Introducción a Ansible: Herramientas de gestión de configuración e infraestructura como código

Luciano Sánchez

January 29, 2025

Índice

Historia de Ansible

- Desarrollo inicial por Michael DeHaan en 2012.
- Adquisición por Red Hat en 2015.
- Origen del término "ansible" en la literatura de ciencia ficción por Ursula K. Le Guin.

Arquitectura de Ansible

- Arquitectura sin agentes.
- Utiliza SSH o Windows Remote Management para operar.
- Módulos temporales ejecutados en nodos gestionados.

Plataformas soportadas

- **Sistemas operativos compatibles:**
  - **Linux:** Ansible se ejecuta en cualquier distribución de Linux como CentOS, Debian, Fedora, Ubuntu, etc.
  - **Unix-like:** Incluye sistemas como FreeBSD, Solaris y otros que comparten características similares a Unix.
  - **Windows:** No nativamente soportado para el nodo de control, pero puede ser gestionado a través del Subsistema de Windows para Linux (WSL) o gestionar nodos Windows usando módulos específicos y WinRM para la comunicación.
- **Requisitos de Python y otras dependencias:**
  - **Python:** Ansible requiere Python (versión 2.7 o 3.5 y posteriores) en el nodo de control.
  - **Otras dependencias:** Dependiendo de las funciones utilizadas, pueden requerirse paquetes adicionales como ‘pywinrm‘ para gestionar Windows a través de WinRM, o paquetes de cifrado para soportar Ansible Vault.

Componentes Principales de Ansible

- **Nodos de Control**:
  - Son los servidores desde los cuales Ansible se ejecuta y administra los nodos remotos.
  - Los nodos de control se conectan a los nodos gestionados, a través de SSH para sistemas Unix-like o WinRM para sistemas Windows.
- **Módulos**:
  - Son unidades de código que Ansible ejecuta directamente en los nodos gestionados.
  - Cada módulo es diseñado para ser idempotente, asegurando que la ejecución repetida del mismo módulo en un entorno en un estado dado producirá siempre el mismo resultado.
  - Ansible tiene una amplia biblioteca de módulos incorporados que pueden gestionar tareas como la instalación de paquetes, manejo de usuarios, manejo de servicios, trabajo con archivos, etc.
- **Playbooks**:

Componentes Principales de Ansible

- **Nodos de Control**:
- **Módulos**:
- **Playbooks**:
  - Son archivos en formato YAML que Ansible utiliza para definir, configurar, y orquestar procedimientos en los nodos gestionados.
  - Cada playbook puede contener uno o varios 'plays', cada uno de los cuales puede dirigirse a un conjunto diferente de hosts para tareas específicas.
  - Los playbooks son la base para scripts de automatización, permitiendo la gestión de configuraciones, despliegues de aplicaciones, y orquestación de servicios a través de múltiples máquinas.

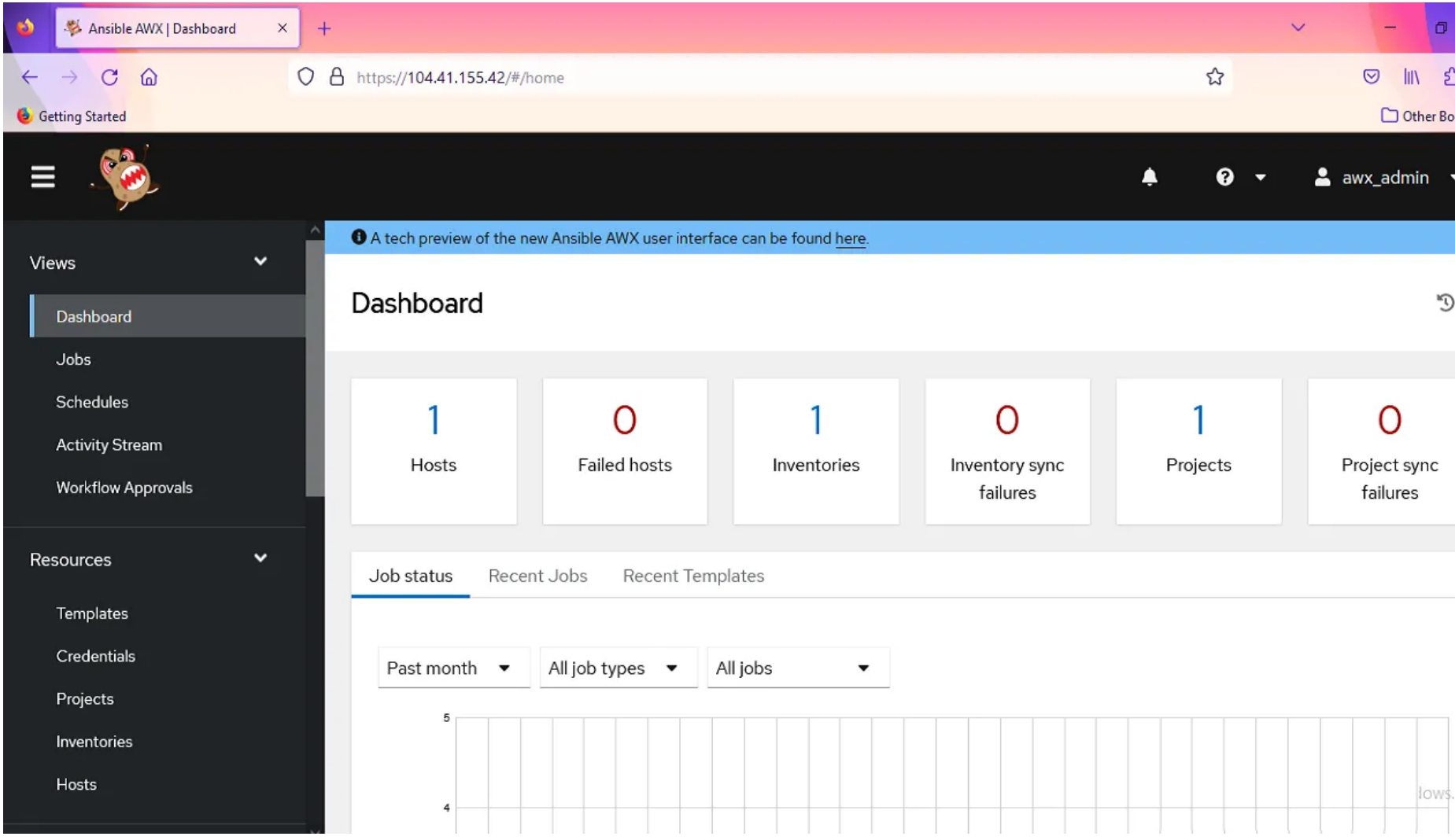
```yaml
---
# Esto es un comentario
usuario:
  nombre: Juan Pérez
  ocupacion: Desarrollador
  habilidades:
    - Python
    - Ansible
    - Cloud
direcciones:
  casa:
    calle: "Calle Falsa 123"
    ciudad: "Ciudad Inventada"
  trabajo:
    calle: "Avenida Siempre Viva 742"
    ciudad: "Springfield"
```

Plataforma de Automatización de Ansible (AAP)

- La Ansible Automation Platform (AAP) es un entorno integrado que extiende las capacidades de Ansible.
- Facilita la creación, compartición y gestión de la automatización a nivel empresarial.
- Combina herramientas para el desarrollo de automatización, operaciones de TI y análisis de seguridad.

Componentes de AAP: AWX y la Interfaz Web

- **AWX:** Versión de código abierto de Ansible Tower, sirve como la interfaz gráfica de usuario y API para Ansible.
  - Permite a los usuarios gestionar inventarios, lanzar trabajos y configurar notificaciones y credenciales.
- **Interfaz web de Ansible Tower:** Proporciona un dashboard visual para controlar y visualizar operaciones de automatización.
  - Incluye características como rastreo de trabajos en tiempo real, logs, y reportes de auditoría.


Ansible AWX | Dashboard
https://104.41.155.42/#/home
Getting Started
awx_admin
Views
Dashboard
Jobs
Schedules
Activity Stream
Workflow Approvals
Resources
Templates
Credentials
Projects
Inventories
Hosts
A tech preview of the new Ansible AWX user interface can be found here.
Dashboard
1
Hosts
0
Failed hosts
1
Inventories
0
Inventory sync failures
1
Projects
0
Project sync failures
Job status
Recent Jobs
Recent Templates
Past month
All job types
All jobs
5
4

Automatización a Escala con Automation Mesh

- Automation Mesh permite configurar y gestionar la infraestructura de automatización distribuida.
- Proporciona capacidad de gestión en clusters para ejecuciones de Ansible a gran escala.
- Ejemplo:
  - Una organización con múltiples centros de datos puede configurar nodos de Automation Mesh en cada ubicación para optimizar la carga de trabajo y mejorar la redundancia.

Ejemplo 1: Instalación de Apache

Objetivo: Automatizar la instalación de un servidor web Apache en un servidor Linux.

- Instala Apache en los hosts designados bajo el grupo 'webservers'.
- Utiliza el módulo ‘apt‘ para instalar el paquete, asegurándose de que el cache esté actualizado.
- Garantiza que el servicio Apache esté corriendo y habilitado para iniciar en el arranque.

Instalación de Apache - Playbook

Instalación de Apache

```
---
- hosts: webservers
  become: yes
  tasks:
    - name: Install Apache
      apt:
        name: apache2
        state: present
        update_cache: yes
```

Instalación de Apache - Playbook

Asegurar que Apache esté corriendo

```
- name: Ensure Apache is running
  service:
    name: apache2
    state: started
    enabled: yes
```

Ejemplo 2: Gestión de Usuarios

Objetivo: Crear un usuario y asignar permisos específicos en servidores Unix.

- Crea un usuario 'ASR' con acceso a Bash y lo añade al grupo sudo.
- Ajusta los permisos del directorio especificado para que solo el usuario 'ASR' tenga acceso completo.

Gestión de Usuarios - Playbook

Crear Usuario

```
---
- hosts: all
  become: yes
  tasks:
    - name: Ensure user 'ASR' is present
      user:
        name: ASR
        state: present
        shell: /bin/bash
        groups: sudo
```

Gestión de Usuarios - Playbook

Configurar Permisos de Directorio

```
- name: Set directory permissions
  file:
    path: /some/directory
    state: directory
    owner: ASR
    group: ASR
    mode: '0755'
```