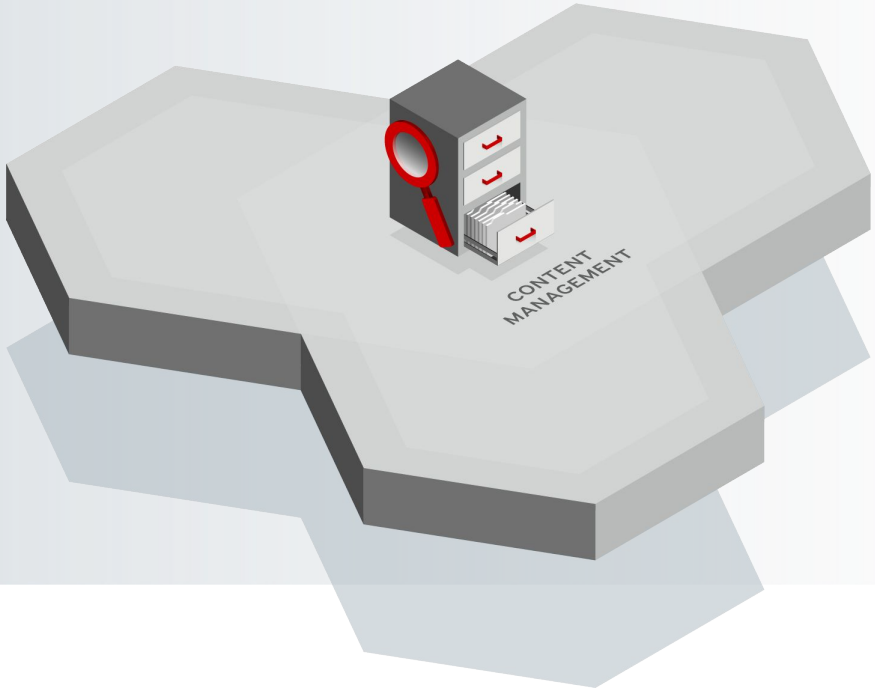
Mapping provisioning steps to Red Hat technologies:
- Pre-steps - What needs to be done so RHEL can be provisioned?
- Creating an instance of RHEL - options, options, options
- Post-steps - How do we get RHEL to conform to our SOE?

Tying it all together:
- What does an enterprise provisioning process look like?
- Synergies between Red Hat Satellite and Red Hat Ansible Automation Platform
- Integrating with existing processes
- Using what makes sense

What is Red Hat Satellite?

Red Hat Satellite is a scalable platform to manage patching, provisioning, and subscription management of your Red Hat infrastructure, regardless of where it is running.

Content Management

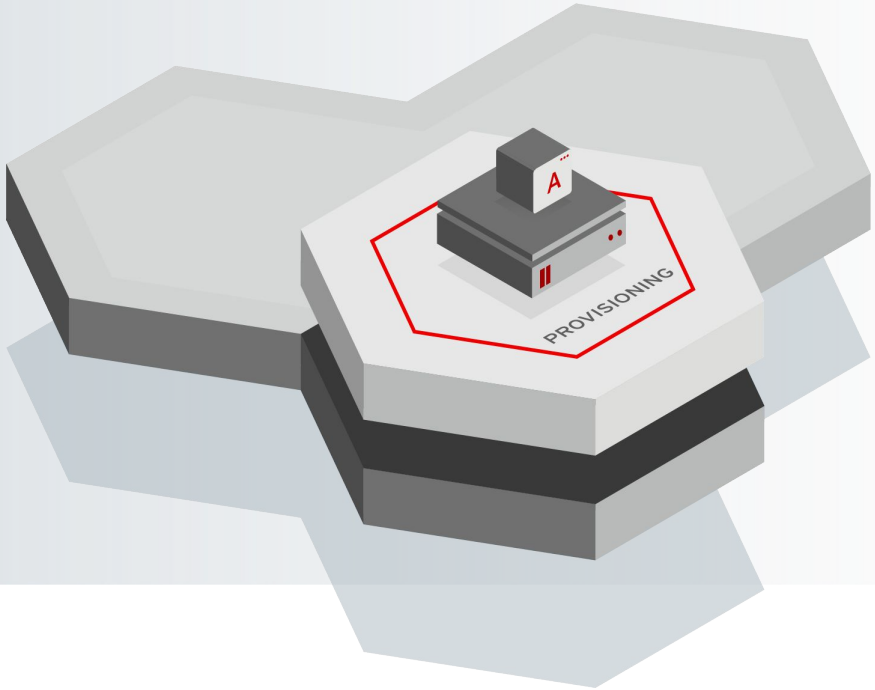


Content Repository any type of content made available to any host

Curation of content prior to distribution

Distribution of content as close as possible to the end point.

Provisioning Management



Provision to bare metal, virtual, private, and public clouds

Import non-provisioned hosts

Automate using Ansible roles to perform post-provisioning steps

What is the Red Hat Ansible Automation Platform?

Red Hat

Red Hat Ansible Automation Platform

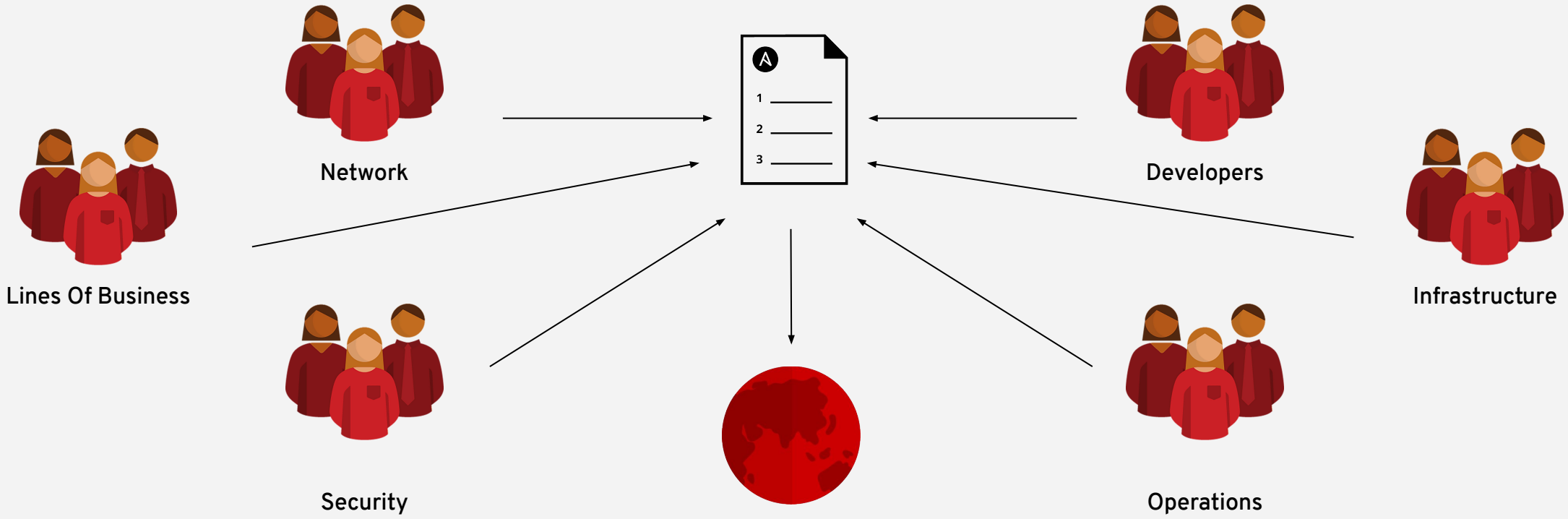
Network | Lines of business | Security |  | Operations | Infrastructure | Developers

Engage	**Ansible Hosted Services:** Engage users with an automation focused experience
Scale	**Ansible Tower:** Operate & control at scale
Create	**Ansible Engine:** Universal language of automation

Fueled by an open source community

Red Hat

When automation crosses teams, you need an automation platform



Why Ansible?

Simple

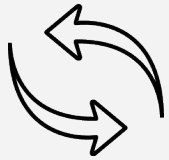
Human readable automation

No special coding skills needed

Tasks executed in order

Usable by every team

Get productive quickly



Powerful

App deployment

Configuration management

Workflow orchestration

Network automation

Orchestrate the app lifecycle

Agentless

Agentless architecture

Uses OpenSSH & WinRM

No agents to exploit or update

Get started immediately

More efficient & more secure

Red Hat

What is a standard operating environment?

An SOE is a standard operating environment, or a specific computer operating system and collection of software that an IT department defines as a standard build.

What is Provisioning?

The provisioning process is the act of creating a new instance of Red Hat Enterprise Linux to a specified compute location that conforms to a standard operating environment.

Pre-steps: Preparing to Provision RHEL

Pre-Steps: Preparing to Provision RHEL

What needs to be done to ensure we can bring up RHEL?

Red Hat Technology:	Red Hat Satellite	Red Hat Ansible Automation Platform
Organization Pre-Reqs:		
Request entered into request system		✓
Reviews and approvals		✓
Entry of data into CMDB/SOR		✓
Transition Request to technical system		✓
Technical Pre-Reqs:		
Request validation		✓
Capacity checks		✓
Obtain IP address	✓*	✓
Create DNS entry	✓*	✓

Creating an Instance of RHEL

Red Hat

Bringing Up an Instance of RHEL

Comparing the two main provisioning techniques

Provisioning Process Component:	Red Hat Satellite	Red Hat Ansible Automation Platform
Provisioning from an image	✓	✓
Provisioning from bootdisk	✓	
Full customization of kickstart	✓	
End to end lifecycle management	✓	✓
Consumable via API	✓	✓
Pre-built provisioning templates	✓	
Templating engine	✓	✓
Variable precedence	✓	✓

Post-steps: Preparing to Provision RHEL

Red Hat

Post-Steps: Aligning to a SOE

Comparing the two main provisioning techniques

Post-Provisioning Component:	Red Hat Technology: Red Hat Satellite	Red Hat Ansible Automation Platform
Run automation to enforce SOE	✓	✓
Build dynamic inventories	✓	✓
Automation workflows with success/failure branches		✓
Coordination with external systems/entities (firewalls, load balancers)		✓
Natively include external automation		✓
Gather information from multiple data sources	✓	✓
Act as a full featured content source	✓	
Full featured RBAC capabilities	✓	✓

What does an enterprise provisioning process look like?

Enterprise Provisioning Process

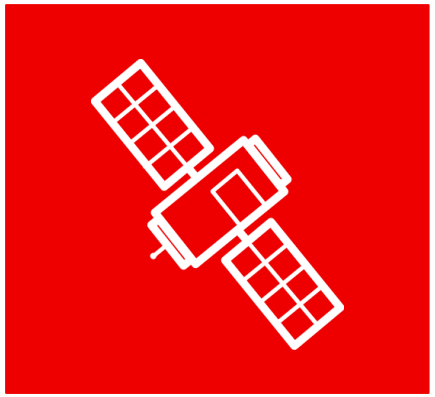
Leverage the Strengths of the Technologies

Use the Ansible Automation Platform for your orchestration:

- Tie together your various IT/management systems
- Leverage a common automation language across teams
- Success/Failure path orchestration
- Role based access control

Treat Satellite as a RHEL Management Platform

- Full content control
- Lifecycle management when appropriate
- Consumable via full RESTful API

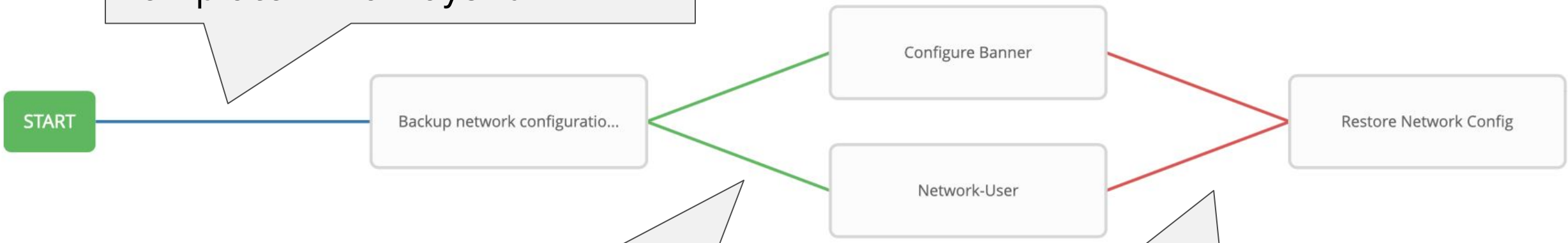


Red Hat

Enterprise Provisioning Process

Workflows allow for true a end-to-end provisioning process

Blue indicates this Job Template will always run



Green indicates this Job Template will only be run if the previous Job Template is successful

Red indicates this Job Template will only be run if the previous Job Template fails

Automation adoption

Empower business transformation with highly scalable automation, orchestration of capabilities, and evolved ways of working in a culture of collaboration.

Define your journey

Strategy

Chart a journey map from foundational use case to target state with measurable business outcomes.

Foundation

Empower a cross-functional team to automate and operate an initial set of workflows in an initial automation framework.

Adoption

Expand skills, integrations, and orchestrated workflows in increments with measurable value.

Technology

Establish automation framework, tooling, and techniques that empower process evolution and business-driven workflows.

Process

Adopt open practices to quickly develop, validate, and launch new services and workflows in response to changing demands.

Culture

Spark innovation and agility with new approaches to increase collaboration and communities that empower and inspire the organization

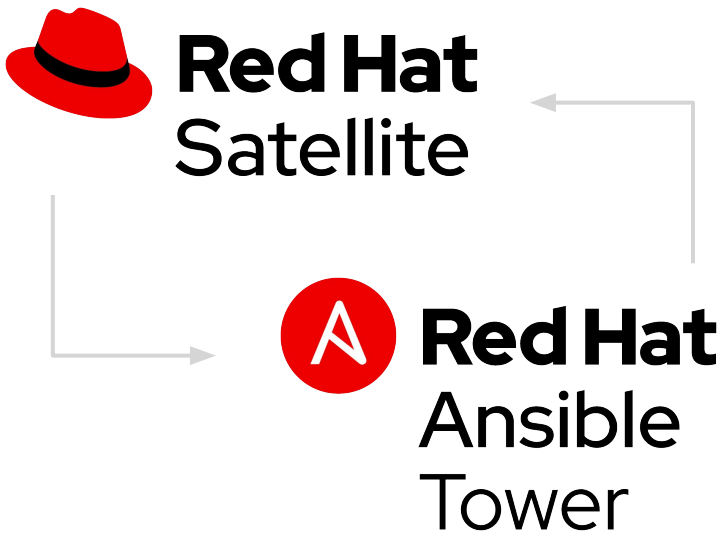
Red Hat Satellite

Satellite and Ansible: Better Together

Red Hat

Satellite and Ansible Tower integration

Documented best practices to help optimize use of both products



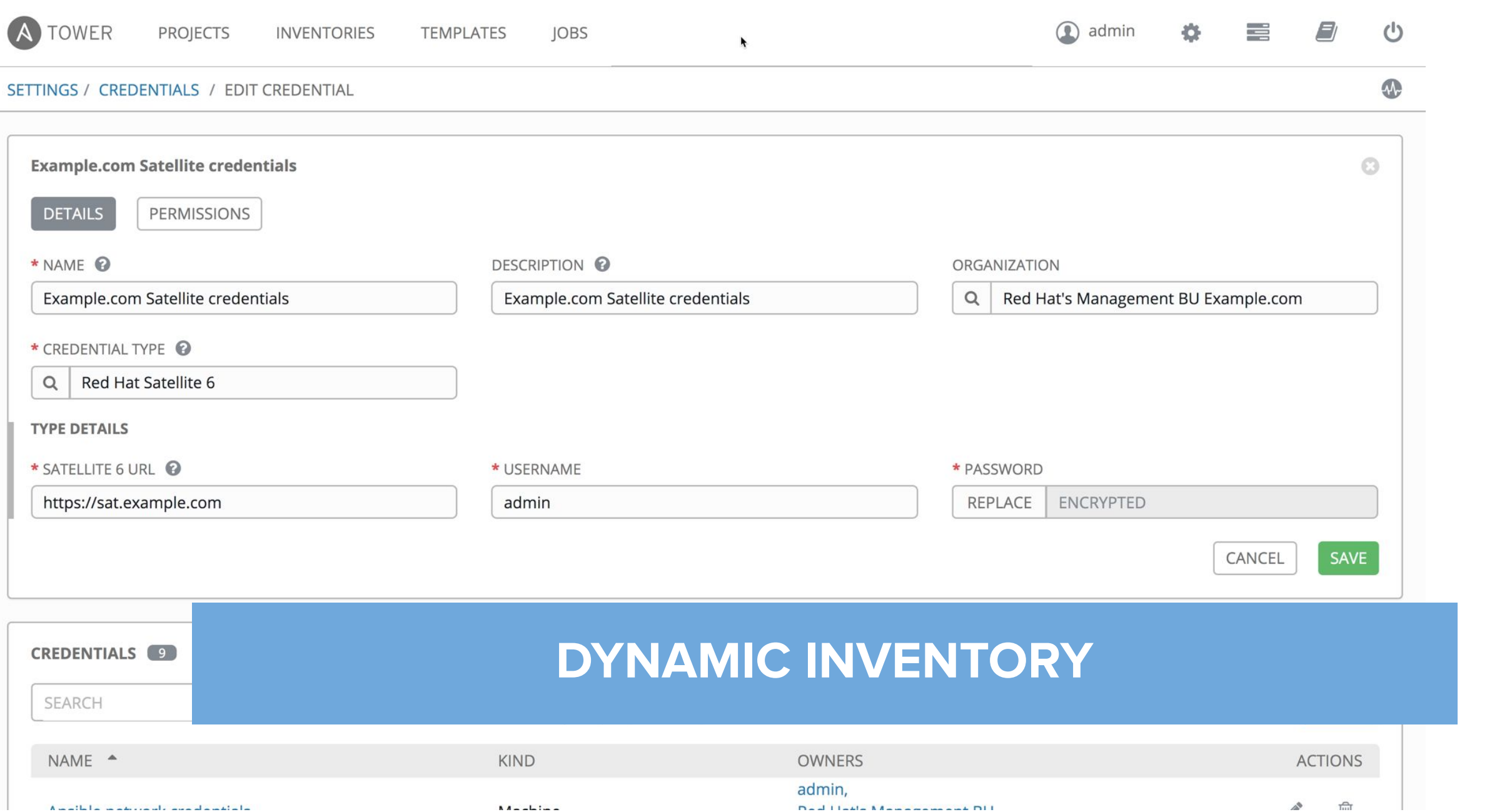
By integrating Red Hat Satellite with Red Hat Ansible® Tower, administrators can now perform the following functions:

Dynamic inventory

Allows Ansible Tower to use Satellite as a dynamic inventory source

Provisioning callbacks

Allows systems provisioned via Satellite to “callback” to Ansible Tower so that playbook runs can happen post-provisioning



DYNAMIC INVENTORY

Red Hat

PROVISIONING CALLBACKS

A definition straight from the Tower documentation

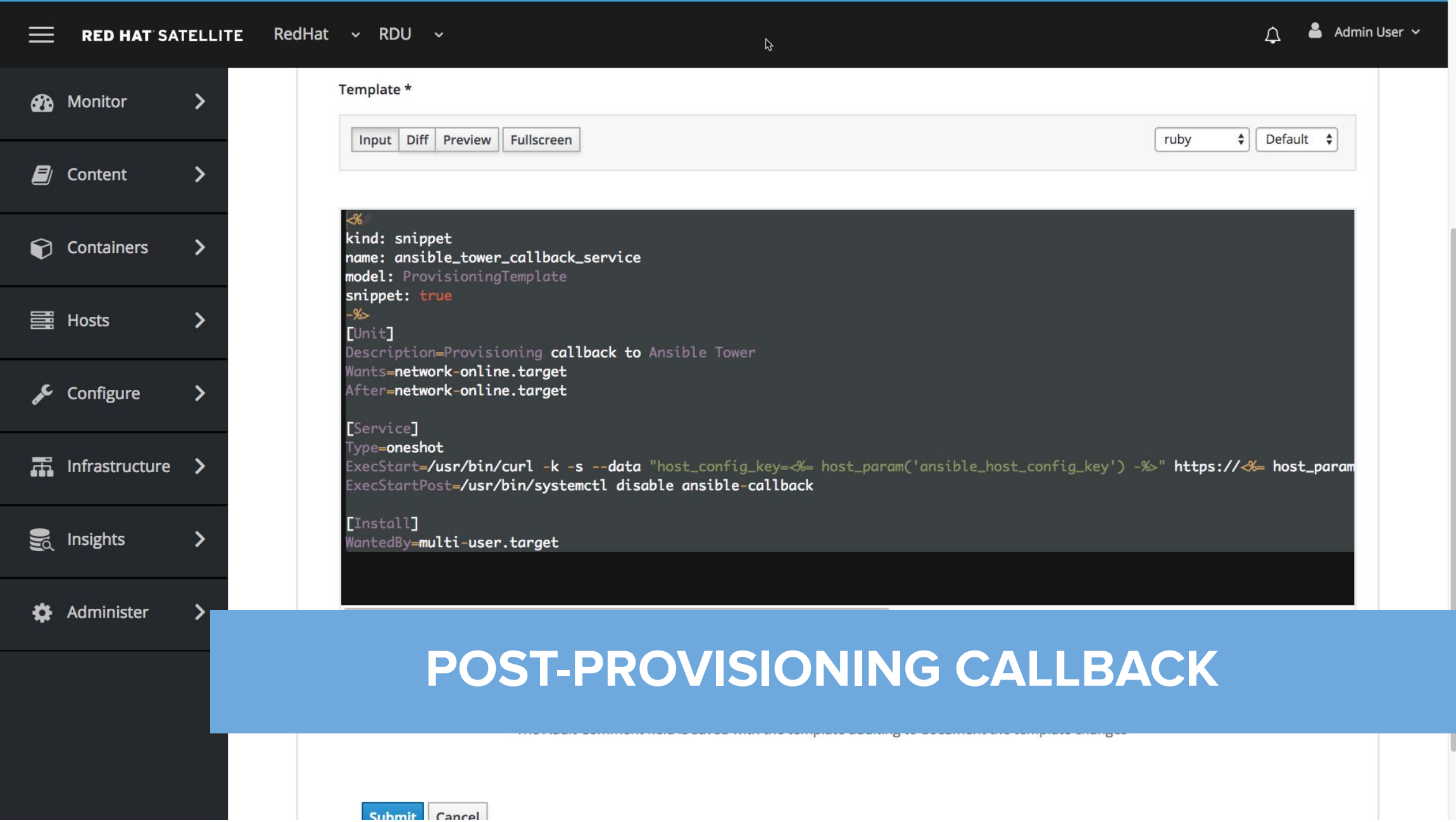
Provisioning callbacks are a feature of Tower that allow a host to initiate a playbook run against itself, rather than waiting for a user to launch a job to manage the host from the tower console.

Red Hat

```
<%#
kind: snippet
name: ansible_provisioning_callback
model: ProvisioningTemplate
snippet: true
-%>
<% if host_param_true?('ansible_tower_provisioning') -%>
<%
  rhel_compatible = @host.operatingsystem.family == 'Redhat' && @host.operatingsystem.name != 'Fedora'
  os_major = @host.operatingsystem.major.to_i
  has_systemd = (@host.operatingsystem.name == 'Fedora' && os_major >= 20) || (rhel_compatible && os_major >= 7)
-%>
<% if has_systemd -%>
<%= save_to_file('/etc/systemd/system/ansible-callback.service',
                 snippet('ansible_tower_callback_service')) %>
# Runs during first boot, removes itself
systemctl enable ansible-callback
<% else -%>
# Assume systemd is not available
<%= save_to_file('/root/ansible_provisioning_call.sh', snippet('ansible_tower_callback_script')) %>
(crontab -u root -l 2>/dev/null; echo "@reboot /root/ansible_provisioning_call.sh" ) | crontab -u root -
<% end -%>
```

POST-PROVISIONING CALLBACK

Red Hat



POST-PROVISIONING CALLBACK

Red Hat

What are the Foreman Ansible Modules?

https://theforeman.org/2019/09/automating-foreman-and-katello-with-ansible.html

Foreman Ansible Modules (FAM) are a set of Ansible modules to manage Foreman ;-)

These modules are an evolution from the foreman and katello modules currently present in Ansible itself, as those are deprecated since Ansible 2.8 and are scheduled for removal in 2.12. Due to the use of a Katello (or rather Satellite) specific library, the old modules would not work properly in plain Foreman setups and often lacked features that were not yet present in Red Hat Satellite 6.

Over the course of the past year, the community sat together, cleaned the modules up, created tests and documentation and finally also ported the modules to a Satellite independent library.

Red Hat

How do the Modules Actually Work?

https://theforeman.org/2019/09/automating-foreman-and-katello-with-ansible.html

MAGIC! Well, actually, no, not magic, DOCUMENTATION!

Foreman has a powerful API with rich API documentation. This documentation is generated by the apipie-rails gem, which also provides a machine readable version of said documentation. You've probably seen long-ish rake apipie:cache processes when installing Foreman and plugins – that's the gem re-generating the documentation to match the set of plugins available in your environment.

The modules use a library (apypie) that can parse the machine readable documentation of your instance and generate correct API requests based on that documentation.

Given almost all modules share a lot of common code, there is an abstraction class ForemanAnsibleModule which takes care of the common tasks like establishing an API connection, executing searches and creating/updating/deleting entities. This allows the modules be clean and only contain data/code relevant for their specific task – have a look at the foreman_organization module for a very simple example.

Red Hat

How can the modules be obtained?

Shipped as a collection

- Github
  - git clone https://github.com/theforeman/foreman-ansible-modules.git
- Automation Hub
  - ansible-galaxy collection install redhat.satellite
- Ansible Galaxy
  - ansible-galaxy collection install theforeman.foreman

```
[jswanson@rocinante configure_satellite]$ ansible-galaxy collection install
theforeman.foreman
Process install dependency map
Starting collection install process
Skipping 'theforeman.foreman' as it is already installed
```


Red Hat

How Can the Modules be Used?

https://theforeman.org/2019/09/automating-foreman-and-katello-with-ansible.html

The foreman-ansible-modules git repository contains instructions how the modules can be installed in your environment and module documentation is available from theforeman.org.

Usually you'll find one module per Foreman entity (Organization, Location, Host Group etc.) or action (Katello Repository Sync, Katello Content Upload, etc).

```
[jswanson@rocinante tasks]$ ls -l
/home/jswanson/.ansible/collections/ansible_collections/theforeman/foreman/plugins/modules
total 384
-rw-r--r--. 1 jswanson jswanson 15243 Jul 21 21:20 activation_key.py
-rw-r--r--. 1 jswanson jswanson  2715 Jul 21 21:20 architecture.py
-rw-r--r--. 1 jswanson jswanson  5955 Jul 21 21:20 auth_source_ldap.py
-rw-r--r--. 1 jswanson jswanson  3389 Jul 21 21:20 bookmark.py
-rw-r--r--. 1 jswanson jswanson  3202 Jul 21 21:20 compute_attribute.py
-rw-r--r--. 1 jswanson jswanson  5641 Jul 21 21:20 compute_profile.py
-rw-r--r--. 1 jswanson jswanson 11338 Jul 21 21:20 compute_resource.py
```


Red Hat

Demo: A provisioning process with AAP and Satellite

Integrating With Existing Processes

Integrating With Existing Processes/Automation

Don’t Reinvent the Wheel

Allow Ansible to orchestrate existing processes

- Instead of bouncing tickets or emails from team to team, consider putting together a workflow in tower
- Let Ansible coordinate between different IT systems
- One central place to view and manage provisioning workflow
- Democratize automation across automation domains

Manage existing automation

- Ansible has script modules, powershell modules
- Anything that has an API, Ansible will be best friends with
- Gain all of the benefits of Ansible (RBAC, tempating, etc) without having to rewrite everything in Ansible
- Add new features and functionality rapidly with Ansible modules

Integrating With Existing Processes/Automation

Don’t Reinvent the Wheel

```
- name: run existing powerCLI provisioning script
  ansible.windows.win_shell: C:\provision-vm.ps1 >> C:\provisioning-log.txt
  args:
    chdir: C:\temp
  delegate_to: win_host

- name: run existing python provisioning script
  script: /usr/local/bin/provision-host.py
  args:
    executable: python3
  delegate_to: localhost
```


Red Hat

Use What Makes Sense

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

youtube.com/user/RedHatVideos

facebook.com/redhatinc

twitter.com/RedHat

Red Hat