Exploring New Possibilities with the AWS Ansible Automation

Rahul Sharma
Amazon Web Services
@rahul031092

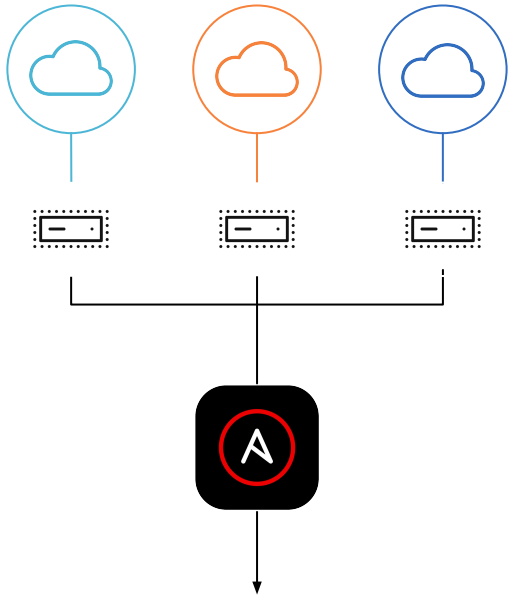
Sean Cavanaugh
Red Hat Ansible
@IPvSean

AnsibleFest

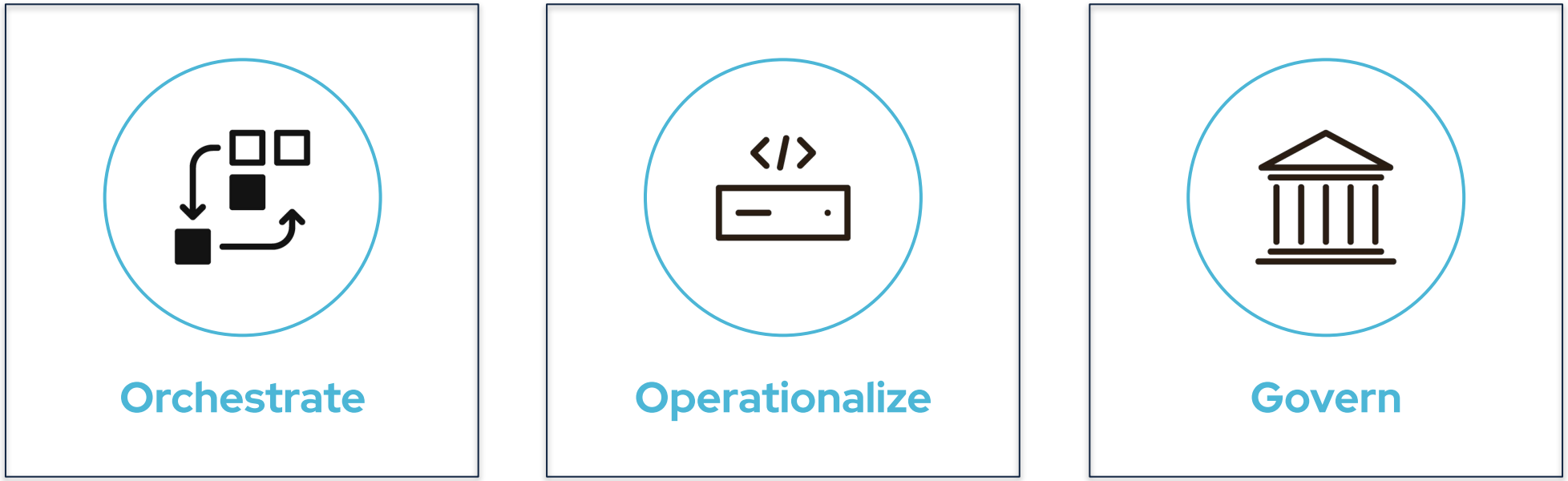
AGENDA

- Intro to Ansible Hybrid Cloud Automation
- Intro to AWS Cloud Control API
- Cloud Control API collection
- Demo of Cloud Control API
- What's new in Ansible AWS content collection 5.0.0
- Q&A

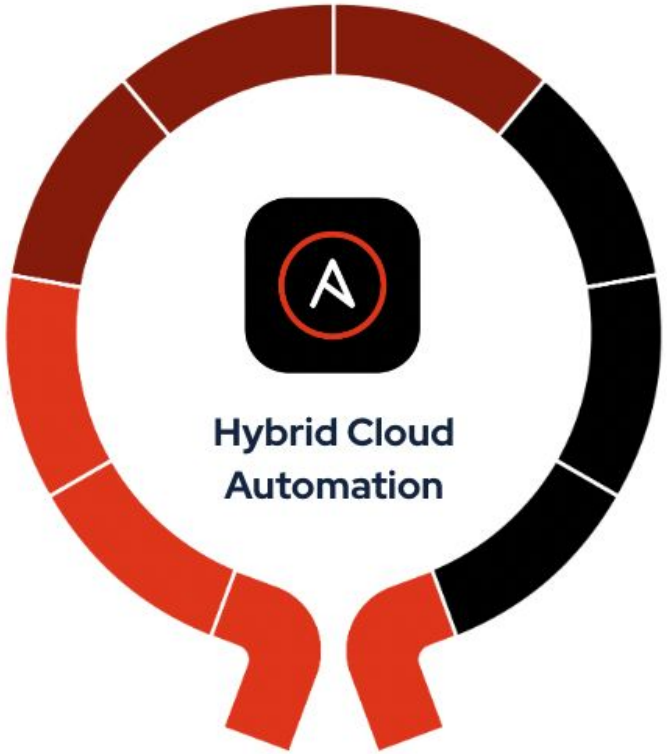
Intro to Ansible Hybrid Cloud Automation



Ansible Automation for the hybrid cloud



Orchestrate | Operationalize | Govern

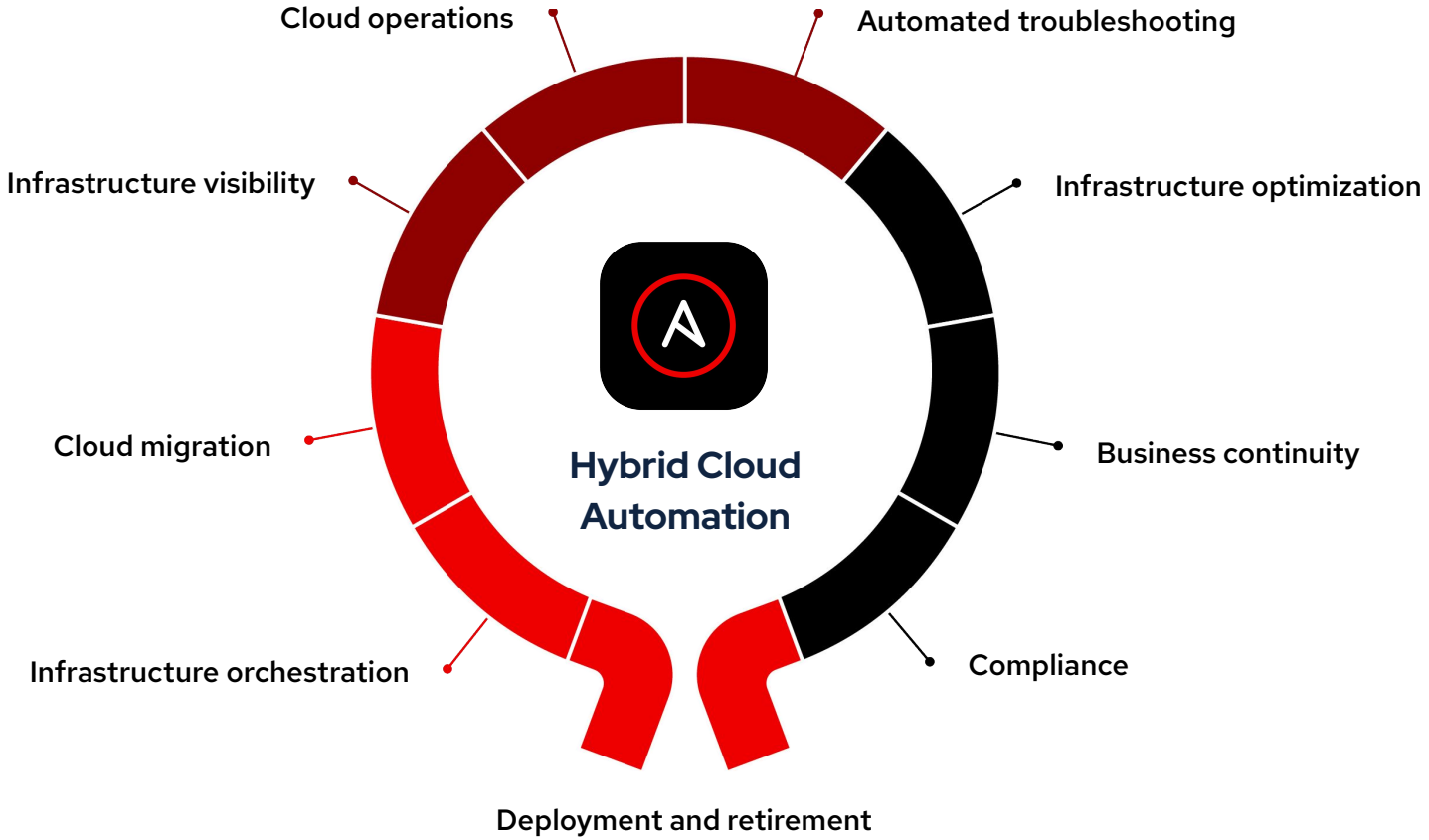


Public cloud

Cloud native

Private cloud

Orchestrate | Operationalize | Govern



Public cloud

Cloud native

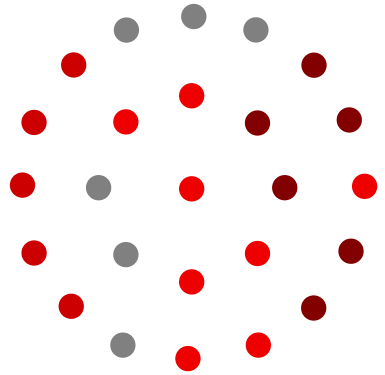
Private cloud

AnsibleFest

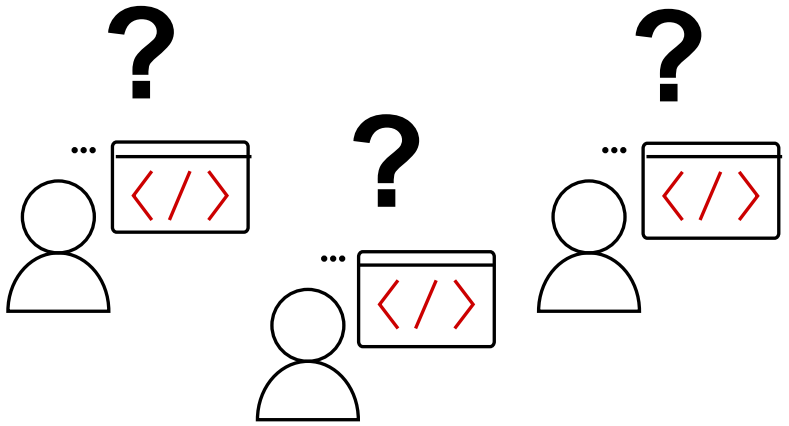
Intro to AWS Cloud Control API

Opportunity #1: Access the latest AWS innovation faster

Developers using APN Partner solutions to build and manage application infrastructure want to accelerate their pace of innovation and time to market for their applications



Can new AWS features and services be supported closer to launch?



Opportunity #2: Automatic support for new AWS capabilities

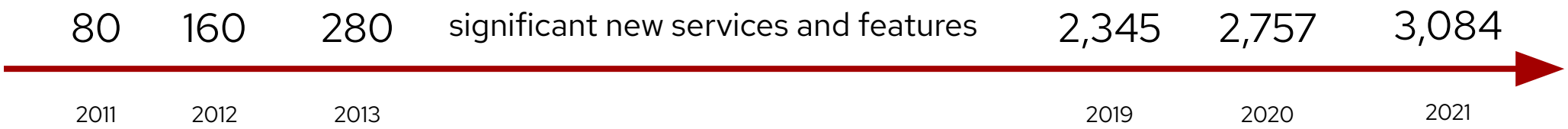
AWS continues to innovate rapidly on behalf of its customers

200+ fully featured services

3,084 significant new features in 2021

Ongoing investments to support new AWS services and features

Few weeks to months for each new AWS resource



How can we automate supporting new AWS capabilities?

Opportunity #3: Standardized APIs to manage AWS services

Multiple AWS services

Applications are becoming increasingly sophisticated

Service-specific API

AWS services and third-party APIs equipped with specific implementation

Author custom code

Variety of API implementation requires engineering effort

How can you use a consistent API method to interact with hundreds of AWS services?

Opportunity #3: Standardized APIs to manage AWS services

How can you use a consistent API method to interact with hundreds of AWS services?

Service-specific API operations

```
aws ec2 run-instances \
  --image-id ami-xxxxxxxx --count 1 \
  --instance-type t2.micro \
  --key-name MyKeyPair \
  --security-group-ids sg-903004f8 \
  --subnet-id subnet-6e7f829e
```

```
aws lambda create-function \
  --function-name my-function \
  --runtime nodejs10.x \
  --zip-file fileb://my-function.zip \
  --handler my-function.handler \
  --role arn:aws:iam::123456789012:role/service-role/My-Function-role
```

```
aws ecs run-task \
  --cluster default \
  --task-definition my-task-def:1
```

Workflow-based API operations

```
aws kinesis create-stream \
  --stream-name samplestream \
  --shard-count 3
```

```
aws kinesis add-tags-to-stream \
  --stream-name samplestream \
  --tags samplekey=example
```

```
aws kinesis increase-stream-retention-period \
  --stream-name samplestream \
  --retention-period-hours 168
```

Introduction to AWS Cloud Control API

Cloud Control API: is a new AWS capability that introduces new standardized APIs that make it easy for developers to leverage the latest AWS capabilities, typically on the day of launch and manage their cloud infrastructure consistently.

AWS Cloud Control API

CreateResource
GetResource
UpdateResource
DeleteResource
ListResources

Consistent CRUD-L

AWS::Amplify::App
AWS::ECS::Cluster
AWS::Logs::LogGroup
...
AWS::S3::Bucket
AWS::Xray::Group

Resource types

Handlers
Error
codes

Standardized behavior

How can users benefit?

Faster access
to new AWS services and features, typically on the day of launch

+

Unified interface
for one-time integration that automatically exposes new AWS launches

+

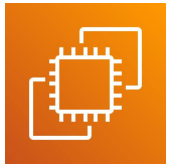
Consistent CRUD + L APIs
designed to make it easy to manage cloud infrastructure consistently.

Introducing the Red Hat AWS Cloud Control Collection

Experimental alpha Collection of generated modules using the AWS Cloud Control API for interacting with AWS Services.

AWS Cloud Control API

AWS CloudFormation Public Registry



Amazon EC2

AWS Transit Gateway

Amazon S3

Amazon Kinesis

Amazon Kendra

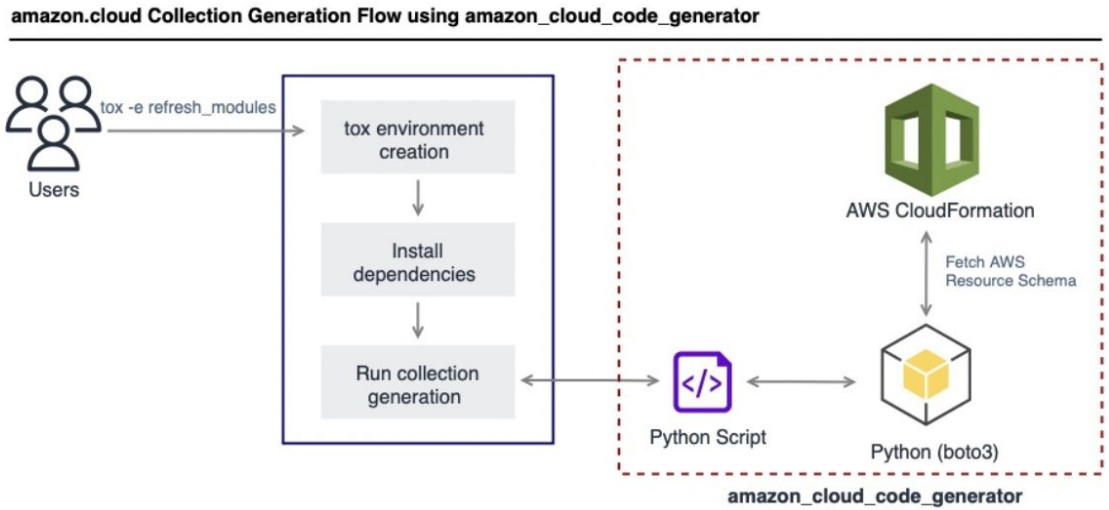
Hundreds of AWS modules and more

Benefits of Red Hat AWS Cloud Control Collection

- ✓ Auto-generated modules based on resources developed and maintained by AWS service teams
- ✓ Rapid introduction of new AWS services and implementation of new features to existing ones.
- ✓ Enable more comprehensive coverage of the vast number of AWS services available.

How does the Ansible AWS Cloud Control Collection Work?

Module Generation: modules contained in this Collection are generated using a tool called amazon_cloud_code_generator - developed and open sourced by the Ansible Cloud team.



Using the amazon.cloud collection:
All the modules of this Collection use boto3 (Amazon Web Services (AWS) Software Development Kit (SDK) for Python) and AWS Cloud Control API (CloudControlApi) client.

(1) boto3 >=1.20.0 and botocore >=1.23.0

Red Hat Ansible AWS Cloud Control collection

What’s next with Ansible AWS Cloud Control Collection?

- ✔ More coverage across AWS services such as Amazon EC2, Amazon S3

Demo

Demo - Screen Share

What's new in Ansible AWS content collection 5.0.0

What's new?!

AWS content collection 5.0.0

NEW!
Released October 4th, 2022

amazon.aws

40 modules migrated

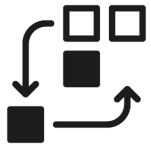
from the community.aws collection
to the amazon.aws collection.

Including:

- autoscaling
- cloudtrail
- cloudwatch
- ec2_eip
- iam
- lambda
- rds
- route53

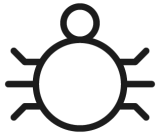
Minimum Versions

botocore	1.21.0
boto3	1.18.0
Python	3.7
ansible-core	2.11



3 NEW modules.

- amazon.aws.**cloudtrail_info**
- amazon.aws.**cloudwatch_metric_alarm_info**
- amazon.aws.**s3_object_info**



Bug Fixes

8 bug fixes:

- aws_ec2
- cloudtrail
- ec2_instance
- ec2_metadata_facts
- module_utils/botocore
- module_utils/elbv2
  (two bugs squashed)
- s3_object

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/ansibleautomation


twitter.com/ansible


linkedin.com/company/ansible/


youtube.com/user/RedHatVideos