



Introduction to automation controller

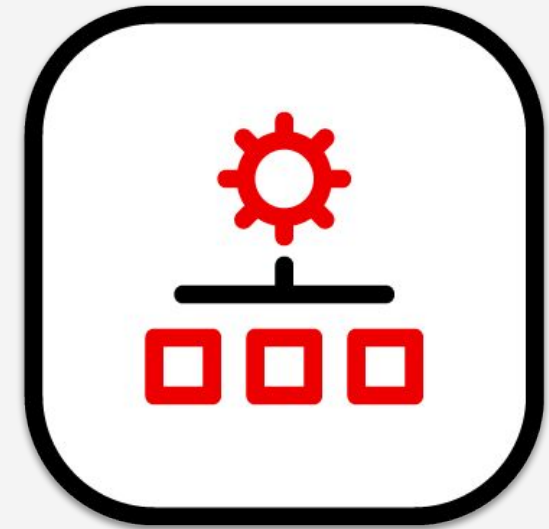
WK1243

Leonardo Gallego
Senior TMM
Ansible

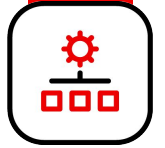
Anshul Behl
Principal TMM
Ansible

Craig Brandt
Principal TMM
Ansible

Automation controller
labs.demoredhat.com



Automation controller. Define, operate, and delegate.

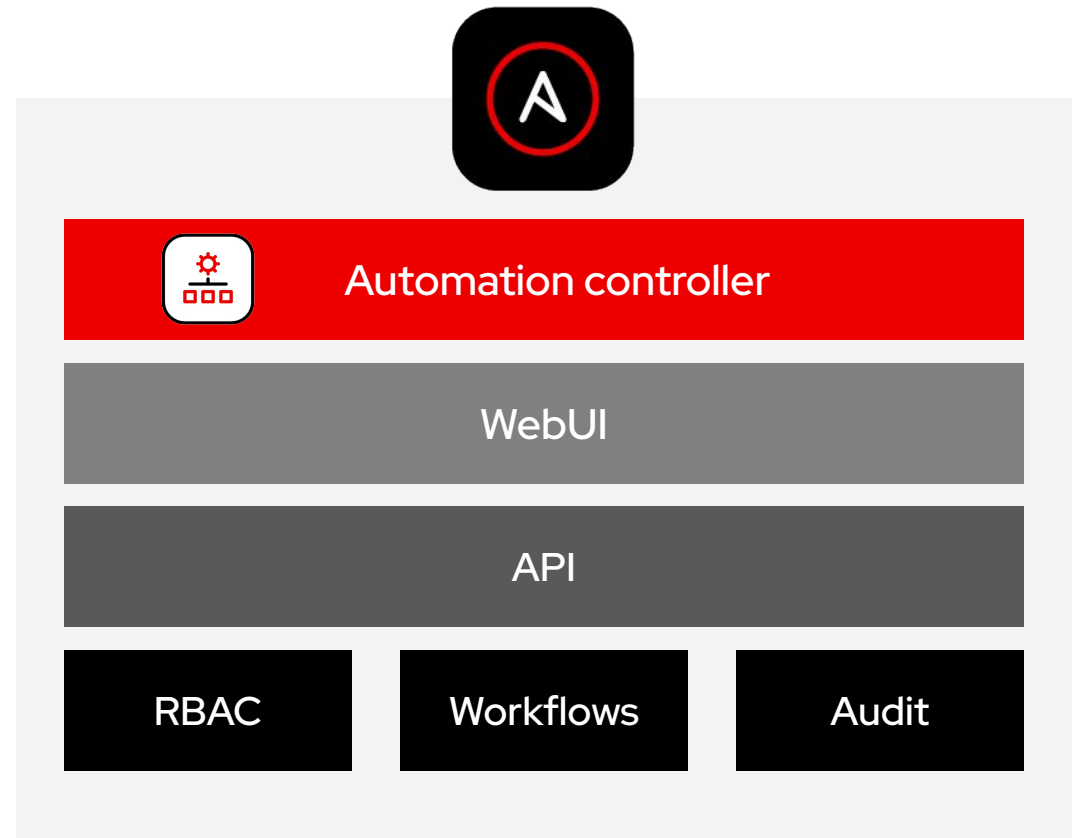


What is it?

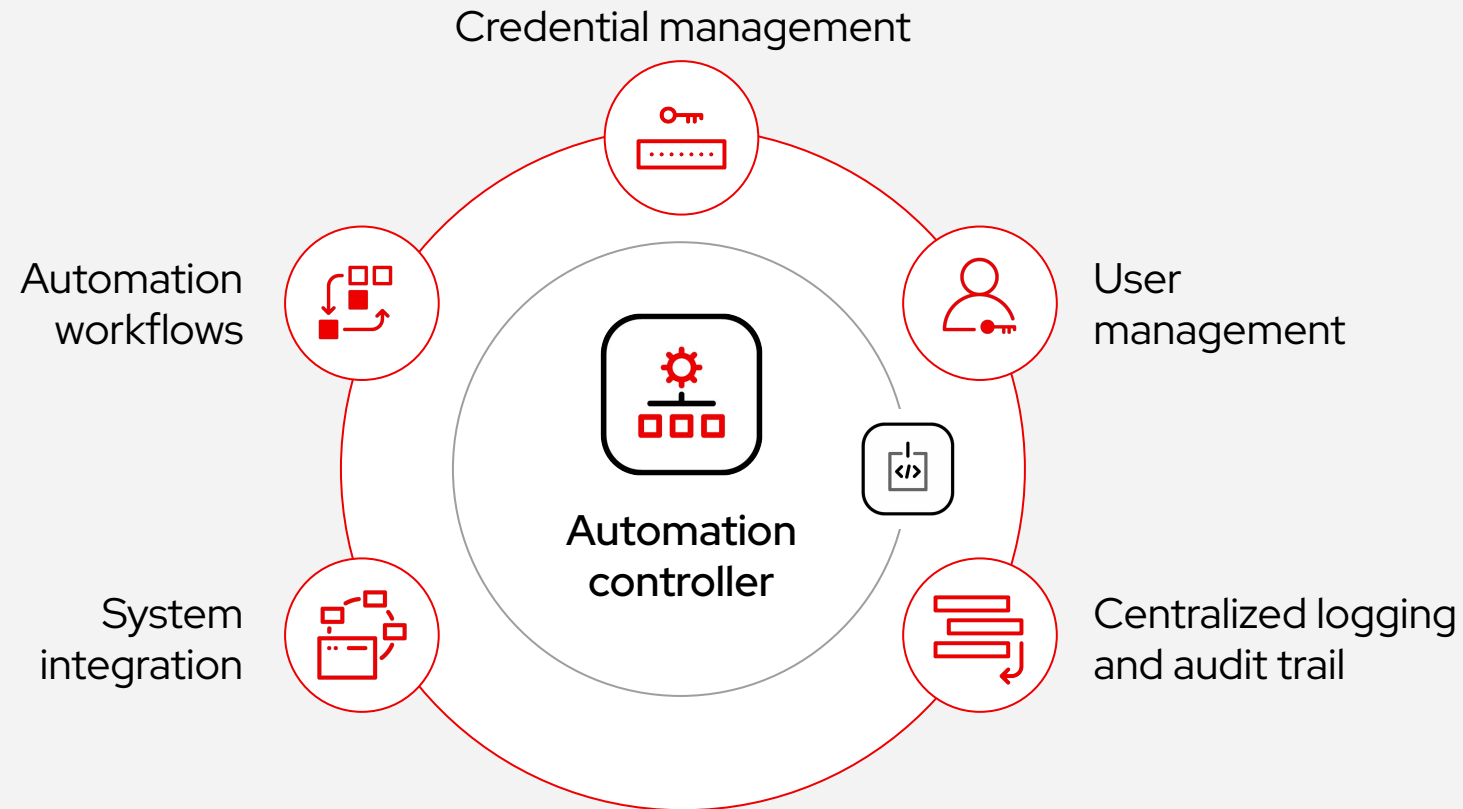
Automation controller is the Ansible Automation Platform control plane which enables users to define, operate, and delegate automation across their enterprise

Automation controller provides:

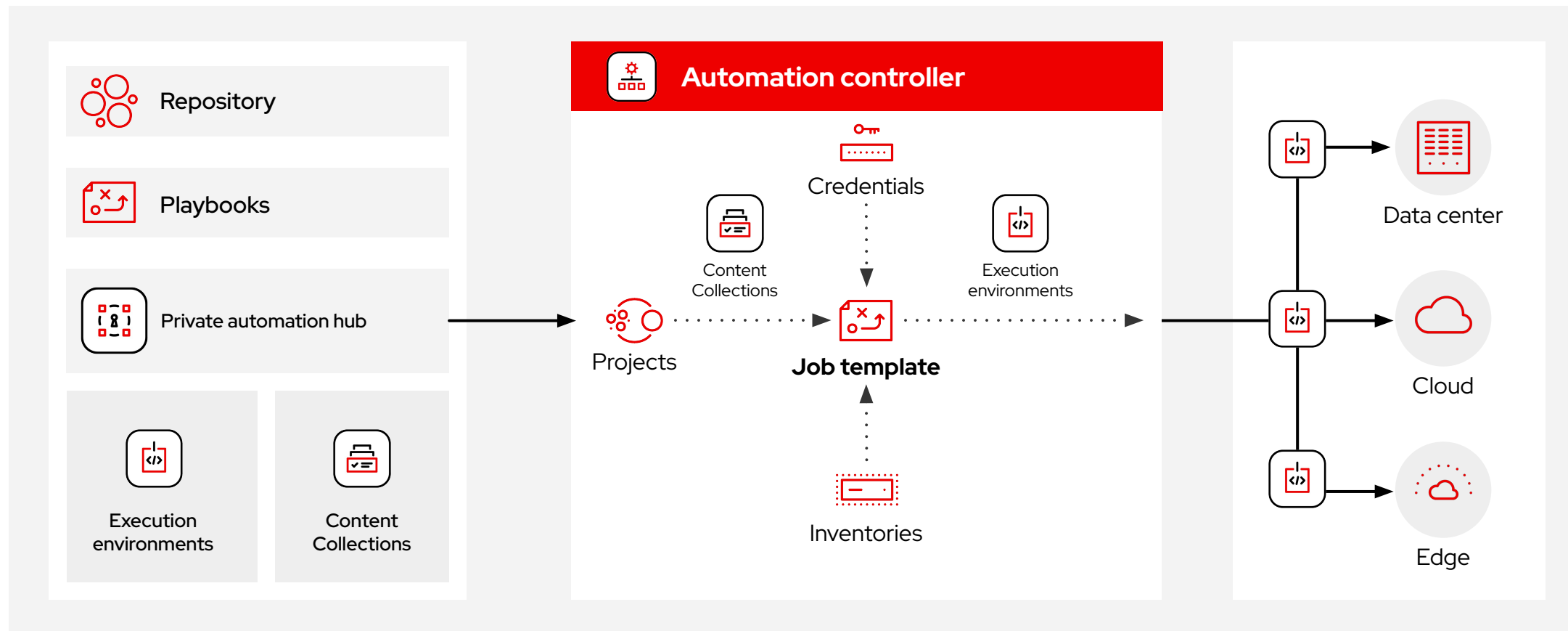
- ▶ WebUI and API
- ▶ Role-based access control
- ▶ Powerful workflows
- ▶ Centralized logging
- ▶ Credential management
- ▶ Push-button automation

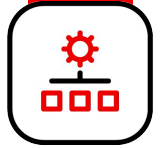


Enterprises **need more for their automation.**



Defining how your content runs in **automation controller**.



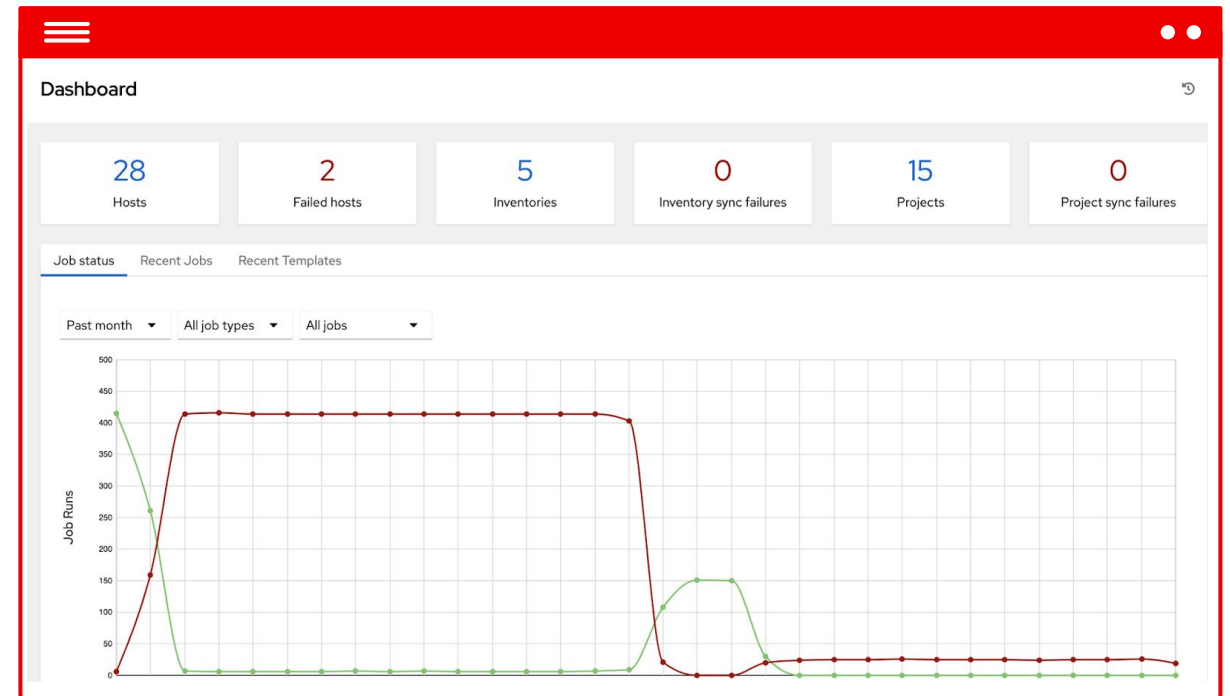


Automation controller dashboard.

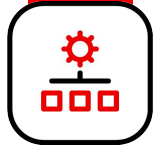
View your automation status.

What is it?

- ▶ Overview of automation controller, the jobs happening, the nodes connected and what worked and failed
- ▶ Provides the ability to quickly spot irregularities and drill into more details



Inventories. What do I want to run my automation on?



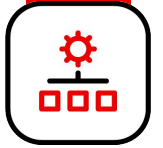
What is it?

Collection of endpoints against which jobs may be launched

- ▶ Multiple inventory sources supported
- ▶ Dynamic endpoint discovery
- ▶ Logically group endpoints by metadata or user-defined filters
- ▶ Granular RBAC permissions

Name	Status	Type	Organization	Actions
<input type="checkbox"/> AWS Inventory	Disabled	Inventory	ACME Corp	
<input type="checkbox"/> Azure Inventory	Disabled	Inventory	ACME Corp	
<input type="checkbox"/> Data center	Disabled	Inventory	ACME Corp	
<input type="checkbox"/> Remote Office A	Disabled	Inventory	ACME Corp	

Inventories. What do I want to run my automation on?



Inventories

Create new inventory

Name * Description Organization *

Instance Groups

Labels ⓘ

Options

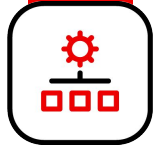
☐ Prevent Instance Group Fallback ⓘ

Variables ⓘ YAML JSON

1 ---

Save Cancel

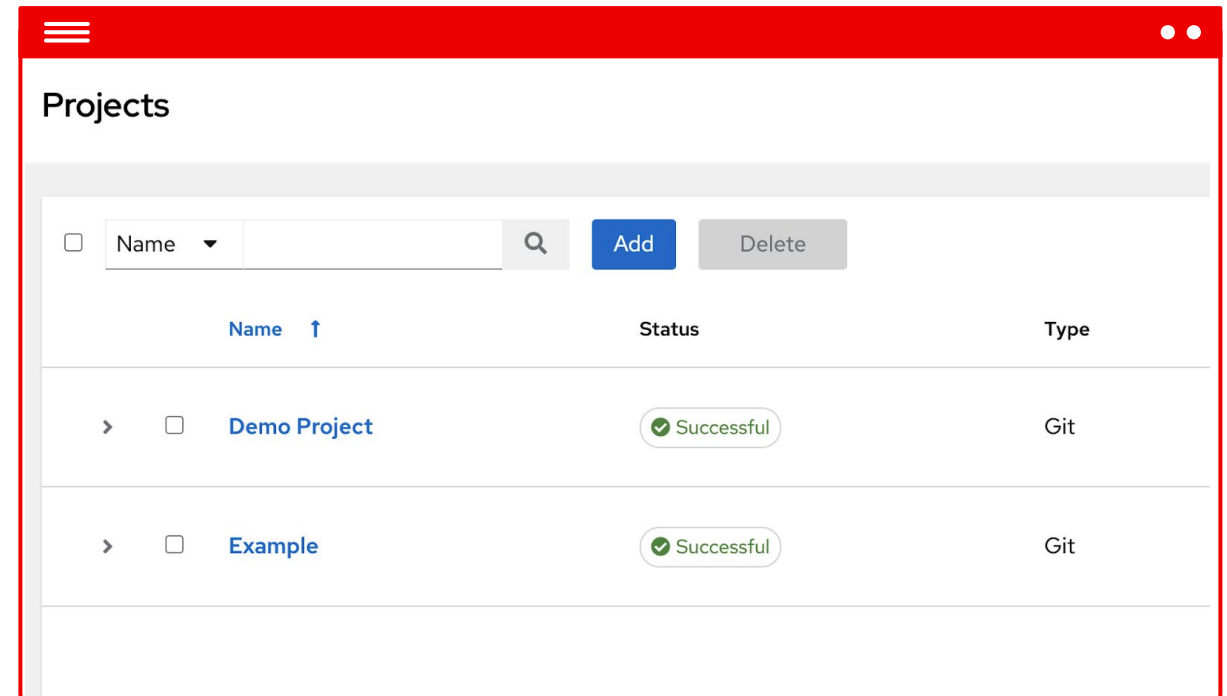
Projects. Adding your automation content to controller.



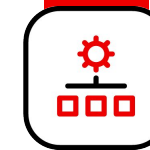
What is it?

Logical collection of your playbooks:

- ▶ Multiple source types supported
- ▶ Source Control Management (SCM) integration and update strategies
- ▶ Red Hat Insights integration
- ▶ Role-based access control (RBAC) and schedules



Projects. Adding your automation content to controller.



Projects

Create New Project



Name *

Description

Organization *

Execution Environment ?

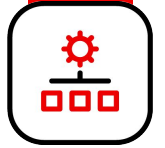
Source Control Type *

Content Signature Validation Credential ?

Save

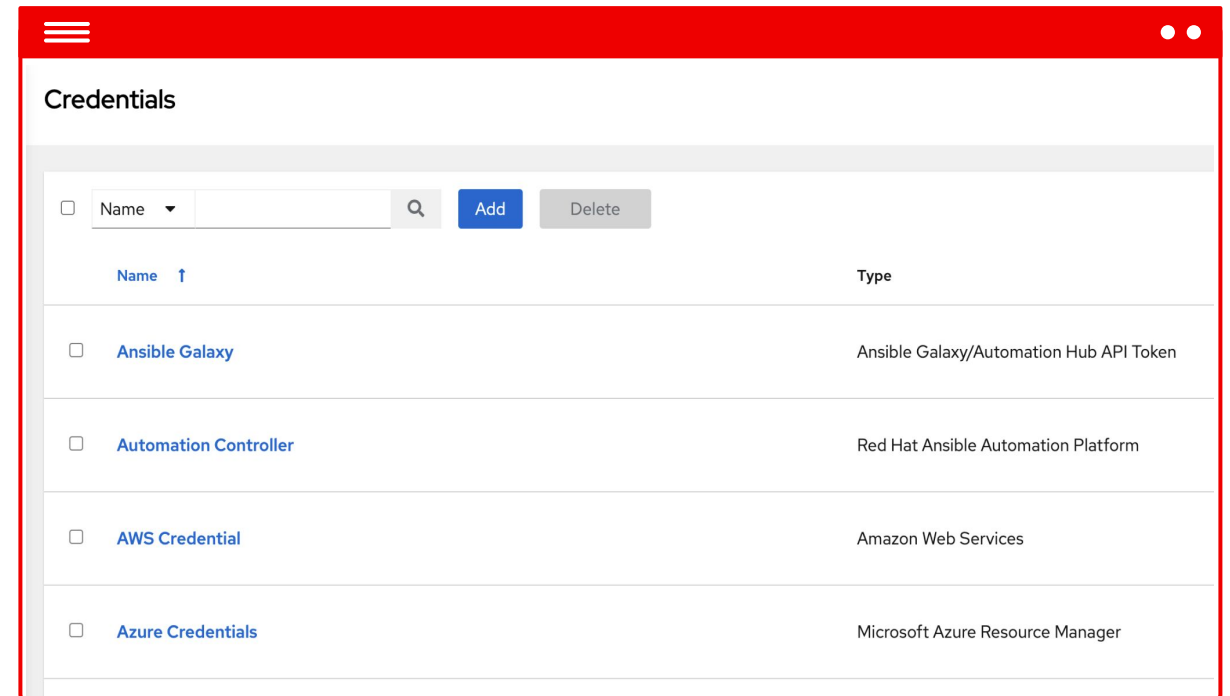
Cancel

Credentials. **Securing resource and endpoint access.**

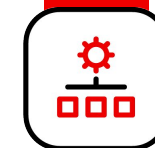


What is it?

- ▶ Securely manage credentials needed for automation resources
- ▶ Multiple credential types supported
- ▶ Integrate external secret management systems
- ▶ Create custom credential types and plugins
- ▶ Use RBAC to govern access
- ▶ Actual credential never exposed



Credentials. **Securing resource and endpoint access.**



Credentials
Create New Credential

Name * Description Organization

Credential Type *
Machine

Type Details

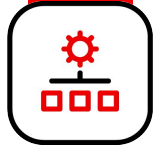
Username Password ☐ Prompt on launch

SSH Private Key

Drag a file here or browse to upload

Browse... Clear

Job Templates. Bringing it all together.



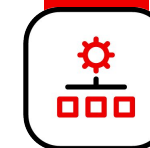
What is it?

- ▶ Define and standardize running automation
- ▶ Reusable and shareable
- ▶ Leverage agile practices, such as GitOps and event-driven automation

The screenshot shows the 'Job template with slicing' configuration page in the Red Hat Ansible Automation Platform. The page has a red header with a menu icon and window controls. Below the header, there are tabs for 'DETAILS', 'PERMISSIONS', 'NOTIFICATIONS', 'COMPLETED JOBS', 'SCHEDULES', and 'ADD SURVEY'. The 'DETAILS' tab is active. The form contains several sections for configuration:

- * NAME:** 'Job template with slicing'
- DESCRIPTION:** (empty)
- * JOB TYPE:** 'Run' (with a 'PR' checkbox)
- * INVENTORY:** 'King PLC' (with a search icon and 'PROMPT ON LAUNCH' checkbox)
- * PROJECT:** 'Project from Git' (with a search icon)
- * PLAYBOOK:** 'gen_host_status.yml'
- CREDENTIALS:** (empty) (with a search icon and 'PROMPT ON LAUNCH' checkbox)
- FORKS:** '0' (with a dropdown arrow)
- LIMIT:** (empty) (with a 'PR' checkbox)
- * VERBOSITY:** '0 (Normal)' (with a dropdown arrow and 'PROMPT ON LAUNCH' checkbox)
- JOB TAGS:** (empty) (with a 'PROMPT ON LAUNCH' checkbox)
- SKIP TAGS:** (empty) (with a 'PR' checkbox)
- LABELS:** (empty)
- INSTANCE GROUPS:** (empty) (with a search icon)
- JOB SLICING:** '1'
- TIMEOUT:** '0' (with a dropdown arrow)
- SHOW CHANGES:** (toggle switch) (with a 'PROMPT ON LAUNCH' checkbox)
- OPTIONS:**
 - ☐ ENABLE PRIVILEGE ESCALATION
 - ☐ ENABLE PROVISIONING CALLBACKS
 - ☐ ENABLE WEBHOOK
 - ☒ ENABLE CONCURRENT JOBS
 - ☐ ENABLE FACT CACHE

Job Templates. Bringing it all together.



Templates

Create New Job Template

Name * ?

Description

Job Type * ? ☐ Prompt on launch

Inventory * ? ☐ Prompt on launch

Project * ?

Execution Environment ? ☐ Prompt on launch

Playbook * ?

Credentials ? ☐ Prompt on launch

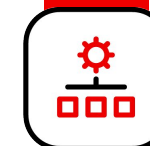
Labels ? ☐ Prompt on launch

Variables ? ☐ Prompt on launch

1 ---

2

Workflows. **Solve complex problems.**



[Templates](#)

Create New Workflow Template



Name *

Description

Organization

Inventory ?

☐ Prompt on launch

Limit ?

☐ Prompt on launch

Source control branch ?

☐ Prompt on launch

Labels ?

☐ Prompt on launch

Variables ?

YAML

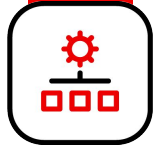
JSON

☐ Prompt on launch



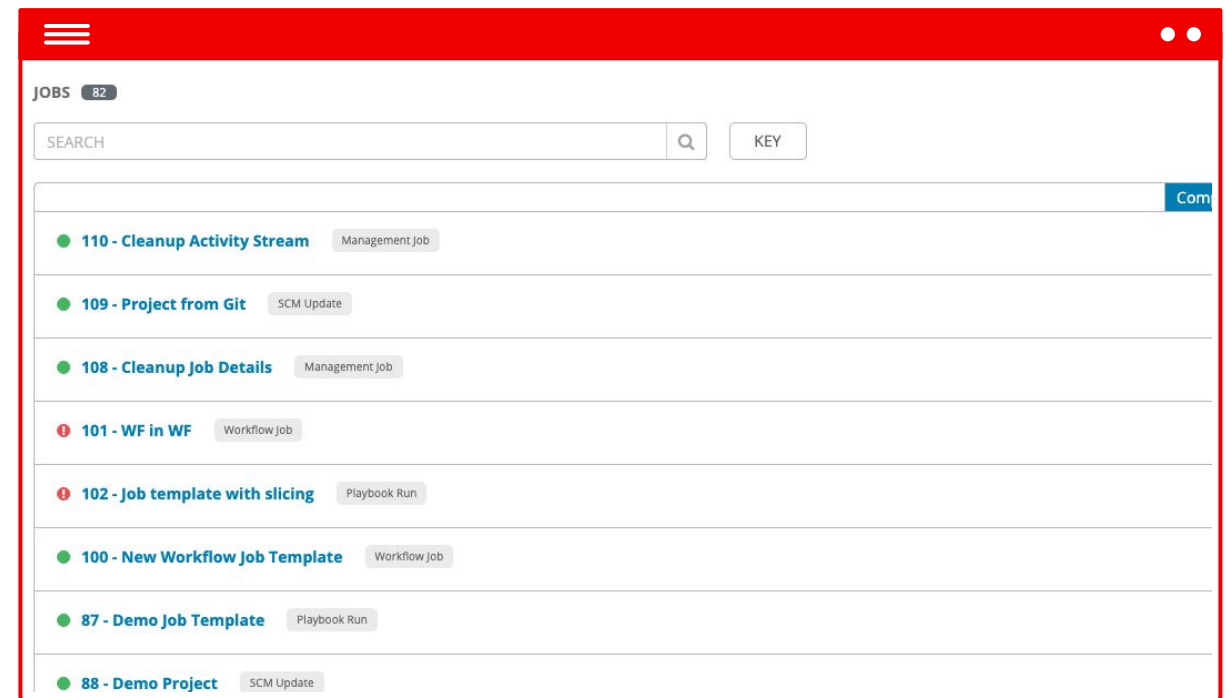
```
1 ---
```


Automation jobs. Executing your defined automation.



What is it?

- ▶ Controller launching an instance of defined automation
- ▶ Relaunch automation jobs
- ▶ Use Job Details to view job outputs
- ▶ Troubleshoot automation execution using filtered views



Lab 2: Inventories

3 minutes

Lab 3: Projects

3 minutes

Lab 4: Credentials

2 minutes

Lab 5: Job Templates

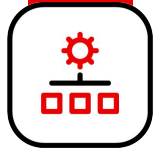
5 minutes

Lab 7: Workflows

10 minutes

Dashboard & recap

Automation controller surveys. **Adopt and grow.**

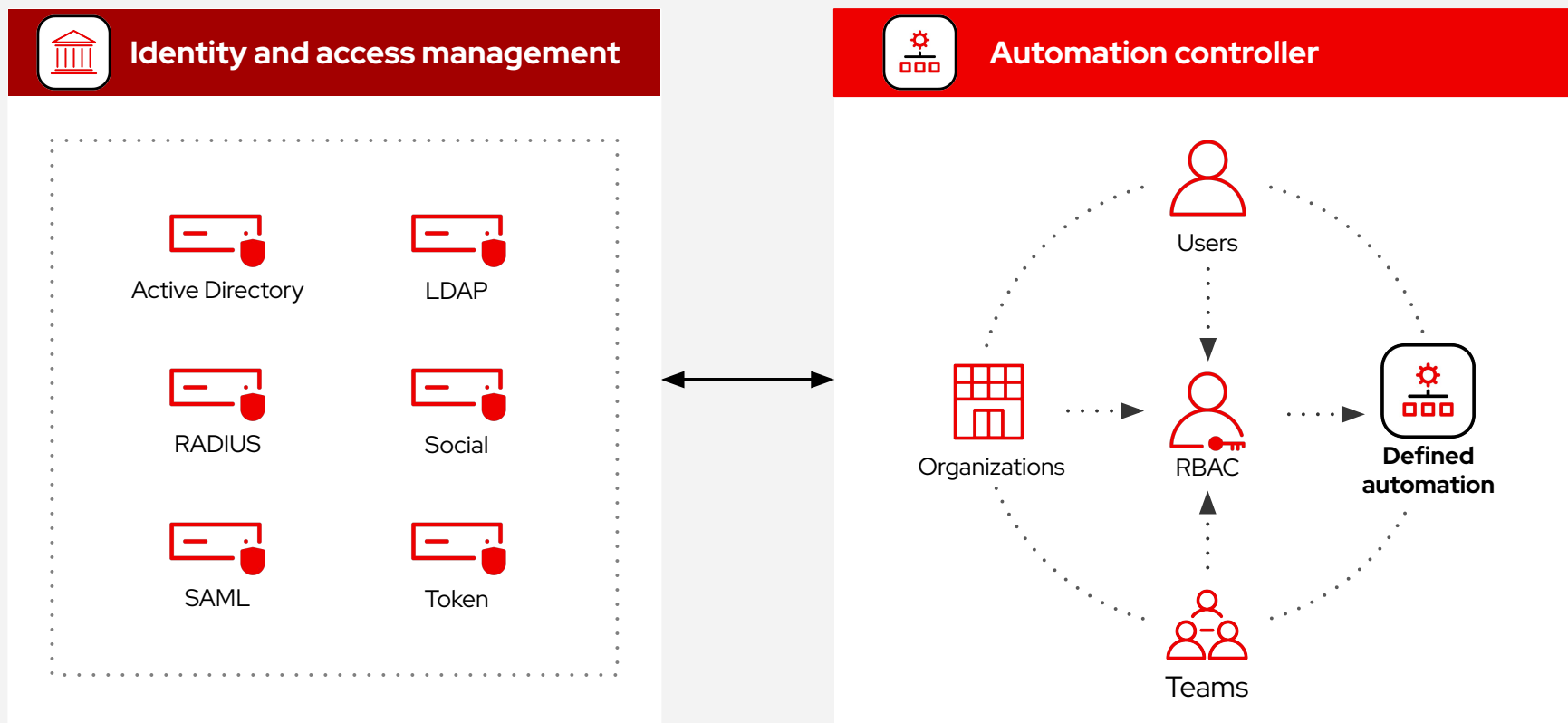


What is it?

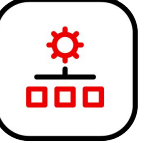
- ▶ User-friendly, self-service interface in automation controller
- ▶ Abstracts complexity using question and answer format
- ▶ Best suited for teams directly accessing automation and close to the automation practice
- ▶ Access and execution governed using controller features

A screenshot of a web application window titled 'LAUNCH JOB | HELLO WORLD'. The window has a red header bar with a hamburger menu icon on the left and window control icons on the right. Below the header, there are two buttons: 'OTHER PROMPTS' and 'SURVEY'. The main content area contains a question: '* WHICH GROUP(S) SHOULD INCLUDE THIS USER?' followed by the instruction 'Enter groups, one per line.' and a text input field. At the bottom of the window, there are two sections: 'INVENTORY' with 'Demo Inventory' and 'CREDENTIAL' with 'Demo Credential'. To the right of these sections are two buttons: 'CANCEL' and 'LAUNCH'.

Securing your content in automation controller. **Delegate.**



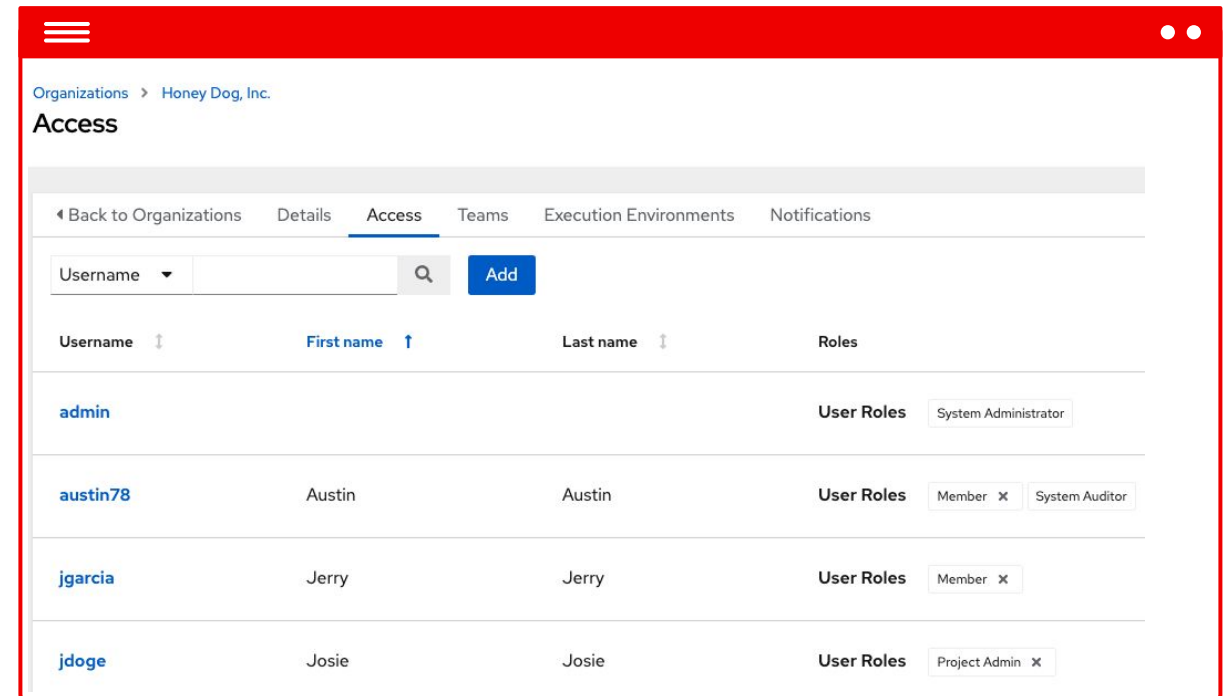
Role-Based Access Control. Who can use my automation?



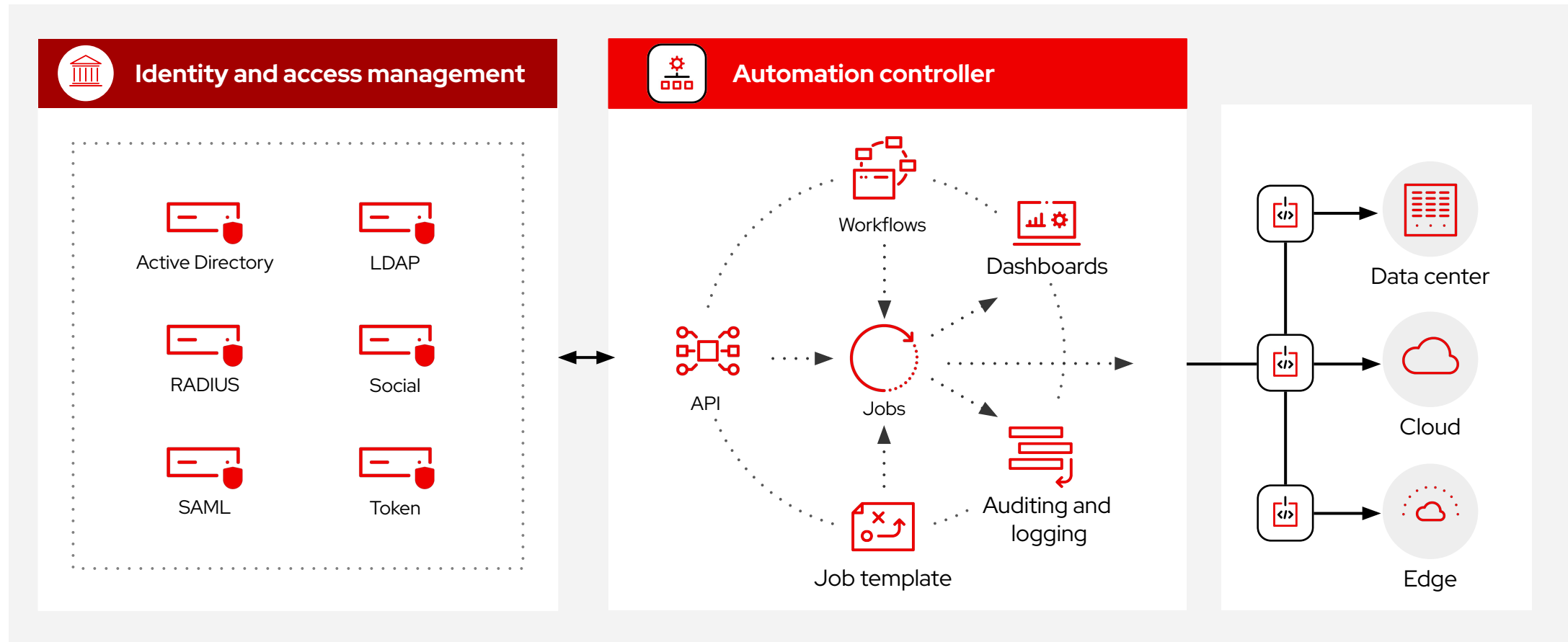
What is it?

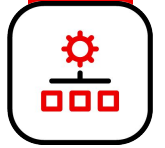
Securely govern access to your automation

- ▶ Logically group controller objects and grant users and teams read, execute, edit permissions
- ▶ Use predefined roles to grant access
- ▶ Integrates with your existing enterprise authentication systems



Launching your content from **automation controller**.



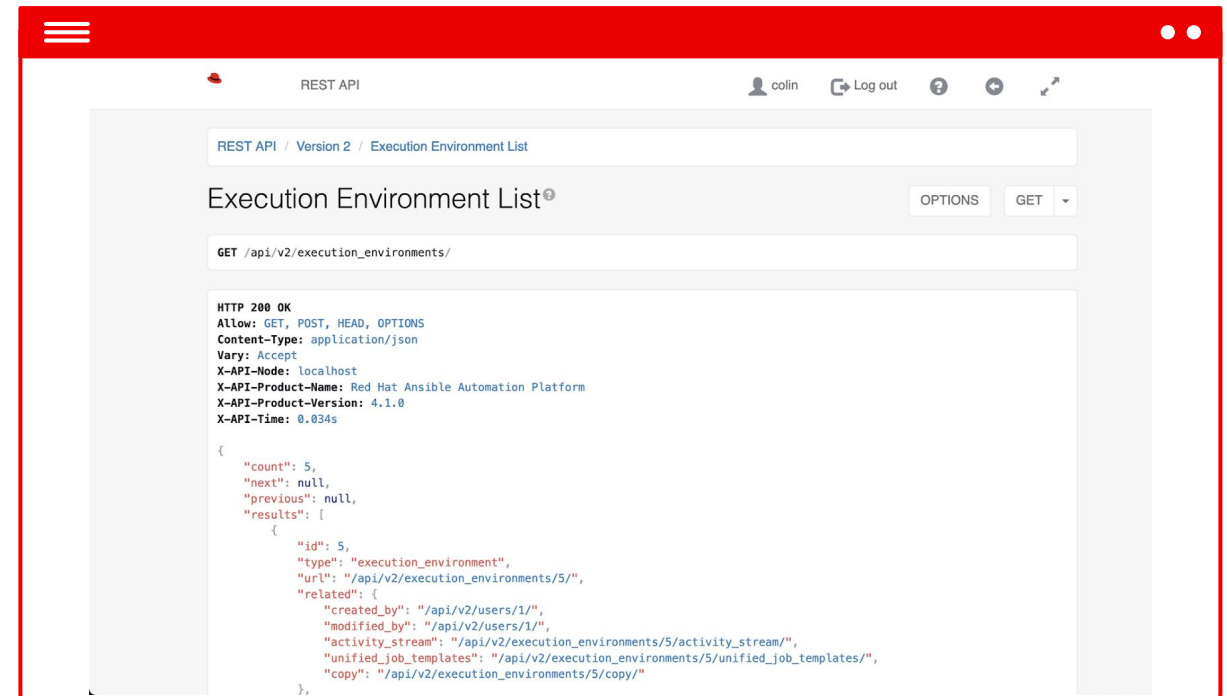


Automation controller API.

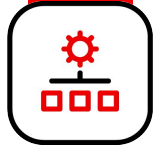
Integrate and elevate your automation.

What is it?

- ▶ The API provides programmatic access to the automation via a defined interface
- ▶ Underneath it is still powered by the same bits and pieces which are at the core: workflows, inventories, etc
- ▶ It offers simple integration into other tools like ITSM, SOAR, etc

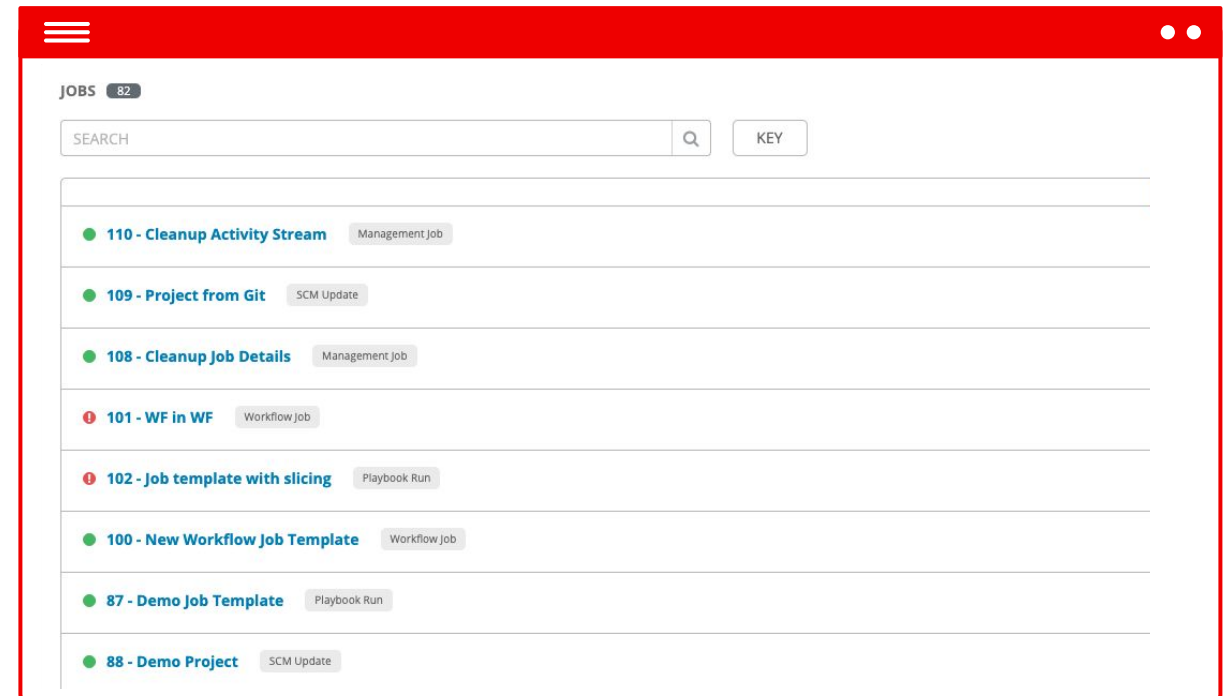


Automation controller logging aggregation.

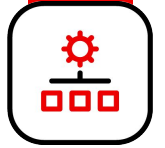


What is it?

- ▶ Provides the capability to send detailed logs to several kinds of 3rd party external log aggregation services
- ▶ Use Red Hat Enterprise Linux rsyslog to aggregate and redirect logs
- ▶ Native integration exists for multiple 3rd party logging systems
- ▶ Multiple detailed logs are gathered for job events, system facts, users, activity streams, and controller features

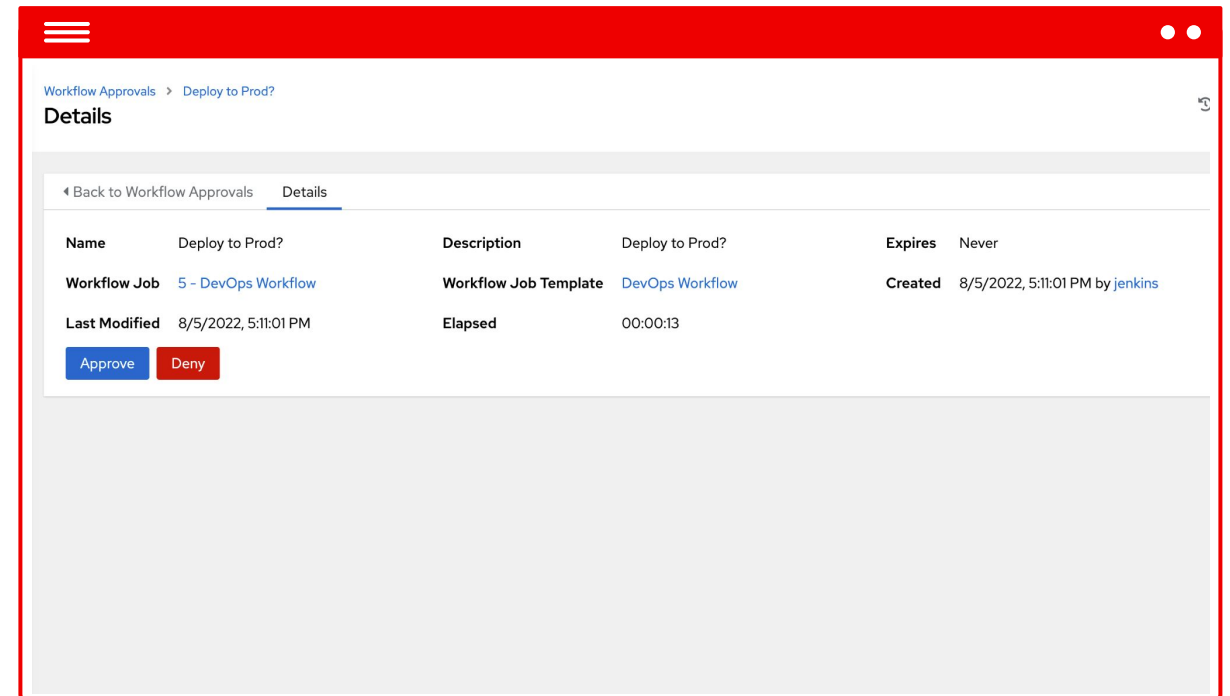


Automation controller approvals. **Integrate and govern.**

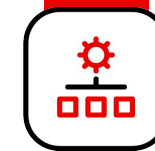


What is it?

- ▶ Adds human interaction to the automation for administration and governance
- ▶ Available at the operational level on the Automation controller UI



Using execution environments in automation controller.



How do I add execution environments (EE)?

EEs can be added via the controller WebUI, API or command line

- ▶ Specify the EE image source
- ▶ Pull policy determines when and how EEs are downloaded when running automation
- ▶ Assign registry credentials to pull the EEs

How do I use execution environments?

- ▶ Select the EE you want to use in your automation job
- ▶ The default EE is used if none is specified

A screenshot of a web browser window showing the 'Create new execution environment' form. The form has a red header bar with a hamburger menu icon on the left and window control icons on the right. The title 'Create new execution environment' is centered below the header. The form contains several input fields: 'Name' with the value 'Security EE', 'Image' with the value 'quay.io/ansible/security-ee:latest', 'Pull' with a dropdown menu set to 'Always pull container before running.', 'Description' (empty), 'Organization' with a search icon, and 'Registry credential' with a search icon. Below the 'Organization' field, there is a note: 'Leave this field blank to make the execution environment globally available.' At the bottom left of the form, there are 'Save' and 'Cancel' buttons.



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.



facebook.com/redhatinc



twitter.com/ansible



linkedin.com/company/red-hat



youtube.com/AnsibleAutomation

What is automation controller?

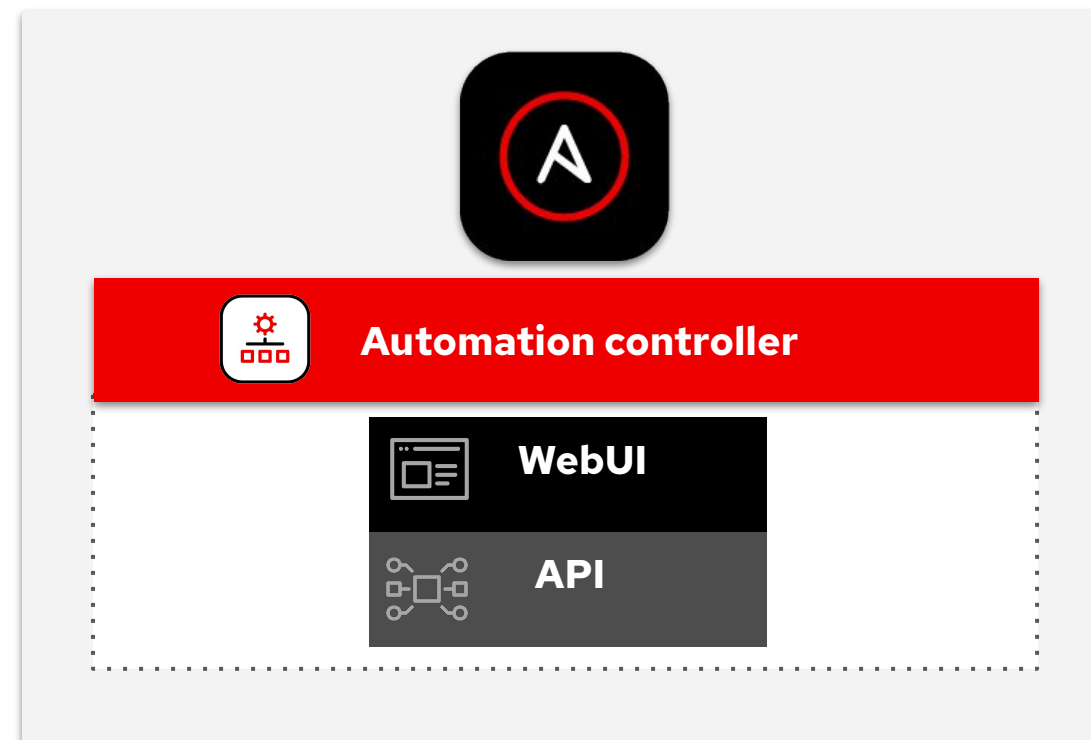


Automation controller (formerly known as Ansible Tower) is a UI and RESTful API allowing you to scale IT automation, manage complex deployments and speed productivity.

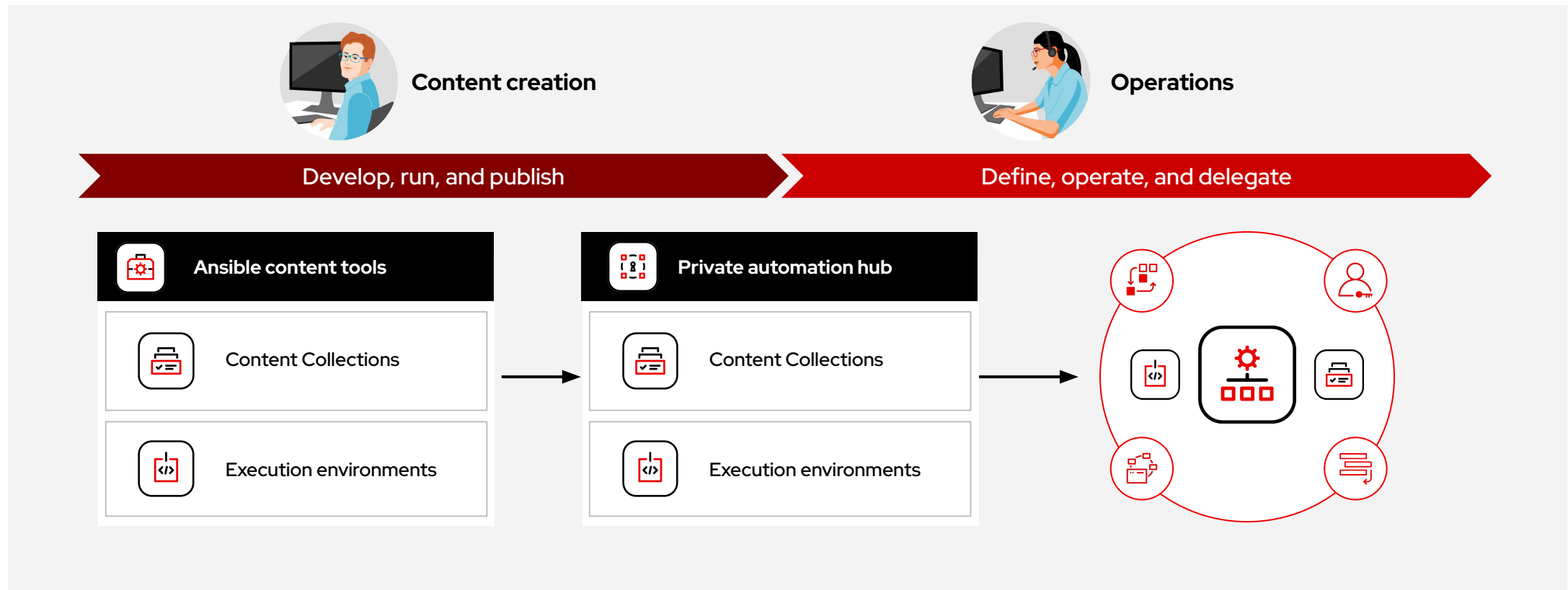
- ▶ Role-based access control
- ▶ Push-button deployment access
- ▶ All automations are centrally logged
- ▶ Powerful workflows match your IT processes

Automation controller 4.0 updates

- ▶ WebUI
 - Refactored to PatternFly 4.0
 - Increased performance
 - Job output filters
 - Distinct “edit” and “read” views
 - Stricter Content Security Policy
- ▶ PostgreSQL 12
 - Installed from RHEL modules
 - Partitioned access - increased performance



Creation to operations handover. **Consistent and portable.**



The automation life cycle. **Manage.**

