Network Automation

Deploy, Validate, Backup and Restore with Ansible

Rohit Thakur
Principal Software Engineer, Ansible Network Engineering team at Red Hat
GitHub: rohitthakur2590
Matrix: @rothakur

ANSIBLE

Agenda

1. Introduction
2. Overview of Ansible's Network Content
3. Create Brownfield Inventory
4. Validating Network Configurations
5. Network Backup and Restore
6. Generating Reports

Introduction

Challenges in Network Automation

- Complexity of managing multiple protocols
- Ensuring consistent configurations
- Minimizing downtime and ensuring quick recovery

How Ansible Helps

- Simplifies automation
- Provides ready to use content
- Scales across diverse environments

Ansible Network Automation

SIMPLE

Human readable automation

No special coding skills needed

Tasks executed in order

Get productive quickly

POWERFUL

Gather Information and Audit

Configuration management

Workflow orchestration

Manage ALL IT infrastructure

AGENTLESS

Agentless architecture

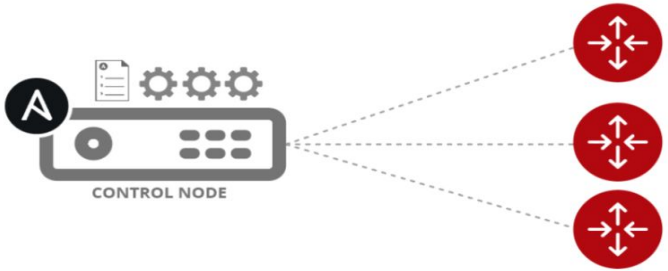
Uses ssh/paramiko

No agent to exploit or update

More efficient and more secure

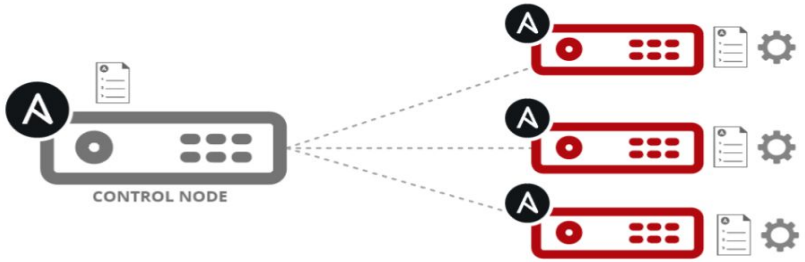
Ansible Network Automation

Module code is executed locally on the control node



NETWORKING DEVICES

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



LINUX/WINDOWS HOSTS

Ansible Network Automation

Ansible offers **ready-to-use** solutions for various networking **use cases** through its network validated content collections. These collections provide a comprehensive set of roles, plugins, and modules designed to simplify network automation by delivering standardized, pre-tested, and production-ready automation workflows.

Collections

network.base – Core modules and roles for managing devices and interfaces.

network.bgp – BGP routing configuration and validation.

network.ospf – OSPF routing setup and monitoring.

network.interfaces – Interface configuration, validation, and monitoring.

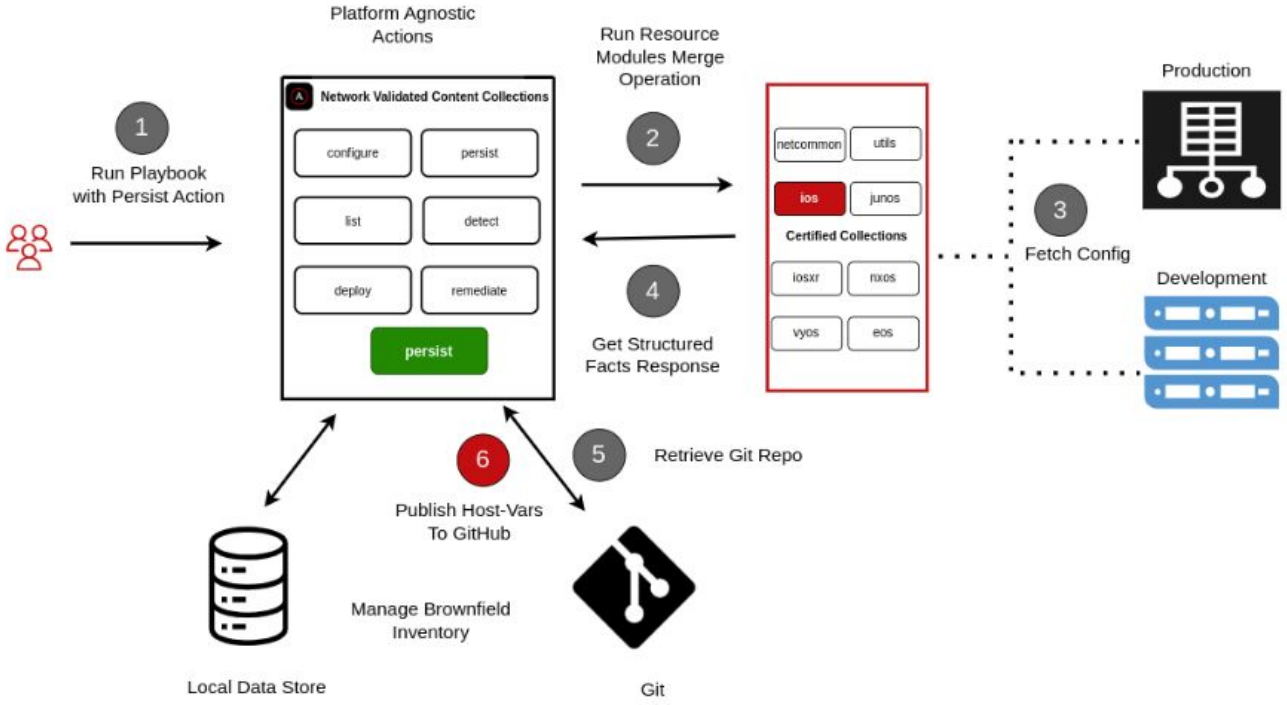
network.backup – Automated backup and restore of network configurations.

Collections used in use cases

Collections:

- network.base enables user to create SOT, remediate.
- network.interfaces enables to perform healh-checks.
- network.interfaces enables to perform healh-checks.

Network Validated Content



Network Validated Content

list:

Retrieve and display supported manageable resources.

deploy:

Deploy consistent network configurations.

detect:

Identify configuration drifts and discrepancies.

remediate:

Correct configuration drifts and restore compliance.

gather:

Collect running configurations from network **devices.**

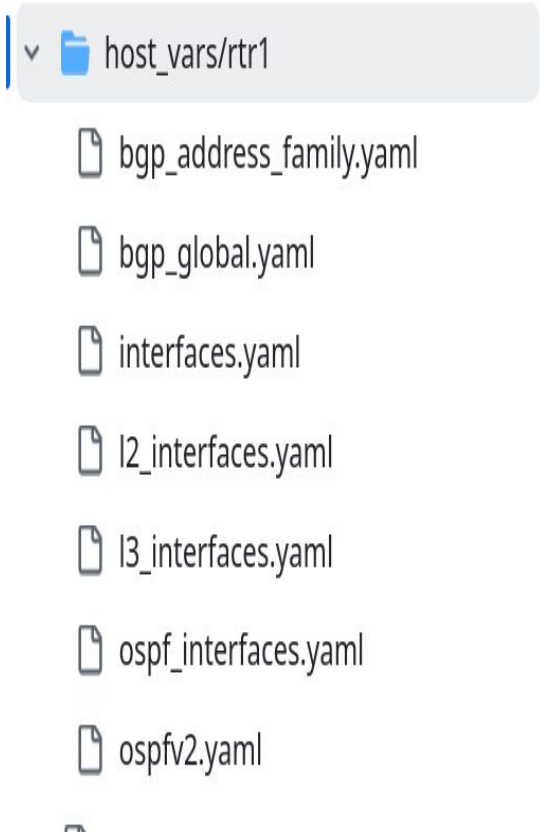
persist:

Save configurations and facts to local or remote storage.

Create Brownfield Inventory

```
---
- name: Create Brownfield Inventory
  hosts: all
  gather_facts: true
  tasks:
    - name: Invoke persist role
      ansible.builtin.include_role:
        name: network.base.persist
      vars:
        data_store:
          scm:
            origin:
              url: "{{ gh_scm_url }}"
              token: "{{ gh_token }}"
              user:
                name: "{{ gh_username }}"
                email: "{{ gh_email }}"
```

Create Brownfield Inventory

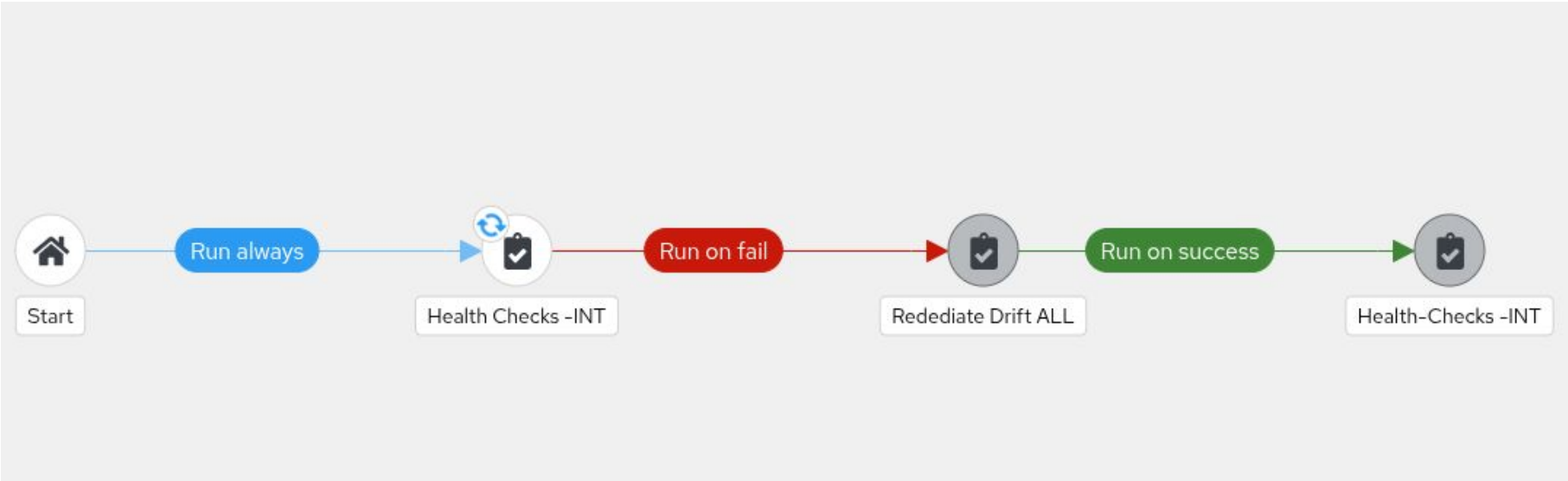



```
interfaces:
-   duplex: auto
    enabled: true
    name: GigabitEthernet0/0
    speed: auto
-   duplex: auto
    enabled: true
    name: GigabitEthernet0/1
    speed: auto
-   duplex: auto
    enabled: true
    name: GigabitEthernet0/2
    speed: auto
-   duplex: auto
    enabled: true
    name: GigabitEthernet0/3
    speed: auto
```

Run health-checks

```
---
- name: Perform interfaces health checks
  hosts: all
  tasks:
  - name: Invoke health_checks role
    ansible.builtin.include_role:
      name: network.interfaces.health_checks
    vars:
      interfaces_health_check:
        name: health_check
        vars:
          details: True
          checks:
            - name: all_operational_state_up
            - name: all_admin_state_up
```

Demo time


Run always
Run on fail
Run on success
Start
Health Checks -INT
Redediate Drift ALL
Health-Checks -INT

Contributing to the Ansible community

Look through community collections, such as community.general, to find easy fix and good first issues.

Ansible documentation: How can I help?

Forum guides:

- How to create an Ansible collection with a simple module
- How to set up a repository for an Ansible collection step by step
- How to release an Ansible collection step by step
- How to submit your first contribution to Ansible step-by-step

Experiment and contribute to the community

Look through community collections, such as community.general, to find easy fix and good first issues.

- How do I get started with network automation if I am new to Ansible?
  - **Step 1:** Learn YAML & Jinja2 basics.
  - **Step 2:** Use Ansible for device/networking labs (Vms/Cisco DevNet, CML, EVE-NG).
  - **Step 3:** Start with ios_command, ios_config playbooks.
  - **Step 4:** Join Ansible community & network automation meetups.

Anyone can contribute. **We are all community.**

Join the forum to participate in discussions and get help!

Want to contribute? Find out how to get involved.

Questions ?

Thanks!

Join the discussion: https://forum.ansible.com/tag/network
GitHub: **rohitthakur2590**
Matrix: **#network:ansible.com**

ANSIBLE