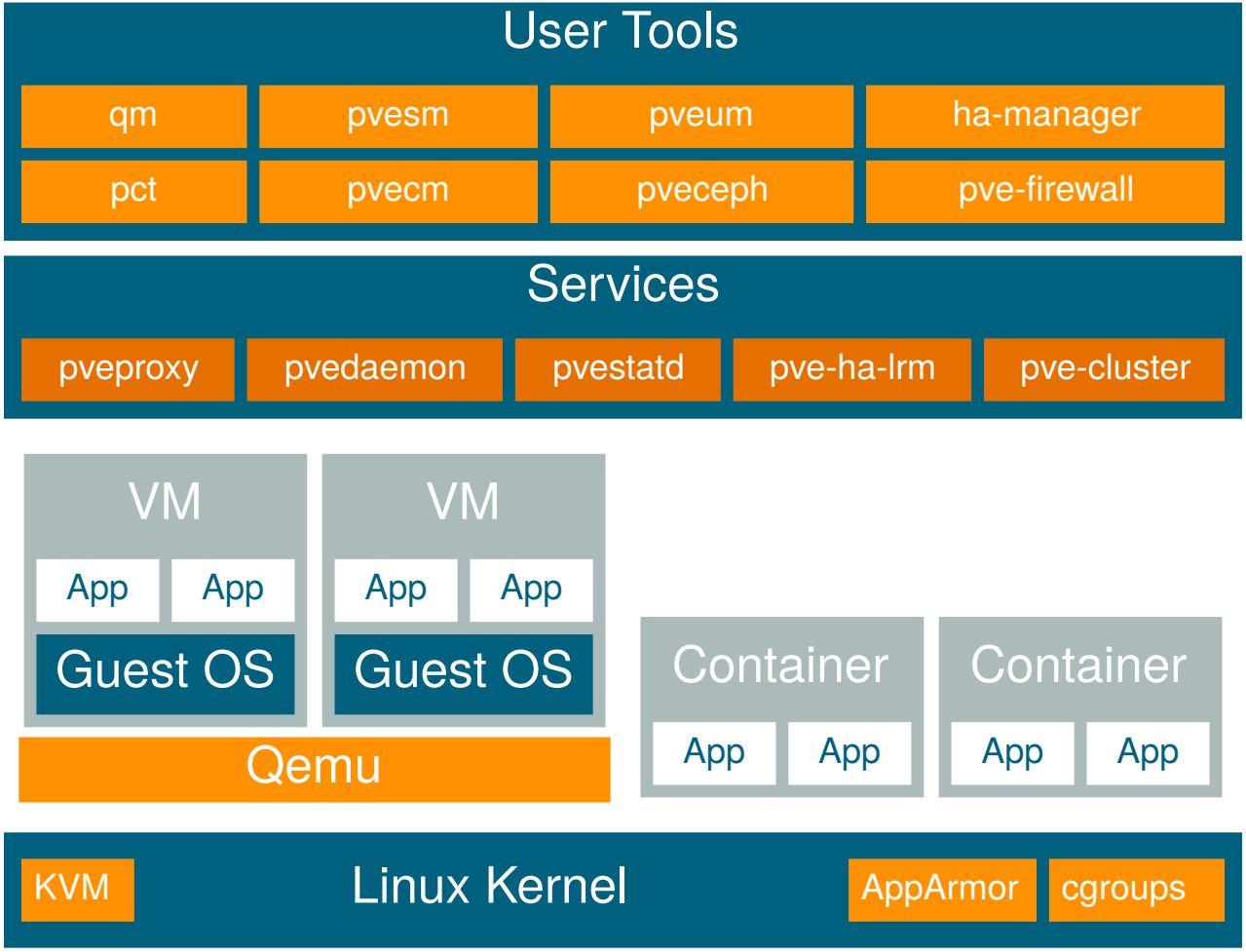
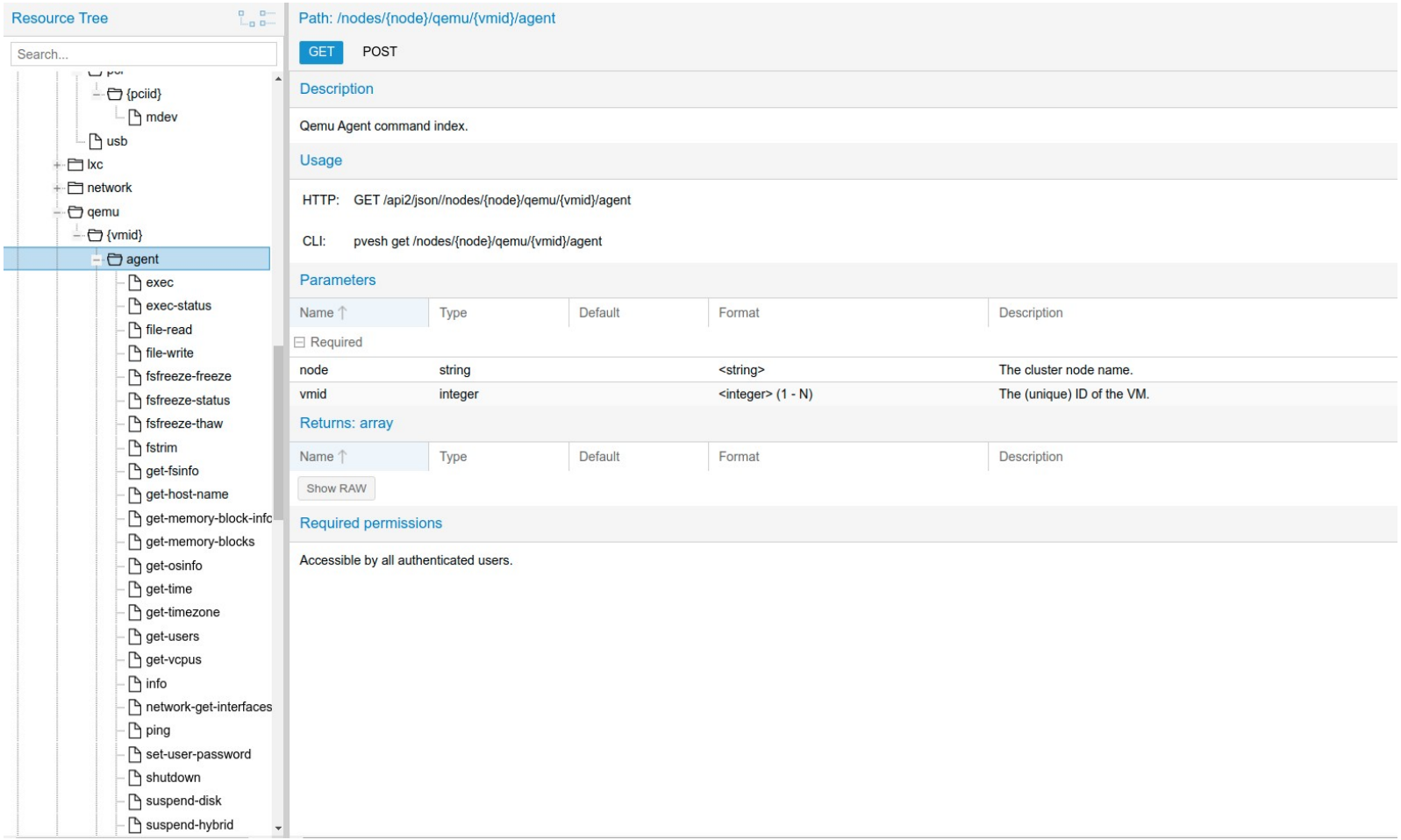
PROXMOX: ESCALABILIZANDO DE FORMA FÁCIL COM PACKER, ANSIBLE E TERRAFORM


PROXMOX


User Tools
qm
pvesm
pveum
ha-manager
pct
pvecm
pveceph
pve-firewall
Services
pveproxy
pvedaemon
pvestatd
pve-ha-lrm
pve-cluster
VM
App
App
Guest OS
VM
App
App
Guest OS
Qemu
Container
App
App
Container
App
App
KVM
Linux Kernel
AppArmor
cgroups


VM
VM


Resource Tree
Search...
{pciid}
mdev
usb
lxc
network
qemu
{vmid}
agent
exec
exec-status
file-read
file-write
fsfreeze-freeze
fsfreeze-status
fsfreeze-thaw
fstrim
get-fsinfo
get-host-name
get-memory-block-info
get-memory-blocks
get-osinfo
get-time
get-timezone
get-users
get-vcpus
info
network-get-interfaces
ping
set-user-password
shutdown
suspend-disk
suspend-hybrid
Path: /nodes/{node}/qemu/{vmid}/agent
GET
POST
Description
Qemu Agent command index.
Usage
HTTP: GET /api2/json//nodes/{node}/qemu/{vmid}/agent
CLI: pvesh get /nodes/{node}/qemu/{vmid}/agent
Parameters
Name
Type
Default
Format
Description
Required
node
string
<string>
The cluster node name.
vmid
integer
<integer> (1 - N)
The (unique) ID of the VM.
Returns: array
Name
Type
Default
Format
Description
Show RAW
Required permissions
Accessible by all authenticated users.

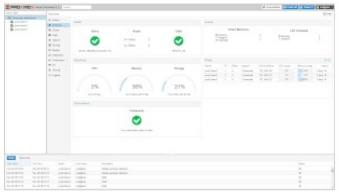
https://www.proxmox.com/en/

PROXMOX

Home | Virtualization | Backup | Email Security | Downloads | Training | Partners | News | About us | search...

Proxmox VE 7.1

The latest version of the open-source virtualization platform is available.



Get Subscription | Learn more

Proxmox Virtual Environment

Proxmox VE is a complete open-source platform for enterprise virtualization. With the built-in web interface you can easily manage VMs and containers, software-defined storage and networking, high-availability clustering, and multiple out-of-the-box tools on a single solution.

About Proxmox VE

Proxmox Backup Server

Proxmox Backup Server is an enterprise backup solution for backing up and restoring VMs, containers, and physical hosts. The open-source solution supports incremental backups, deduplication, Zstandard compression, and authenticated encryption.

About Proxmox Backup Server

Proxmox Mail Gateway

Proxmox Mail Gateway is an open-source email security solution protecting your mail server against all email threats from the moment they emerge. The full featured mail proxy can be easily deployed between the firewall and your internal mail server in just a few minutes.

About Proxmox Mail Gateway

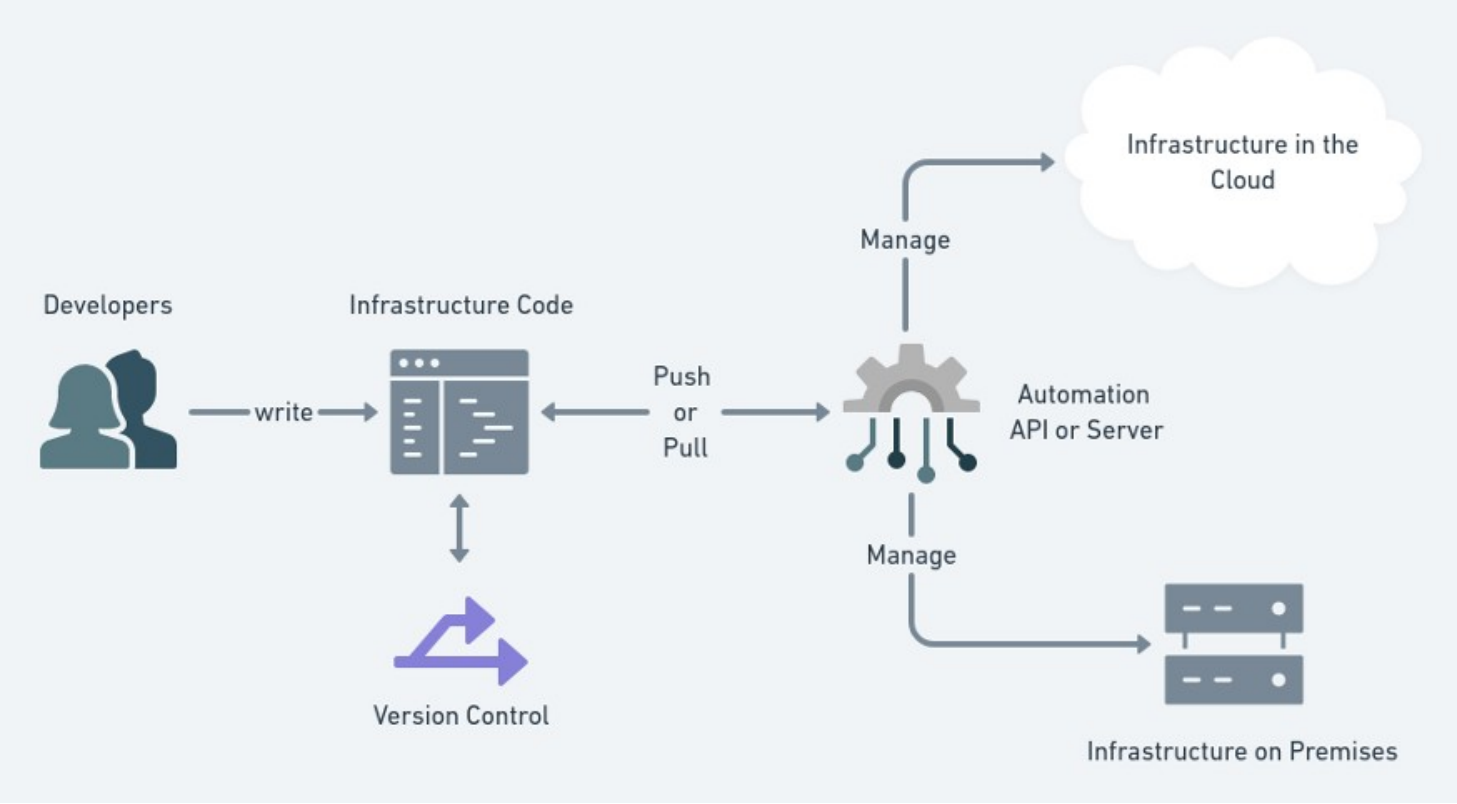
Infraestrutura como código

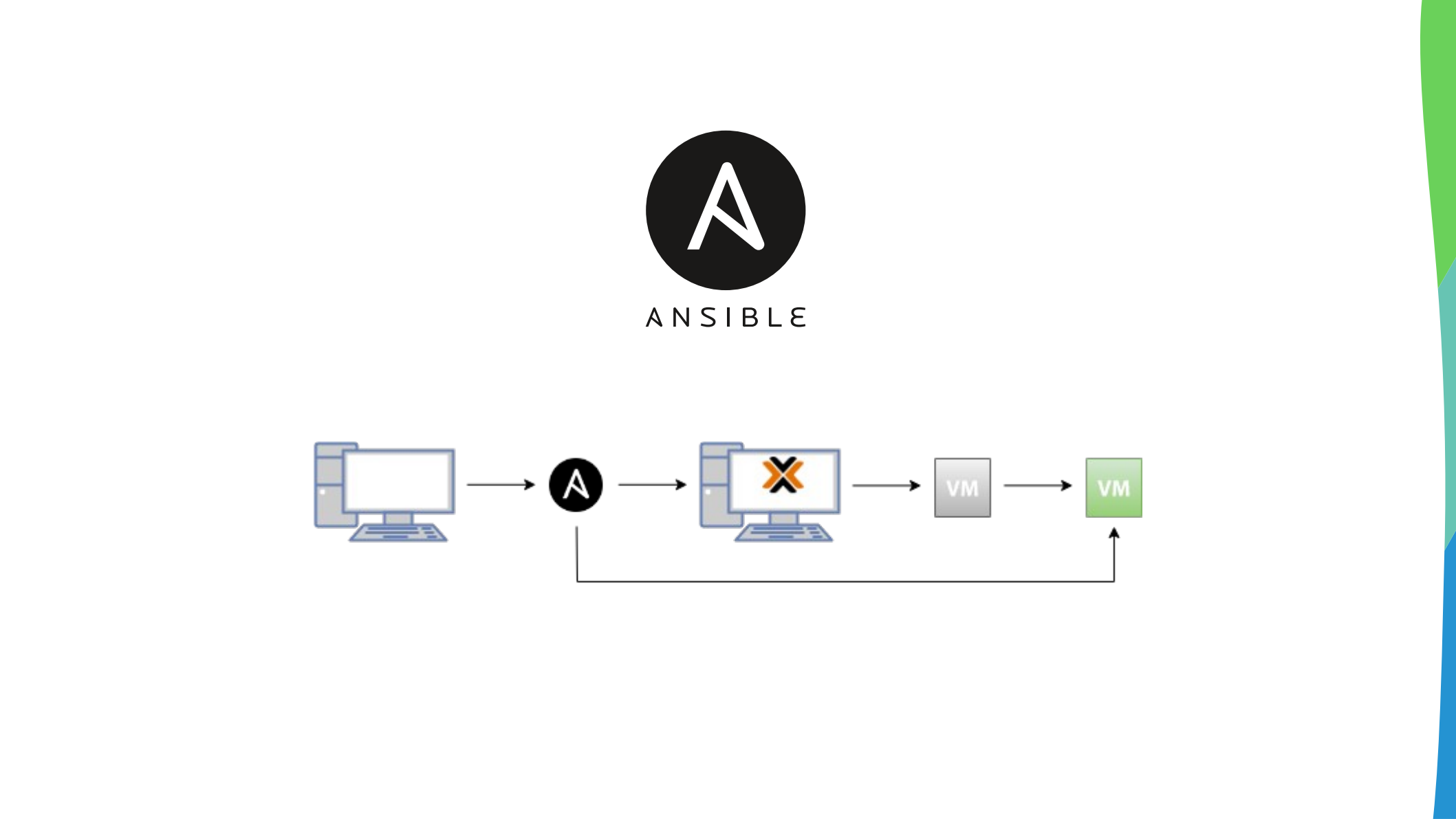
ou

Infrastructure as code

ou

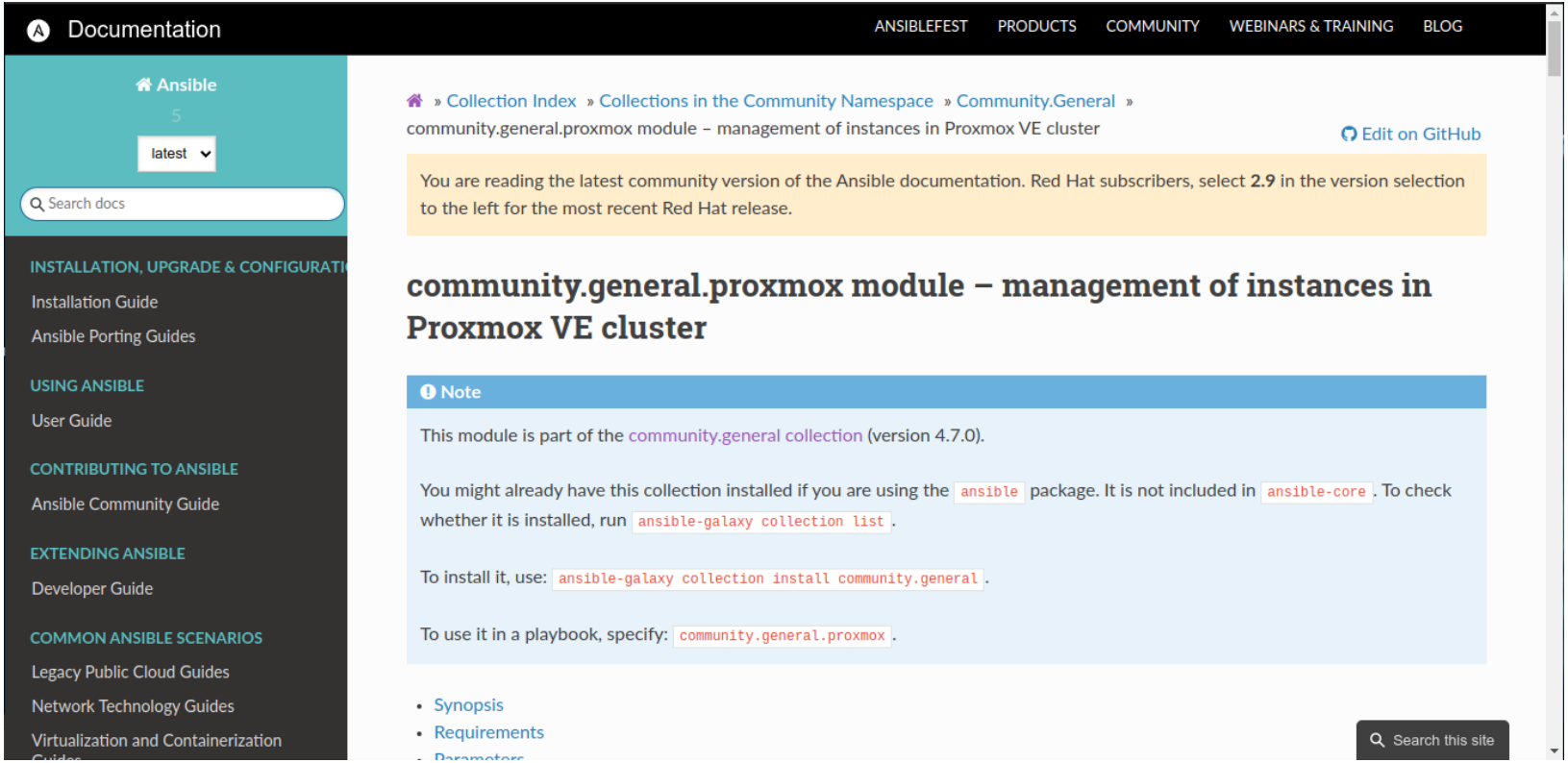
IaC

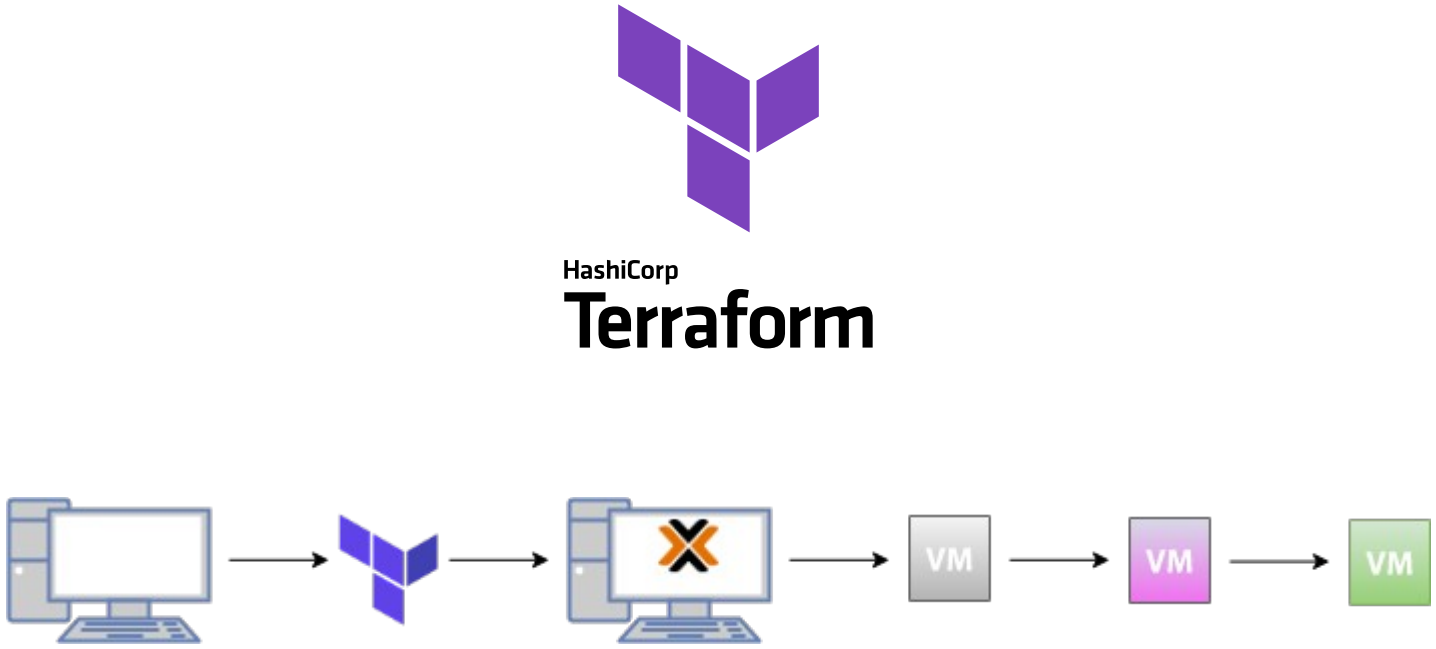

Infrastructure in the Cloud
Manage
Developers
Infrastructure Code
write
Push
or
Pull
Automation
API or Server
Manage
Version Control
Infrastructure on Premises


ANSIBLE
VM
VM

Documentação Ansible

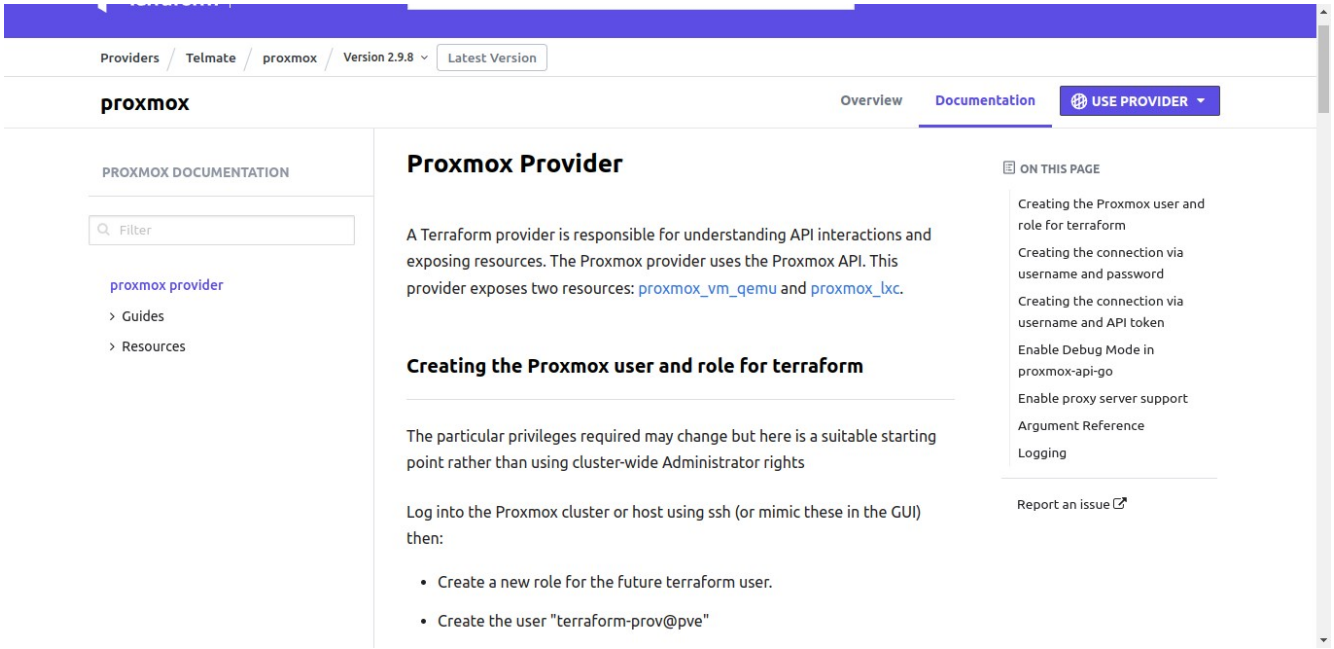
https://docs.ansible.com/ansible/latest/collections/community/general/proxmox_module.html




HashiCorp
Terraform
VM
VM
VM

Documentação Terraform

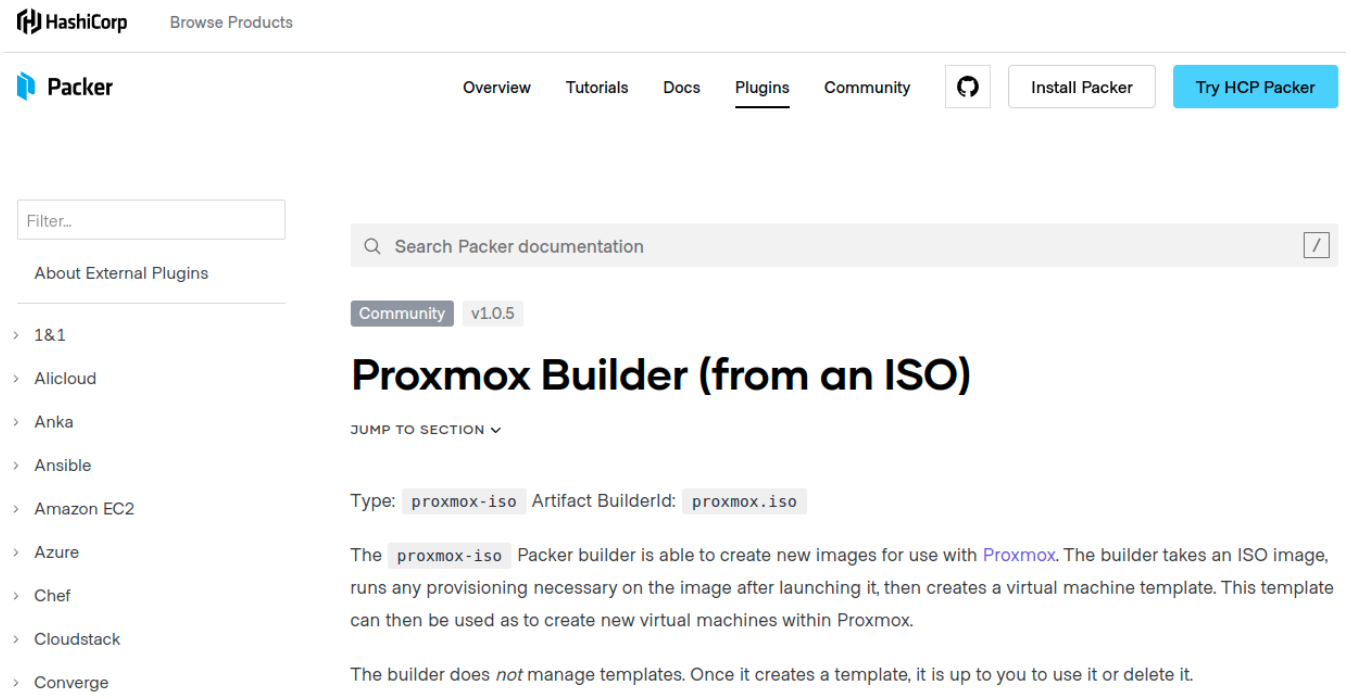
https://registry.terraform.io/namespaces/Telmate

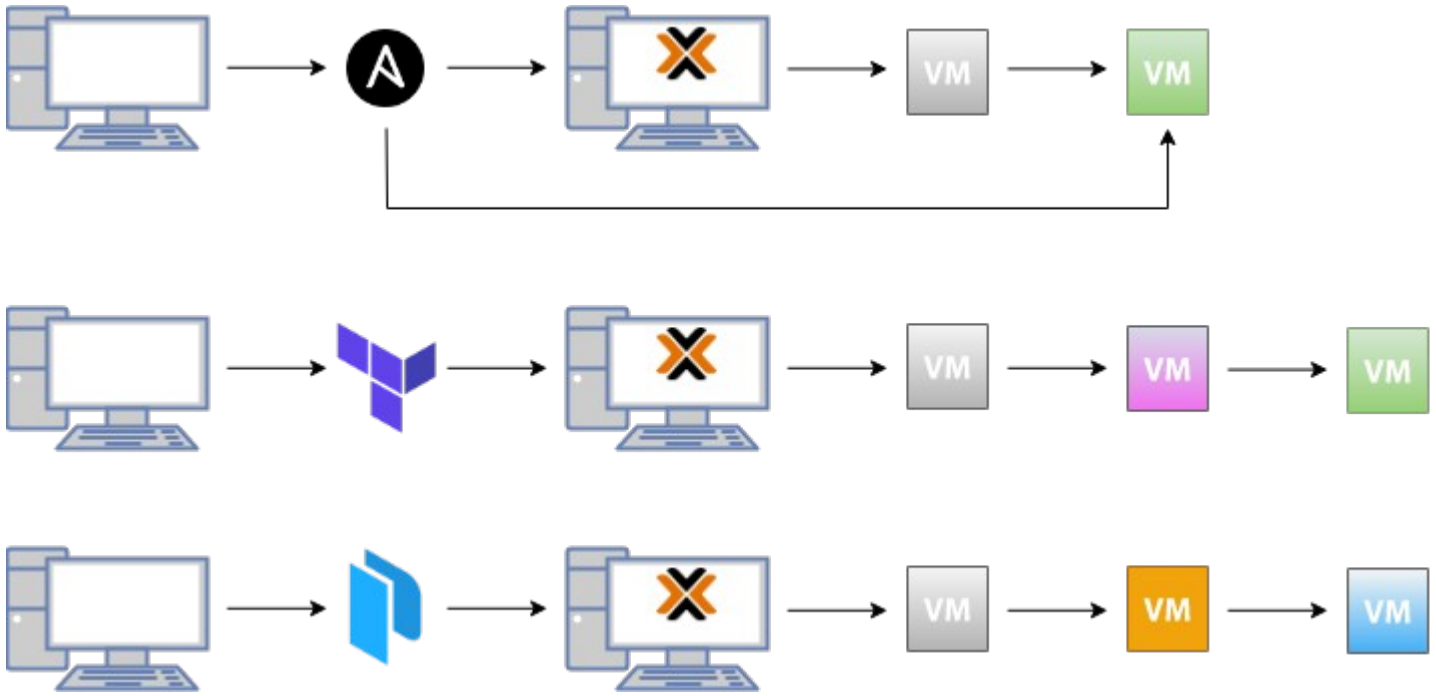


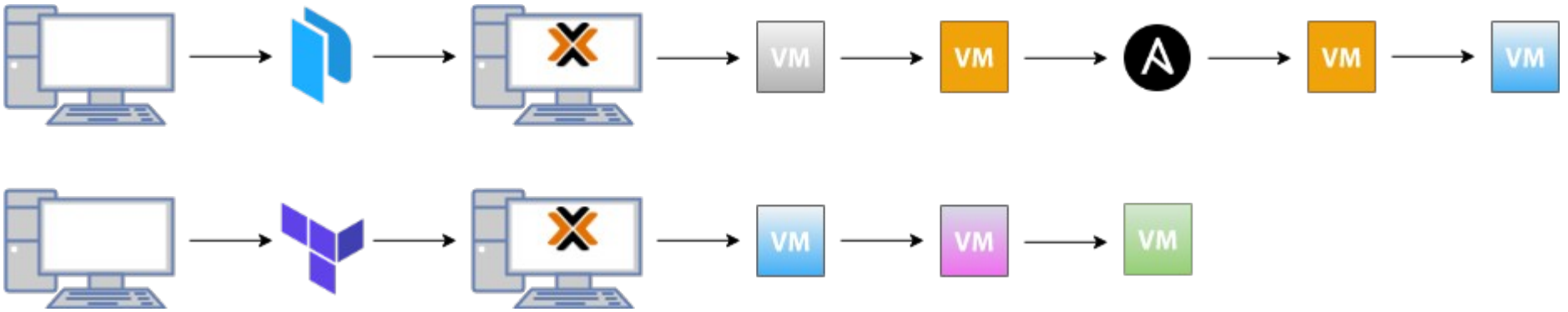

VM
VM
VM

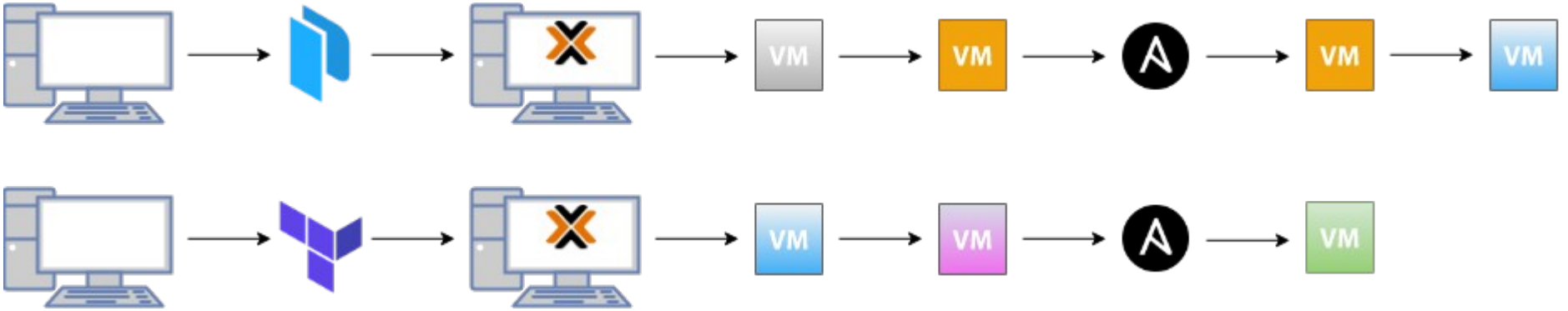
Documentação Packer

https://www.packer.io/plugins/builders/proxmox/clone

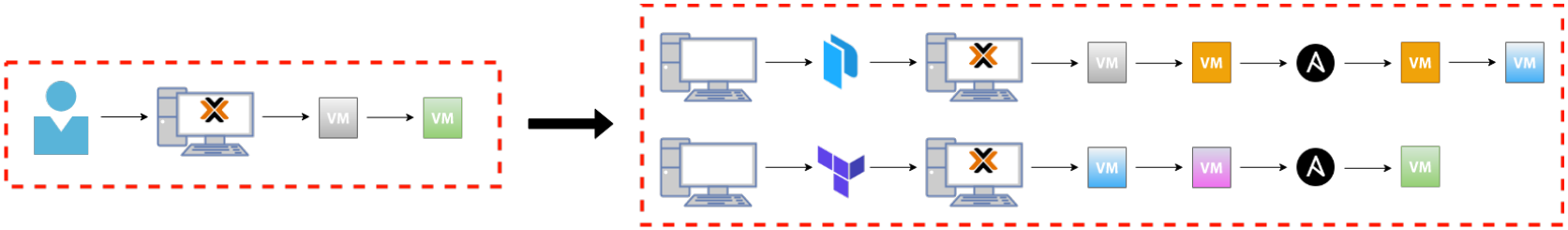



VM
VM
VM
VM
VM
VM


VM
VM
VM
VM
VM
VM

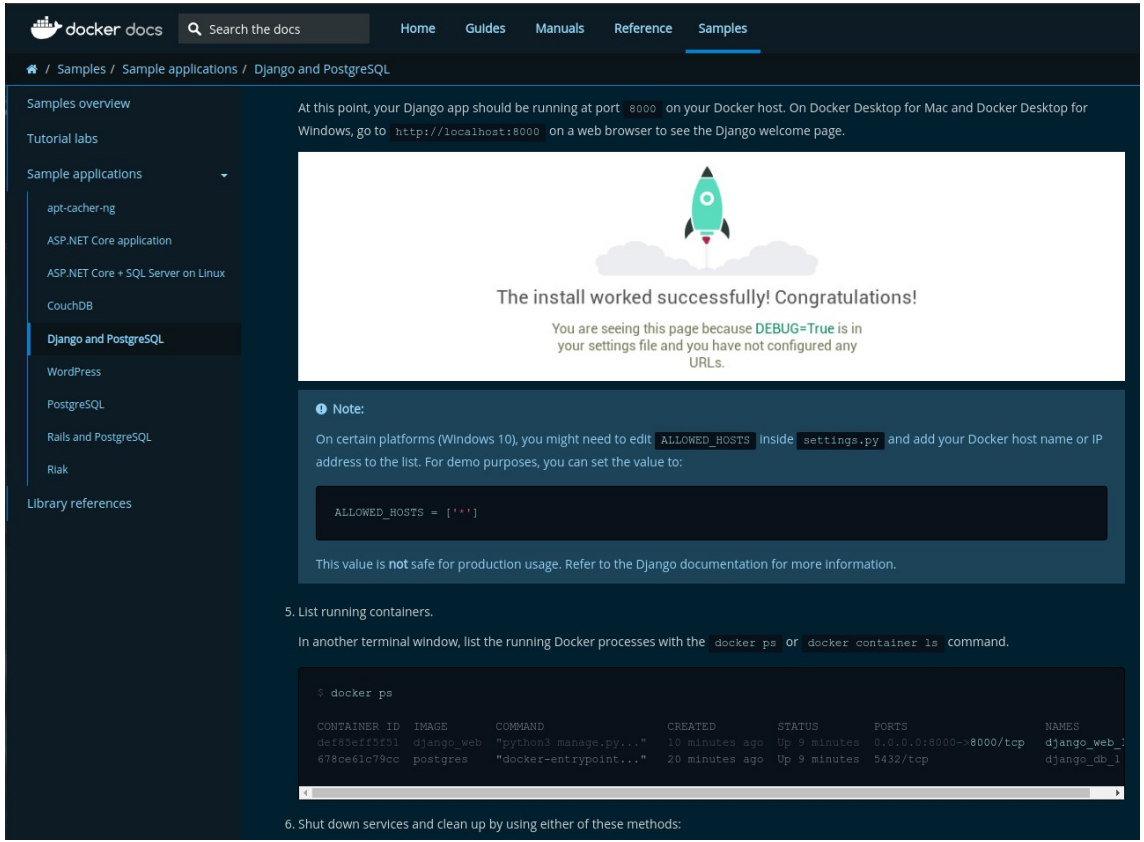

VM
VM
VM
VM
VM
VM
VM

OBJETIVO



Aplicação de teste

https://docs.docker.com/samples/django/



Dependências

Build:
- virt-customize (libguestfs-tools)
- vm exclusiva para customização de imagens “.img”
- Ubuntu Cloud Images: focal-server-cloudimg-amd64.img
- qemu-guest-agent

Deploy:
- Transferir a imagem customizada para o servidor Proxmox
- “subir” a imagem para o Proxmox via comando “qm”

Virt-customize

Instalação:
sudo apt update -y && sudo apt install libguestfs-tools -y

Execução:
sudo virt-customize \
--add focal-server-cloudimg-amd64.img \
--run-command 'sed -i "68 s/^ */#/" /etc/cloud/cloud.cfg' \
--install qemu-guest-agent

qm - Qemu/KVM Virtual Machine Manager

```
qm create 9000 --name ub20.04-flisol-2022-amd64 --memory 1024 --net0 virtio,bridge=vmbr0
qm importdisk 9000 focal-server-cloudimg-amd64.img local-lvm
qm set 9000 --scsihw virtio-scsi-pci --scsi0 local-lvm:vm-9000-disk-0
qm set 9000 --ide2 local-lvm:cloudinit
qm set 9000 --boot c --bootdisk scsi0
qm set 9000 --serial0 socket --vga serial0
qm set 9000 --agent 1
qm set 9000 --ipconfig0 ip=dhcp
qm resize 9000 scsi0 8G
qm template 9000
```

comandos-sccript.sh+

```
#!/usr/bin/env bash

echo -n "nome da imagem:"
read IMAGEM

echo -n "name da vm resultante:"
read RESULTVM

echo -n "id da vm:"
read IDVM

# create a new VM
qm create "$IDVM" --name "$RESULTVM" --memory 1024 --net0 virtio,bridge=vmbr0

# import the downloaded disk to local-lvm storage
qm importdisk "$IDVM" "$IMAGEM" local-lvm

# finally attach the new disk to the VM as scsi drive
qm set "$IDVM" --scsihw virtio-scsi-single --scsi0 local-lvm:vm-"$IDVM"-disk-0

# create disk for use cloudinit
qm set "$IDVM" --ide2 local-lvm:cloudinit

## Set up a boot disk
qm set "$IDVM" --boot c --bootdisk scsi0

# Set up serial socket
qm set "$IDVM" --serial0 socket --vga serial0

# Set up qemu-guest-agent
qm set "$IDVM" --agent 1

# Set up dhcp
qm set "$IDVM" --ipconfig0 ip=dhcp

# Resize volume
qm resize "$IDVM" scsi0 8G

# Create a template from vm
qm template "$IDVM"
```