A decorative graphic on a light blue background. On the left, four white paper airplanes are positioned at different heights, each with a vertical white line extending downwards. A white line starts from the bottom left, goes up and to the right, then turns left and continues upwards, ending at a red paper airplane. The red airplane is positioned to the left of the title text.

ANSIBLE Y VAGRANT PARA APROVISIONAR MÁQUINAS EN LA NUBE (AWS)

*Planificación y Gestión de
Infraestructuras TIC*

MIGUEL DE LA CAL BRAVO

FÉLIX ÁNGEL MARTÍNEZ MUELA

ÍNDICE

- Introducción
- Puesta en marcha
- Tutorial
- Otras posibilidades
- Ventajas/Inconvenientes
- Conclusión y opinión personal
- Referencias

INTRODUCCIÓN

- Crear, aprovisionar y configurar máquinas virtuales...
- **Suministrador de servicios en la nube: Amazon Web Services (AWS)**
- Herramientas de automatización: Vagrant, Ansible

PUESTA EN MARCHA

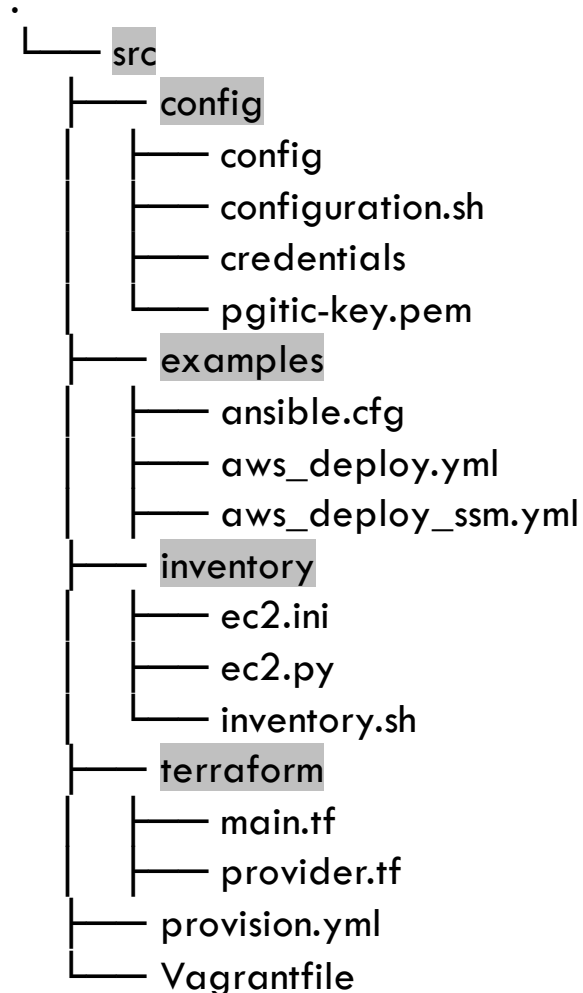
- Ansible
- Vagrant
- VirtualBox



- Paquetes y plugins
- Cuenta en AWS, par de claves EC2 y AWS, grupo de seguridad, etc

PUESTA EN MARCHA – ESTRUCTURA DE DIRECTORIOS

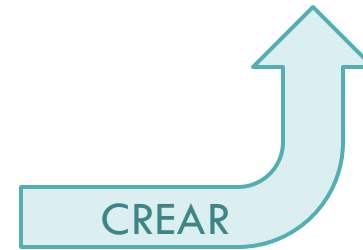
■ Directorios:



- Lanzaremos todos los comandos desde el directorio /src
- /config: ficheros de configuración y clave PEM
- /examples: ejemplos de playbooks de Ansible para iniciar instancias EC2 de AWS
- /inventory: configuración de inventarios dinámicos
- /terraform: ejemplo con Terraform
- provision.yml: playbook de aprovisionamiento para Vagrant.
- Vagrantfile: fichero de configuración de Vagrant

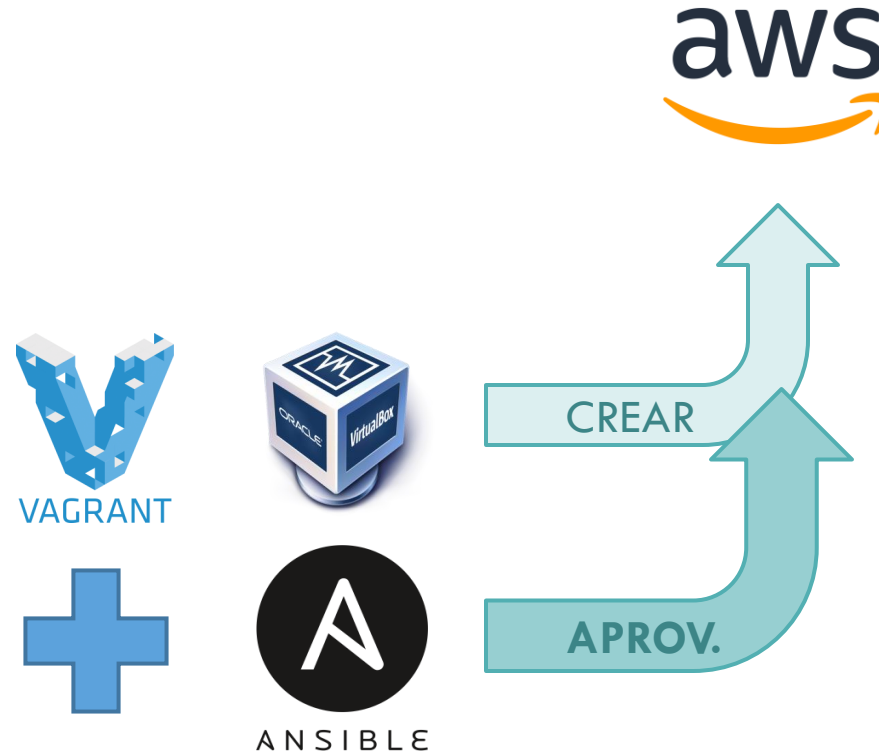
TUTORIAL 1 – CREAR UNA INSTANCIA EC2 CON VAGRANT

- Vagrantfile
- Box "dummy"
- `$ vagrant up --provider=aws`



TUTORIAL 2 – CREAR Y APROVISIONAR UNA INSTANCIA EC2 CON ANSIBLE Y VAGRANT

- Vagrantfile
- **Playbook de Ansible: provision.yml**
 - Actualizar caché apt
 - Instalar Apache
 - Verificar instalación de Apache
- `$ vagrant up --provider=aws`



TUTORIAL 3 – PLAYBOOKS DE ANSIBLE PARA CREAR Y APROVISIONAR MÁQUINAS EN AWS

- Playbook de Ansible: `aws_deploy.yml`
 - **Crear y añadir instancia EC2 al grupo de hosts**
 - **Esperar conexión SSH**
 - Actualizar caché apt
 - Instalar Apache
 - Verificar instalación de Apache



- `$ ansible-playbook ./examples/aws_deploy.yml`
`--private-key=./config/pgitic-key.pem --ssh-common-args '-o StrictHostKeyChecking=no'`

OTRAS POSIBILIDADES – INVENTARIOS DINÁMICOS Y COMANDOS AD-HOC

- Paquete pip ansible
- **Inventario de hosts dinámico = Script python (ec2.py)**
- Listar instancias levantadas:
 - `$./inventory/ec2.py --list`
- Módulo ping:
 - `$ ansible all -i ./inventory/ec2.py -u ubuntu -m ping --private-key ./config/pgitic-key.pem`
- Módulo command:
 - `$ ansible all -i ./inventory/ec2.py -u ubuntu -m command -a "id" --private-key ./config/pgitic-key.pem`

OTRAS POSIBILIDADES – AWS SYSTEMS MANAGER + ANSIBLE

- Crear asociación
 - Nombre: Ansible_Install_Apache
 - Tipo documento: AWS-RunAnsiblePlaybook
 - Playbook: provision.yml
 - Targets: instancias EC2 que queramos
 - Schedule, guardar salida en contenedor Amazon S3...



Name - optional
Provide a name for your Association.

Ansible_Install_Apache

Document


Document

AWS-RunAnsiblePlaybook

Document description

Use this document to run Ansible playbooks on Amazon EC2 managed instances. Specify either YAML text or URL. If you specify both, the URL parameter will be used. Use the extrav parameter to send runtime variables to the Ansible execution. Use the check parameter to perform a dry run of the Ansible execution. The output of the dry run shows the changes that will be made when the playbook is executed.

Use the service-linked role [AWSServiceRoleForAmazonSSM](#) to allow State Manager to manage AWS resources on your behalf.

Q AWS-RunAnsiblePlaybook X < 1 >			
Document name prefix: Equals: AWS-RunAnsiblePlaybook X		Clear filters	
Name	Owner	Platform types	Document type
 AWS-RunAnsiblePlaybook	Amazon	Linux	Command

VENTAJAS E INCONVENIENTES

✓ Ventajas

- Ansible es muy potente para crear y aprovisionar máquinas por sí sola
 - Inventarios dinámicos
 - Comandos ad-hoc + módulos AWS
- AWS Systems Manager



■ Inconvenientes

- Vagrant NO es buena opción para gestionar máquinas en AWS
 - Se hace uso de un box dummy y un plugin "desactualizado"
 - En su lugar, nosotros usaríamos Terraform



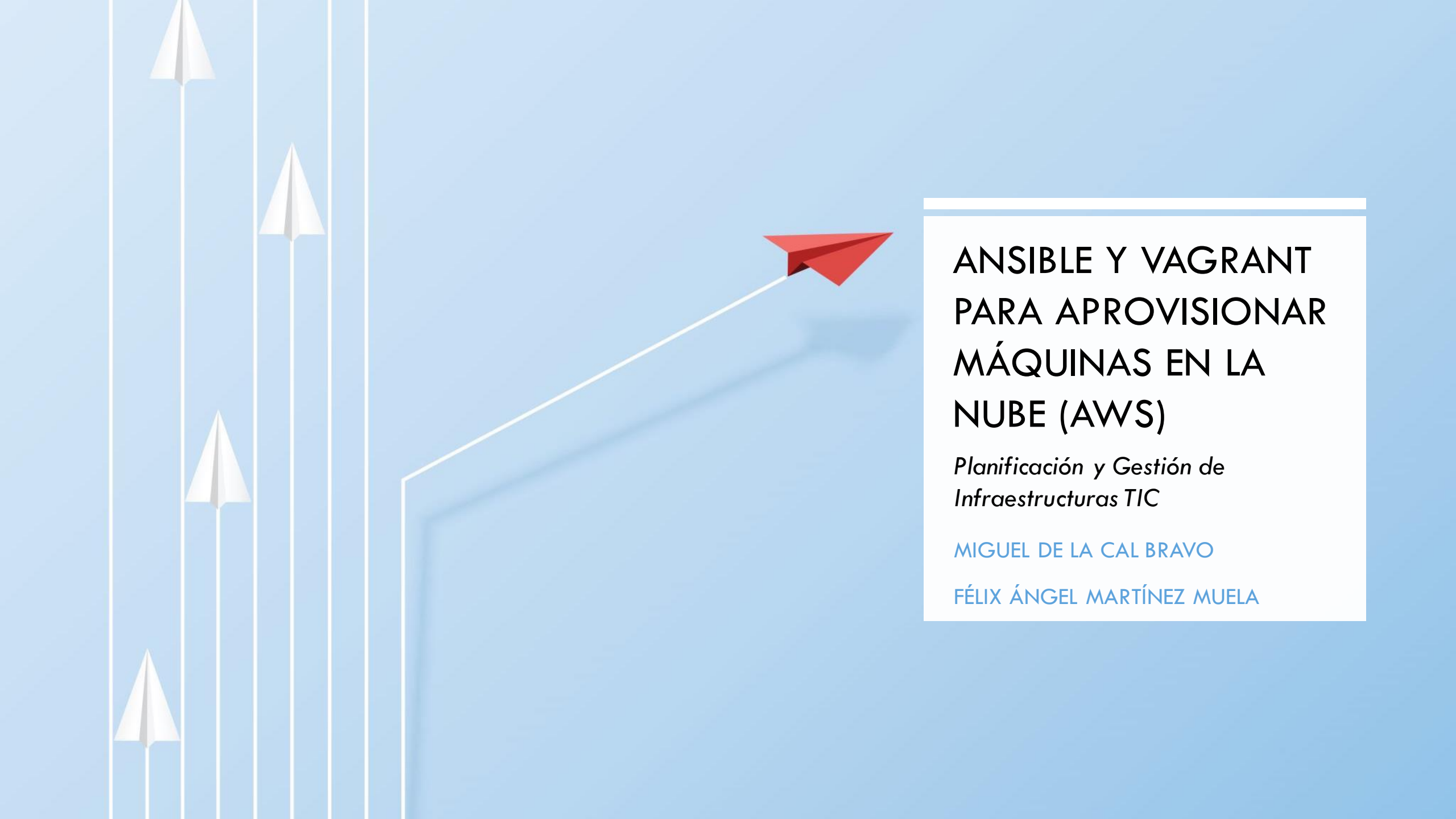
CONCLUSIONES Y OPINIÓN PERSONAL

- Vagrant y VirtualBox => Crear y aprovisionar máquinas en local
- Ansible => Crear y aprovisionar máquinas en la nube (AWS)
- AWS Systems Manager es un servicio interesante
- Terraform como alternativa a Vagrant y VirtualBox



REFERENCIAS

- Repositorio plugin Vagrant AWS: <https://github.com/mitchellh/vagrant-aws>
- Tutorial Vagrant levantar instancias EC2 en AWS: <https://openwebinars.net/blog/vagrant-sobre-aws-amazon-videotutorial/>
- Inventarios dinámicos en Ansible para AWS:
https://docs.ansible.com/ansible/latest/user_guide/intro_dynamic_inventory.html#inventory-script-example-aws-ec2
- Playbooks de Ansible desde AWS Systems Manager: <https://docs.aws.amazon.com/systems-manager/latest/userguide/systems-manager-state-manager-ansible.html>
- Terraform y Ansible para aprovisionar máquinas EC2 en AWS: <https://alex.dzyoba.com/blog/terraform-ansible/>
- Documentación de AWS: <https://docs.aws.amazon.com/>
- Documentación de Vagrant: <https://www.vagrantup.com/docs/>
- Documentación de Ansible: <https://docs.ansible.com/>
- Documentación de Terraform: <https://www.terraform.io/>
- Bash scripting cheatsheet: <https://devhints.io/bash>

The background is a solid light blue. On the left side, there are four white paper airplanes of varying heights, each with a thin white vertical line extending from its base to the bottom of the frame. A white line starts from the bottom left, goes up and to the right, then turns left and goes up to a red paper airplane. The red airplane is positioned above the title text.

ANSIBLE Y VAGRANT PARA APROVISIONAR MÁQUINAS EN LA NUBE (AWS)

*Planificación y Gestión de
Infraestructuras TIC*

MIGUEL DE LA CAL BRAVO

FÉLIX ÁNGEL MARTÍNEZ MUELA