



Análisis y Diseño de Sistemas 2 [B]

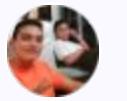
Luis Angel Barrera Velásquez

AGENDA

- 1 Ejemplos Terraform
- 2 Teoria Terraform
- 3 Teoria Ansible
- 4 Dudas Fase 3

Terraform: Infraestructura como código

Terraform es una herramienta de código abierto que permite a los desarrolladores y equipos de infraestructura definir y aprovisionar recursos de infraestructura de manera declarativa. Esto les permite gestionar la infraestructura de forma eficiente, reproducible y escalable.

 by **Luis Angel Barrera
Velásquez**





¿Qué es Terraform?

1 Lenguaje declarativo

Terraform utiliza un lenguaje de alto nivel para describir la infraestructura deseada, en lugar de comandos imperativos.

2 Multiplataforma

Terraform es compatible con múltiples proveedores de nube, como AWS, Azure, GCP y más.

3 Open Source

Terraform es un proyecto de código abierto con una gran comunidad que contribuye a su desarrollo.

Ventajas de Terraform

Reproducibilidad

Terraform permite crear y gestionar entornos idénticos, lo que facilita la implementación y el despliegue de aplicaciones.

Colaboración

Varios miembros del equipo pueden trabajar en paralelo en la misma infraestructura sin conflictos.

Eficiencia

Terraform automatiza tareas manuales y reduce el tiempo y esfuerzo requerido para gestionar la infraestructura.

Principios de Terraform

Declarativo

Terraform se basa en un lenguaje declarativo que describe el estado deseado de la infraestructura.

Idempotente

Terraform puede aplicar o revertir cambios sin afectar el estado actual de la infraestructura.

Modular

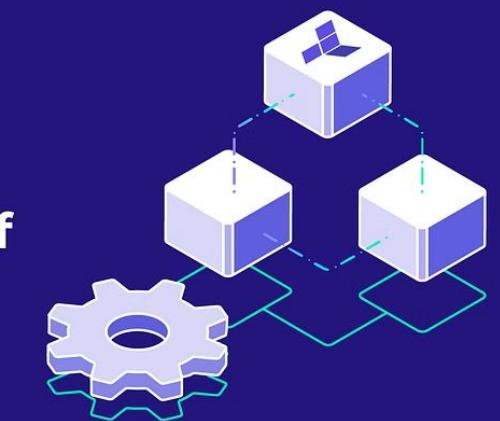
La arquitectura de Terraform permite reutilizar y compartir código entre diferentes proyectos.

Multiproveedor

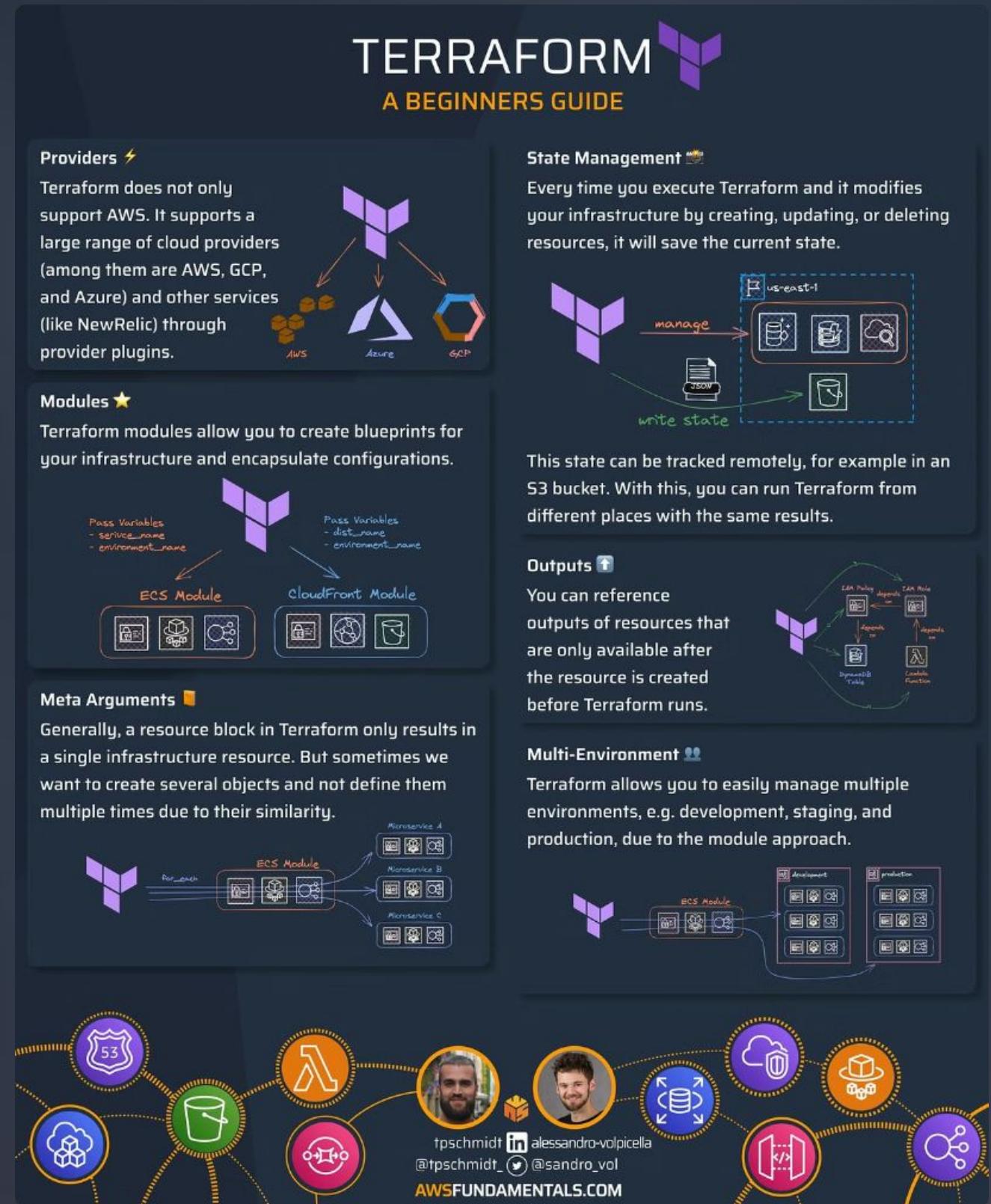
Terraform admite una amplia gama de proveedores de infraestructura, lo que brinda flexibilidad.



[What is a Terraform Plan and Examples of How it Works](#)



Sintaxis y recursos de Terraform



- 1 Bloques**
Terraform utiliza bloques de configuración para definir recursos, módulos y variables.
- 2 Recursos**
Los recursos representan los componentes de infraestructura que se crearán, como instancias de EC2, VPCs o bases de datos.
- 3 Salidas**
Las salidas permiten exponer información importante sobre la infraestructura, como direcciones IP o IDs de recursos.

Flujo de trabajo de Terraform



Escribir

Crear archivos de configuración de Terraform que describan la infraestructura deseada.



Planificar

Revisar los cambios que Terraform aplicará a la infraestructura.



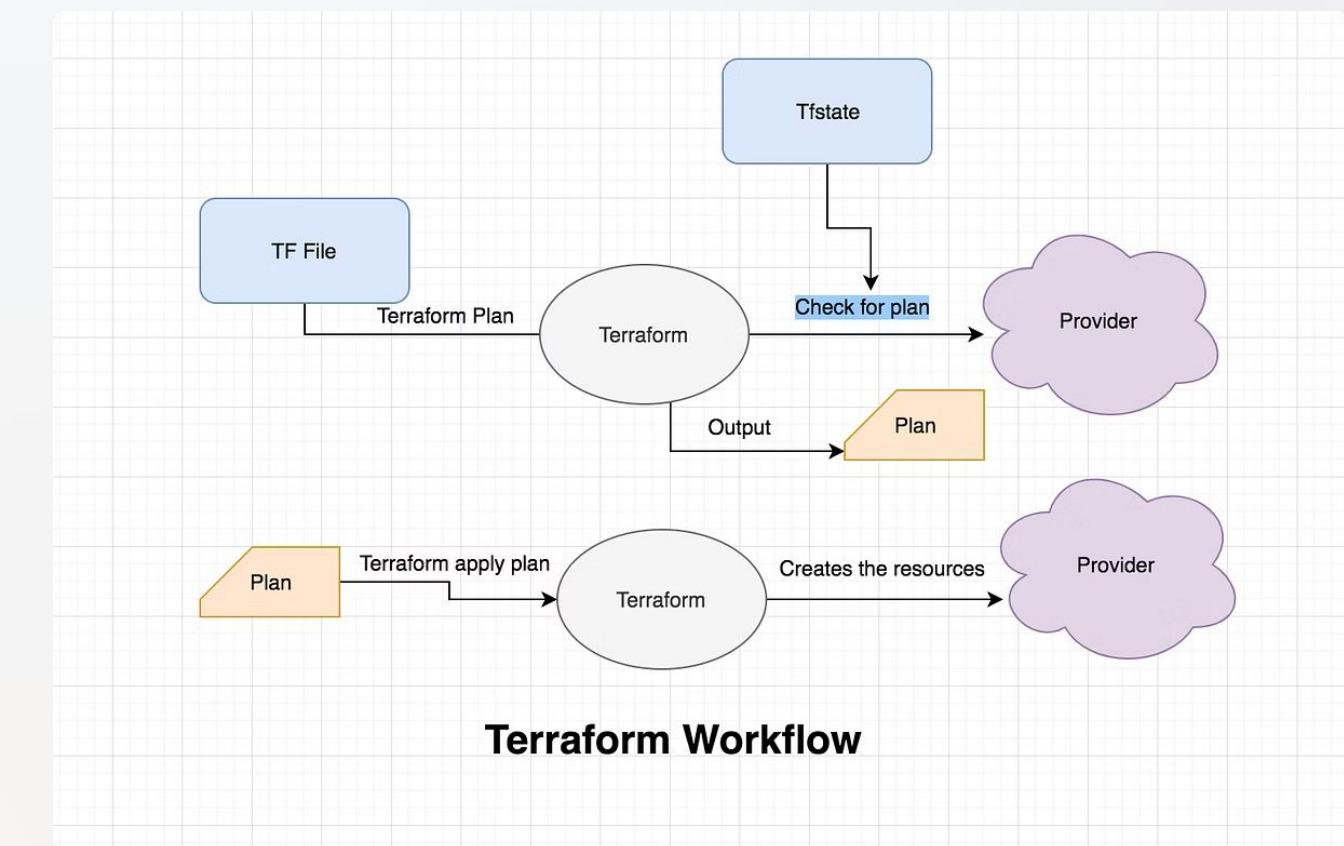
Aplicar

Ejecutar Terraform para crear o actualizar la infraestructura según la configuración.



Destruir

Eliminar la infraestructura cuando ya no se necesite.



Módulos y proveedores en Terraform

Módulos

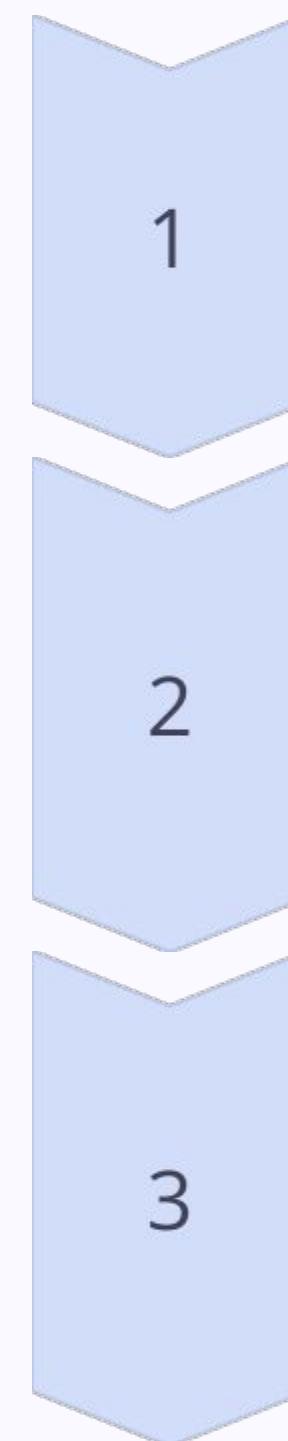
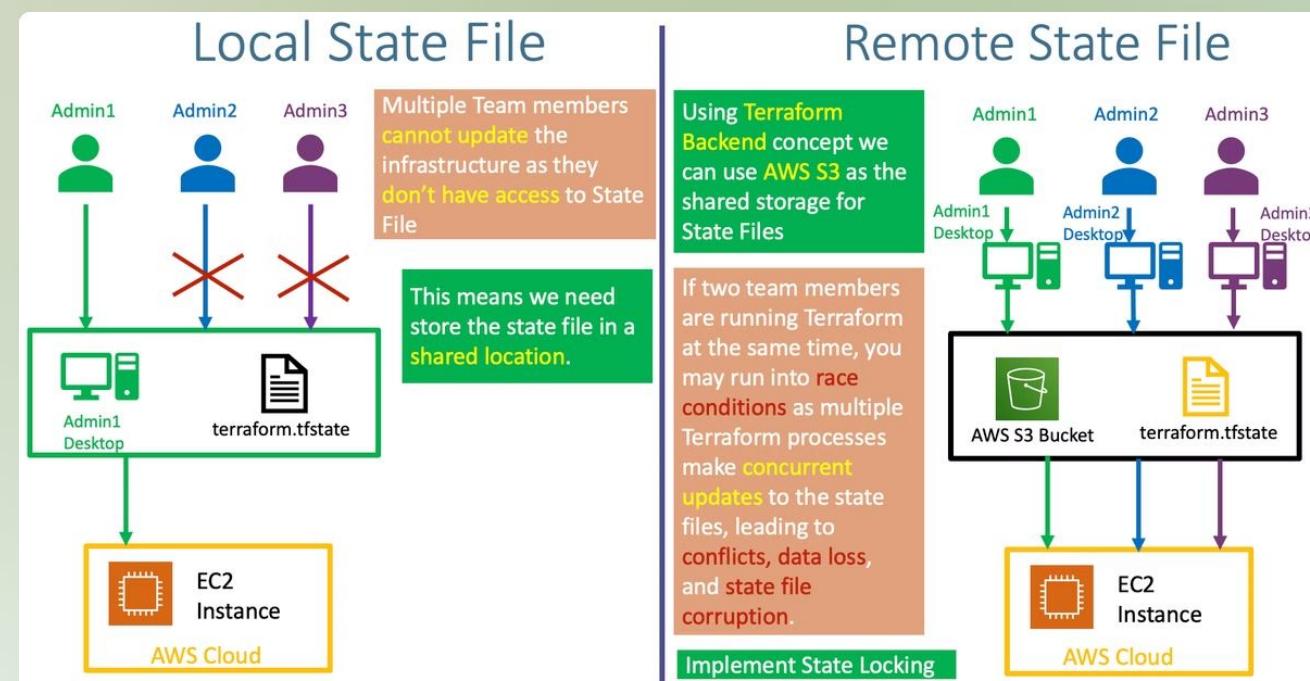
Los módulos permiten reutilizar y compartir bloques de configuración entre diferentes proyectos.

Proveedores

Terraform admite una amplia variedad de proveedores de infraestructura, lo que permite la gestión de recursos en múltiples plataformas.



Gestión del estado en Terraform



Estado local

Terraform almacena el estado de la infraestructura en archivos locales.

Estado remoto

Terraform también admite el almacenamiento del estado en ubicaciones remotas, como Bucket de S3 o Blob Storage de Azure.

Bloqueo de estado

Terraform implementa bloqueo de estado para evitar conflictos entre múltiples usuarios que trabajan en la misma infraestructura.



Casos de uso y mejores prácticas

1 Automatización de despliegues

Terraform permite automatizar el despliegue y la gestión de la infraestructura, lo que acelera los procesos y reduce los errores.

2 Entornos efímeros

Terraform facilita la creación y eliminación de entornos de infraestructura de manera rápida y escalable.

3 Cumplimiento normativo

La gestión de la infraestructura como código con Terraform ayuda a mantener la consistencia y el cumplimiento de las políticas.

Introducción a Ansible

Ansible es una herramienta de código abierto para la automatización de infraestructura y procesos. Permite a los equipos de TI administrar, aprovisionar y orquestar sus sistemas de manera sencilla y eficiente.



by Luis Angel Barrera
Velásquez



Image ID: 2E43DRH
www.alamy.com

Qué es Ansible y cómo funciona

1 Control Remoto

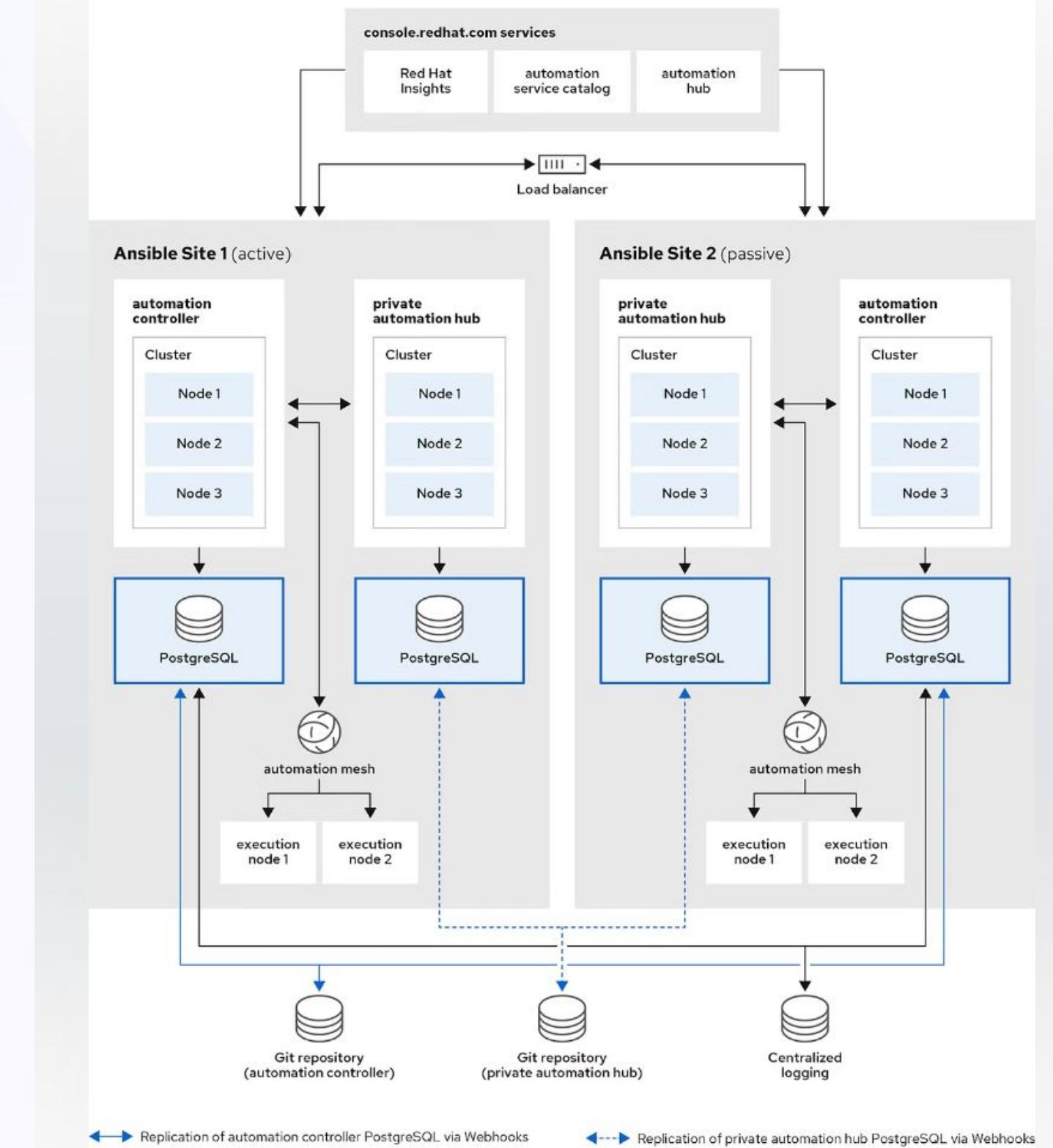
Ansible utiliza SSH para conectarse y controlar de forma remota los diferentes sistemas.

3 Agente Menos

No requiere la instalación de agentes en los nodos administrados, lo que simplifica su implementación.

2 Declarativo

Los usuarios definen el estado deseado y Ansible se encarga de alcanzarlo.



Beneficios de Ansible para la automatización

Eficiencia

Ansible automatiza tareas repetitivas y reduce el riesgo de errores humanos.

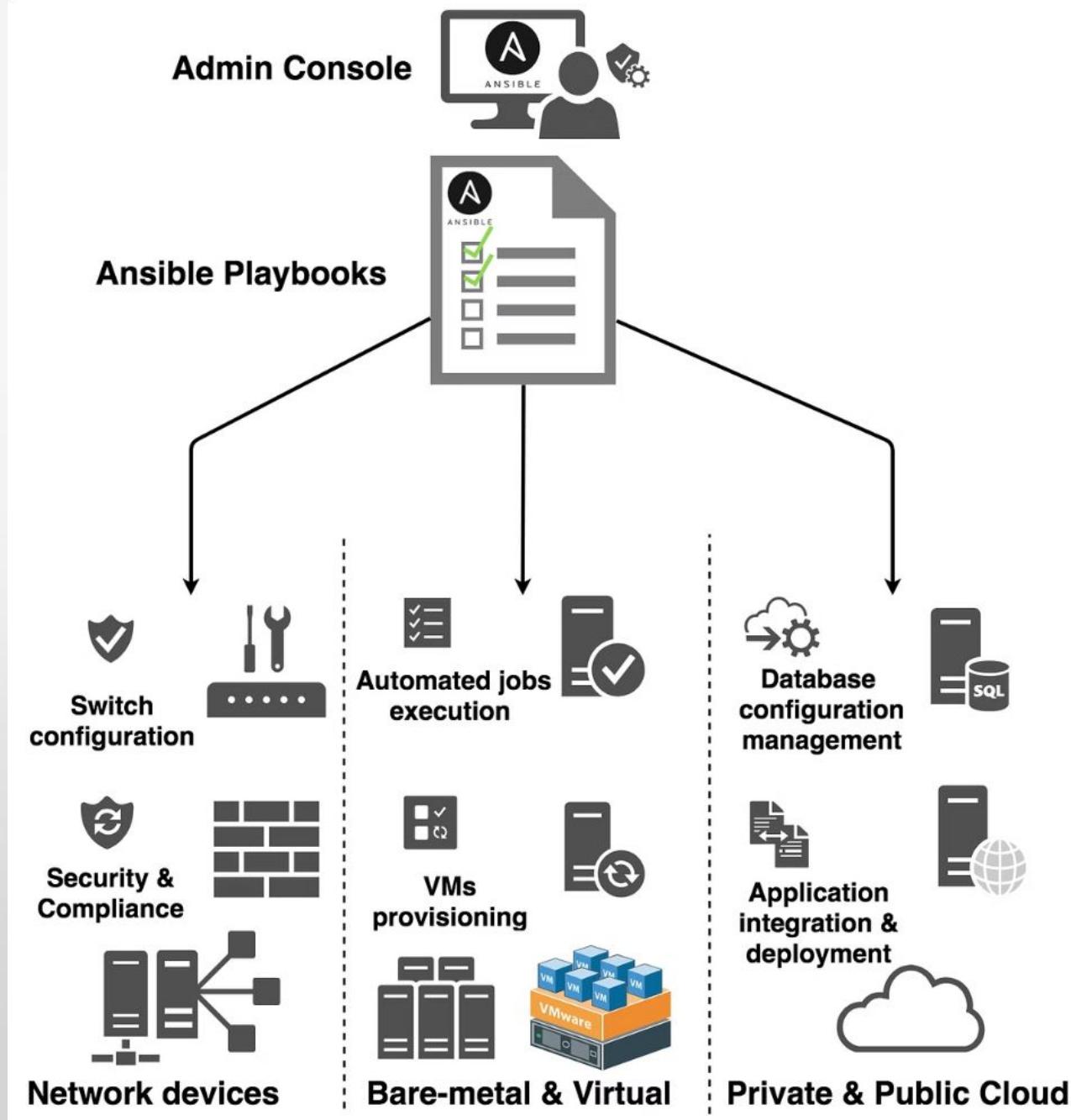
Escalabilidad

Permite administrar y aprovisionar múltiples sistemas de manera simultánea a escala.

Simplificación

Su sintaxis sencilla y legible facilita la adopción y el uso por parte de los equipos.

Casos de uso de Ansible en diferentes industrias



Infraestructura de TI

Aprovisionar y gestionar servidores, redes, almacenamiento y nubes.

Desarrollo de Software

Automatizar la construcción, el despliegue y la configuración de aplicaciones.

Seguridad

Aplicar configuraciones de seguridad y solucionar vulnerabilidades de forma automatizada.

DevOps

Integrar y orquestar los diferentes componentes de la cadena de entrega de software.

Conceptos clave de Ansible

Playbooks

Archivos YAML que definen las tareas y la lógica de automatización.

Módulos

Componentes que encapsulan la lógica para interactuar con diferentes recursos.

Inventarios

Definición de los sistemas administrados, con sus características y grupos.

```
---
- name: install and start apache
  hosts: webservers
  user: root

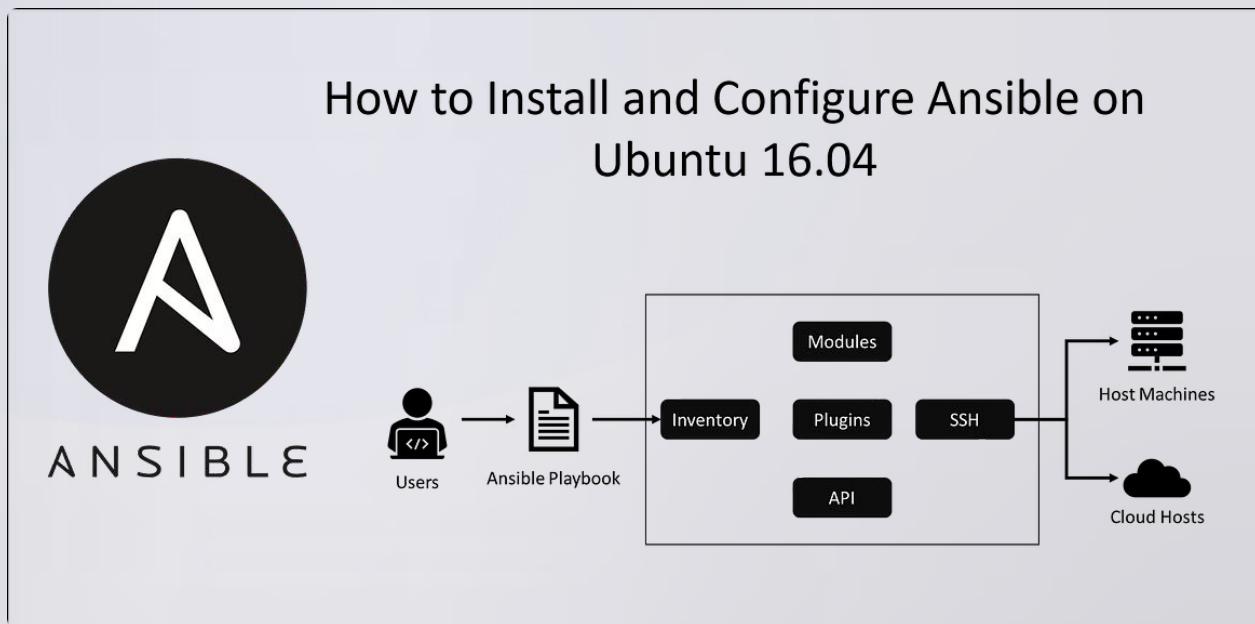
  tasks:
    - name: install httpd
      yum: name=httpd state=latest
    - name: start httpd
      service: name=httpd state=running
```

Playbook

Play

Tasks

Configuración e implementación de Ansible



- 1 **Instalación**
Ansible se instala fácilmente en sistemas operativos como Linux y macOS.
- 2 **Configuración**
Se define el inventario, las credenciales y las variables de entorno.
- 3 **Implementación**
Los usuarios crean playbooks y los ejecutan para automatizar sus procesos.

Gestión de configuraciones con Ansible

1 Idempotencia

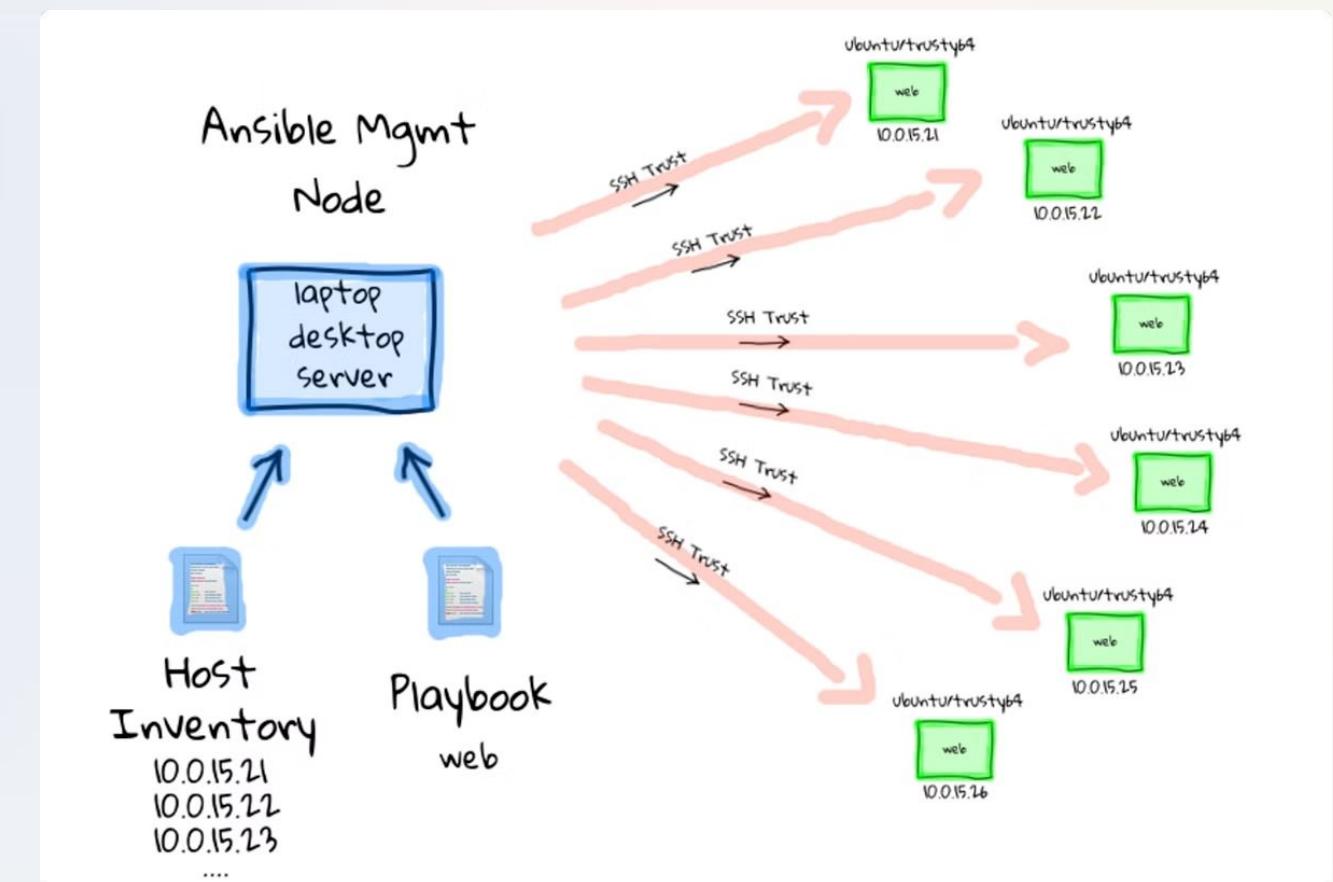
Ansible garantiza que las tareas se ejecuten solo cuando sea necesario.

2 Configuración como Código

Los playbooks permiten versionar y mantener la configuración de los sistemas.

3 Registros y Reportes

Ansible proporciona una visibilidad detallada de las acciones realizadas.



Orquestación de infraestructura con Ansible



Aprovisionamiento

Creación de máquinas virtuales, contenedores y otros recursos de infraestructura.



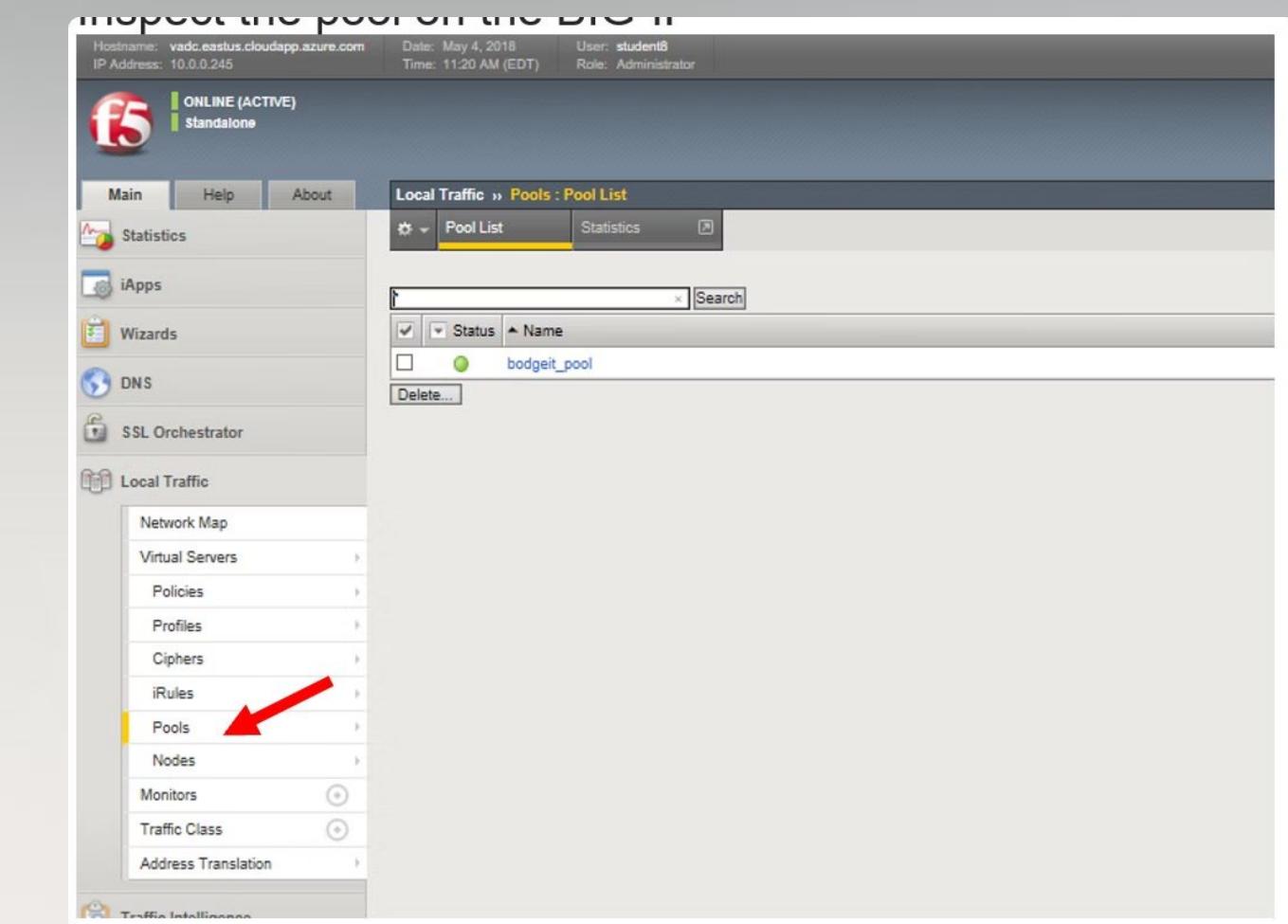
Configuración

Aplicación de configuraciones específicas a los recursos recién aprovisionados.

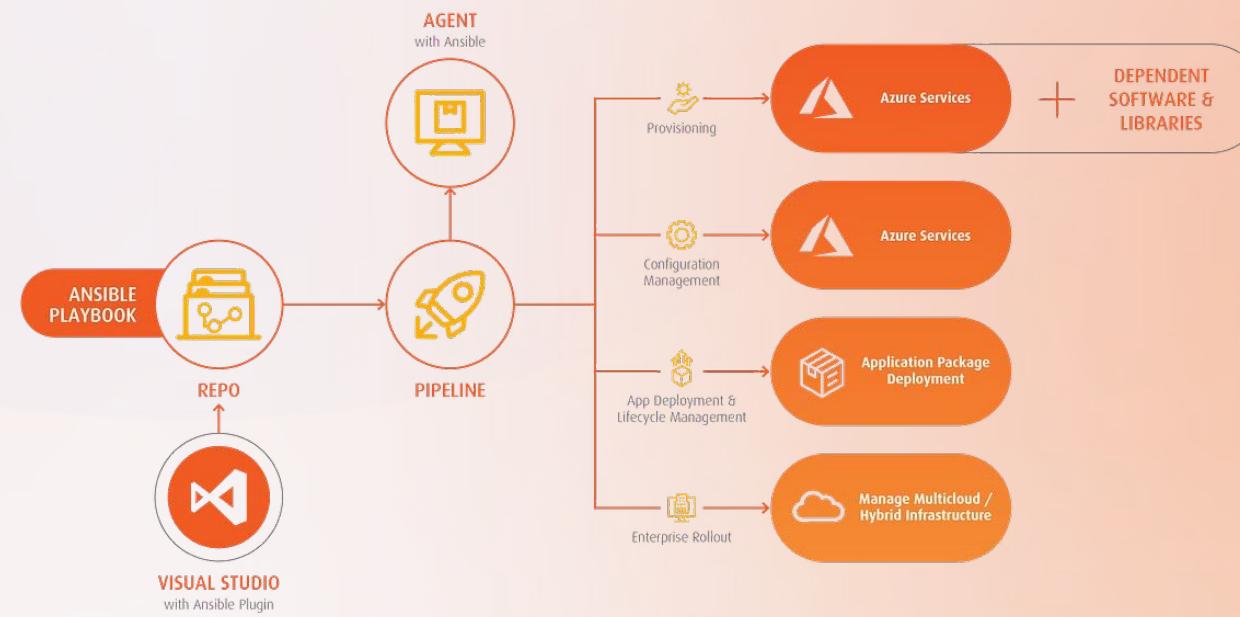


Integración

Conexión e interacción entre los diferentes componentes de la infraestructura.



Integración de Ansible con otras herramientas



Git

Control de versiones para los playbooks y la configuración.



Nubes

Aprovisionamiento y gestión de recursos en diferentes proveedores de nube.



Monitoreo

Integración con herramientas de monitoreo para obtener visibilidad de la infraestructura.



Nombre de la actividad:	Evaluación de rendimiento
Cantidad de participantes:	38
Doy fe que esta actividad está planificada en dtt (Sí/No):	Si

Hora de inicio:	15:53
Hora de fin:	16:03
Duración (min):	10 min

Participantes: llenar las siguientes cajas de texto (tomar información del chat del meet)

BRIAN JOSUE ERAZO SAGASTUME
15:52
Brian Erazo - 201807253
LUISA MARÍA ORTÍZ ROMERO
15:52
Luisa Ortiz - 202003381
ANDRÉ JOAQUIN ORTEGA DE PAZ
15:52
Joaquin Ortega - 201900597
MARCO ALEXANDER DE LEÓN HERNÁNDEZ
15:52
Marco de León - 202010014
ANGEL EDUARDO MARROQUÍN CANIZALES
15:52
Angel Marroquín - 202003959
ANGEL MIGUEL GARCÍA URIZAR
15:52
Angel García - 201901421
JOSÉ ANDRÉS LÓPEZ PINEDA
15:52
Andrés López - 201901756
LUIS FERNANDO SANCHEZ SOTO
15:52
Luis Sánchez - 201700339
JOSÉ MANUEL IBARRA PIRIR
15:52
José Ibarra - 202001800
CARLOS AUGUSTO HERNÁNDEZ ORDOÑEZ
15:52
Carlos Hernández - 201611269
ZENAIDA IRAZEMA CHACÓN GARCIA
15:52
Zenaida Chacón - 201801460
BRAYAN GIOVANNY RIVAS ESTRADA
15:53
Brayan Rivas - 201325641

KEVIN ERNESTO GARCIA HERNANDEZ
15:53
Kevin García - 202113553
MARÍA CECILIA COTZAJAY LÓPEZ
15:53
María Cotzajay - 201602659
CESAR DANIEL MARTÍNEZ MENCOS
15:53
Cesar Martínez - 201904016
HUGO SEBASTIAN MARTÍNEZ HERNÁNDEZ
15:53
Hugo Martinez - 202002793
JAIME ALEJANDRO ARMIRA US
15:53
Jaime Armira - 201602983
DIEGO ENRIQUE ARRIAGA MELENDEZ
15:53
Diego Arriaga - 202003892
RAÚL JOSUE CASTILLO BARCO
15:53
Raúl Castillo - 202001932
DAVID EDUARDO LOPEZ MORALES
15:53
David López - 201907483
SAUL ESTEBAN CASTELLANOS UBEDA
15:56
Saul Castellanos - 201801178
ESTUARDO GABRIEL SON MUX
15:56
Estuardo Son - 202003894
DOUGLAS ALEXANDER SOCH CATALAN
15:56
Douglas Soch - 201807032

EDDY FERNANDO DIAZ GALINDO
15:56
Eddy Diaz - 201906558
EDGARDO ANDRÉS NIL GUZMÁN
15:56
Edgardo Nil - 201801119
CESAR ANDRE RAMIREZ DAVILA
15:56
César Ramírez - 202010816
JESER EMANUEL RODAS MEDINA
15:56
Jeser Rodas - 202105561
JORGE MARIO CANO BLANCO
15:56
Jorge Cano - 201902128
HARRY AARON GOMEZ SANIC
15:56
Harry Gómez - 202103718
XHUNIK NIKOL MIGUEL MUTZUTZ
15:56
Xhunik Miguel - 201900462
RONY ORMANDY ORTÍZ ALVAREZ
15:57
Rony Ortiz - 201807328
JOSÉ MANUEL SOLIS ORTIZ
15:57
José Solis - 201800517
ALVARO NORBERTO GARCIA MEZA
15:57
Alvaro Garcia - 202109567
ANTHONY SAMUEL ZEA HERRERA
15:58
Anthony Zea - 202104782
CARLOS EDUARDO SOTO MARROQUÍN
15:58
Carlos Soto - 201902502

ALBERTO GABRIEL REYES NING
15:58
Alberto Reyes - 201612174
BRAYAN ALEXANDER MEJIA BARRIENTOS
15:58
Brayan Mejia 201900576