MAAS "METAL AS A SERVICE"



FRANCISCO JAVIER CRUCES DOVAL

PROYECTO INTEGRADO

Administración de Sistemas Informaticos en Red

Junio 2024 IES Gonzalo Nazareno

Índice

1.Descripción del proyecto	4
1.1 Tecnologías que se van a utilizar	4
1.2 Resultados que se esperan obtener	5
1.2.1 Automatización del Aprovisionamiento	5
1.2.2 Personalización mediante Cloud-init	5
1.2.3 Mantenimiento de un Inventario Actualizado	5
2. Fundamentos teóricos y conceptos	6
2.1 Metal as a service.	6
2.2 PXE	7
2.3 TFTP	8
2.4 Cloud-init	9
2.5 Commissioning	10
2.6 Juiu	11
2.7 Ansible	12
3. Preparación del escenario	13
3.1 Creación de la maguina virtual en Proxmox	13
3.2 Configuración de red de la maguina virtual	14
3.3 Instalación de MAAS	15
3.3.1 Añadir repositorios	15
3.3.2 Creación de un usuario administrador.	15
3.3.3 Configuración inicial web	16
3.4 Configuración del servicio DHCP	20
4 Despliegue de imágenes de MAAS en Proxmox	<u>-</u> °
4.1 Introducción	
4 2 Creación de la maquina virtual en proxmox	22
4.3 Añadir la maquina a MAAS	22
4 4 Explorando la interfaz gráfica de MAAS	27
4 4 1 Network	27
4 4 2 Storage	28
4 4 3 PCI Devices	20
4 4 4 IISB	20
4.45 Commissioning	30
4.4.5 Commissioning	31
4 4 7 I ogs	32
1/18 Configuración	22
4.5 Commissioning	3/
4.5 Commissioning	36
4.0 Instalación de la instalación	38
5. Desplique de cluster de las con MAAS y Ansible	
5.1 Introducción	40 //
5.2 Croación de las maguinas en Dreymoy	40 //1
5.2 Creación de las inaquinas en Proxinox	41 //
5.4 Commissioning	4Z
5.5 Instalación porsonalizada con cloud init	44 1C
5.6 Instalación do k2s con Ansible	40 10
5.7 Desplique de la aplicación de enquestas	,49 בח

MAAS METAL AS A SERVICE

6.	Conclusiones	55
7.	Bibliografía	56

1.Descripción del proyecto

El propósito de este proyecto es simplificar y optimizar el despliegue de sistemas operativos en hardware real en entornos con una gran cantidad de máquinas físicas utilizando la herramienta MAAS. La cual nos ayuda a automatizar la configuración y el despliegue de servidores físicos, ofreciendo una solución eficiente para la gestión de infraestructuras a gran escala , reduciendo tanto el tiempo como los costos operativos.

1.1 Tecnologías que se van a utilizar

A lo largo del proyecto se utilizaran las siguientes tecnologías :

- Maquinas virtuales : Como sistema de virtualización utilizare KVM/QEMU y estará hospedado en Proxmox.
- Linux bridges : Utilizaremos para interconectar en la misma red las diferentes maquinas del escenario
- PXE : Para el arranque y la provisión de sistemas operativos en máquinas sin sistema operativo
- DHCP : Para la asignación automática de direcciones IP y otros parámetros de red.
- GitHub : Nos permitirá mantener controles de versiones de los ficheros cloud-init.
- Cloud-init : Sirve para realizar tareas como la configuración de red, la instalación de paquetes y la ejecución de scripts personalizados, asegurando un despliegue consistente y automatizado.
- Packer : Utilizaremos para generar imágenes personalizadas para añadirlas a MAAS.

1.2 Resultados que se esperan obtener

El objetivo principal del proyecto es automatizar el proceso de aprovisionamiento de sistemas operativos en hardware físico de manera similar a cómo se hace con máquinas virtuales. Esto implica la creación de un flujo de trabajo eficiente y automatizado que permita desplegar sistemas operativos en servidores físicos de forma rápida y consistente.

1.2.1 Automatización del Aprovisionamiento

La automatización del aprovisionamiento garantizará que los sistemas operativos se instalen de manera coherente y sin intervención manual. Esto se logrará mediante el uso de herramientas como MAAS (Metal as a Service), que permiten la gestión remota y centralizada de los recursos de hardware.

1.2.2 Personalización mediante Cloud-init

El uso de Cloud-init permitirá la personalización de las instalaciones de sistemas operativos de acuerdo con requisitos específicos. Cloud-init es una herramienta ampliamente utilizada en entornos de nube que facilita la configuración automática de instancias virtuales y físicas. Esto incluye la configuración de usuarios, contraseñas, claves SSH, redes, almacenamiento y otras opciones de configuración del sistema operativo.

1.2.3 Mantenimiento de un Inventario Actualizado

Uno de los beneficios clave del proyecto será mantener un inventario actualizado de los sistemas desplegados. Esto proporcionará visibilidad sobre el estado y la configuración de cada sistema, lo que facilitará la gestión y el mantenimiento continuo. El inventario incluirá detalles como la, su estado operativo, la configuración de hardware y software, y otra información relevante para la administración del sistema.

2. Fundamentos teóricos y conceptos

2.1 Metal as a service

Metal as a Service (MAAS) es una solución de administración de infraestructura física que convierte servidores físicos en recursos virtualizados fáciles de gestionar y usar. MAAS permite a los administradores de sistemas y desarrolladores tratar servidores físicos de la misma manera que se tratan las máquinas virtuales en la nube, proporcionando una plataforma automatizada para la implementación y gestión de hardware.

Este nuevo concepto introducido por Canonical nos permite tener una serie de beneficios al usarlo :

- Eficiencia: Automatiza tareas repetitivas y complejas de despliegue de hardware, ahorrando tiempo y reduciendo errores humanos.
- **Escalabilidad**: Facilita el escalado horizontal al permitir la rápida implementación de nuevos servidores físicos según sea necesario.
- **Flexibilidad**: Soporta múltiples sistemas operativos y configuraciones de hardware, adaptándose a diversas necesidades empresariales y de desarrollo.
- **Visibilidad y control**: Proporciona una vista unificada de todos los recursos físicos disponibles y su estado, mejorando la gestión y utilización de los activos de hardware.

MAAS implementa todas las características estándar de una nube pública, como los metadatos de instancias y cloud-init. Sus clientes obtienen un control completo sobre la máquina desplegada.

Canonical creó cloud-init y lidera el proyecto a nivel mundial; aseguramos que MAAS proporciona una experiencia de nube de primera clase para servidores físicos basados en arquitecturas x86, ARM64, POWER y Z.

2.2 PXE

PXE (Preboot eXecution Environment) es un entorno estándar para el arranque de una maquina mediante una red, en lugar de hacerlo desde un dispositivo de almacenamiento local como un disco duro, una unidad SSD o una unidad USB. PXE permite que una maquina, conocida como cliente PXE, descargue y ejecute una imagen del sistema operativo desde un servidor remoto.

Este protocolo tiene el siguiente funcionamiento :

- 1. **Solicitud DHCP:** Al encenderse, la computadora cliente PXE envía una solicitud DHCP para obtener una dirección IP y la ubicación del servidor PXE.
- 2. **Respuesta del Servidor DHCP/PXE:** El servidor DHCP responde con una dirección IP y la ubicación del servidor TFTP, donde se encuentra la imagen de arranque.
- 3. **Descarga de la Imagen de Arranque:** El cliente PXE descarga la imagen de arranque desde el servidor TFTP.
- 4. **Ejecución de la Imagen de Arranque:** Una vez descargada, la imagen de arranque se ejecuta y el sistema operativo se carga en la memoria de la computadora cliente.

Al utilizar este protocolo en nuestra red nos permite obtener una serie de ventajas :

- Eficiencia: Simplifica la administración y el despliegue de sistemas operativos en múltiples máquinas.
- **Reducción de Costos:** Reduce la necesidad de medios físicos de instalación y el tiempo de configuración manual.
- **Flexibilidad:** Permite un despliegue rápido y fácil de sistemas operativos, facilitando la gestión de entornos dinámicos y escalables.

2.3 **TFTP**

TFTP (Trivial File Transfer Protocol) es un protocolo de red simple para la transferencia de archivos entre computadoras. Se diseñó para ser fácil de implementar y usar, lo que lo hace adecuado para aplicaciones donde se necesita una transferencia de archivos básica y rápida, sin las características más complejas de otros protocolos como FTP (File Transfer Protocol).

Las características principales de TFTP son :

- Simplicidad: TFTP es un protocolo extremadamente simple en comparación con otros protocolos de transferencia de archivos. Utiliza el protocolo UDP (User Datagram Protocol) en lugar de TCP (Transmission Control Protocol), lo que reduce la complejidad de la implementación.
- Uso de UDP: Al usar UDP, TFTP no tiene control de flujo, control de errores, o gestión de conexiones como TCP. Esto significa que es más rápido, pero también menos robusto en entornos con alta pérdida de paquetes o problemas de red.
- **Operaciones Básicas:** TFTP soporta solo un conjunto limitado de operaciones, incluyendo la lectura y escritura de archivos. No tiene características avanzadas como autenticación, cifrado o comandos de directorio.
- Tamaño de Archivo Limitado: Originalmente, TFTP tenía limitaciones en el tamaño máximo de archivo que podía transferir debido a su uso de un tamaño de bloque fijo y la falta de extensiones para manejar archivos grandes. Sin embargo, se han implementado extensiones para superar estas limitaciones en implementaciones modernas.

2.4 Cloud-init

Cloud-init es una solución de configuración inicial para instancias de máquinas virtuales en la nube. Su principal función es automatizar el proceso de configuración de estas instancias durante su arranque inicial, permitiendo que las máquinas virtuales se configuren de acuerdo con las necesidades específicas del entorno o de la aplicación.

Las principales características de este son :

- Automatización de Configuración: Cloud-init permite automatizar una variedad de tareas de configuración, como la configuración de red, la instalación de paquetes, la configuración de usuarios y la ejecución de scripts personalizados.
- **Soporte Multiplataforma:** Es compatible con una amplia gama de sistemas operativos, incluyendo diversas distribuciones de Linux (como Ubuntu, CentOS, RHEL, Debian) y algunos sistemas operativos basados en Unix.
- Personalización de Imágenes: Permite que las imágenes de máquinas virtuales sean personalizadas para adaptarse a los requisitos específicos de la organización o del entorno de despliegue.
- Múltiples Fuentes de Datos: Cloud-init puede obtener datos de configuración de múltiples fuentes, como metadatos de la nube, configuraciones predefinidas y sistemas de archivos locales.
- **Modularidad:** Está compuesto por varios módulos que realizan tareas específicas. Estos módulos pueden ser habilitados o deshabilitados según las necesidades, proporcionando una gran flexibilidad en la configuración.

2.5 Commissioning

Commissioning en el contexto de Metal as a Service (MAAS) se refiere al proceso de preparación y configuración inicial de un servidor físico antes de que esté disponible para su uso. Durante el commissioning, MAAS realiza una serie de pasos para asegurar que el hardware del servidor está funcionando correctamente y que está configurado de acuerdo con las especificaciones y necesidades de la infraestructura. Este proceso es crucial para garantizar que el servidor esté listo para recibir y ejecutar cargas de trabajo de manera eficiente y confiable.

Este proceso tiene un determinado orden :

- 1. **Descubrimiento de Hardware :** MAAS detecta y cataloga todos los componentes de hardware del servidor, incluyendo CPU, memoria, discos, interfaces de red y otros dispositivos.
- 2. **Pruebas de Hardware :** Se realizan pruebas de diagnóstico para verificar la salud y el rendimiento del hardware, asegurando que no haya fallos ni problemas que puedan afectar su funcionamiento.
- 3. **Configuración de Red MAAS :** configura las interfaces de red del servidor, asegurando que pueda comunicarse correctamente con la red de gestión y otros recursos.
- 4. **Instalación de Software Básico :** Se instala un sistema operativo ligero y se ejecutan scripts de configuración inicial para preparar el servidor para su uso posterior.
- 5. **Captura de Datos de Inventario :**MAAS recoge y almacena información detallada del inventario del hardware, incluyendo números de serie, modelos de componentes y otros detalles relevantes.
- 6. **Aplicación de Configuraciones Personalizadas :** Cualquier configuración personalizada especificada por el administrador, como particionamiento de discos o configuraciones de red específicas, se aplica durante este paso.

2.6 Juju

Juju es una poderosa herramienta de orquestación de servicios desarrollada por Canonical, la misma empresa detrás de Ubuntu. Diseñada para facilitar el despliegue, la gestión y la escala de aplicaciones en entornos de nube, Juju utiliza un modelo basado en charms, que son paquetes de código y configuración necesarios para desplegar y gestionar servicios de software. Con Juju, los administradores pueden automatizar tareas complejas de configuración y mantenimiento, logrando un entorno más eficiente y menos propenso a errores.

Una de las integraciones más destacadas de Juju es con MAAS, la combinación de ambas proporciona un potente entorno híbrido que facilita la administración de infraestructuras físicas y virtuales.

Utilizar Juju ofrece una serie de ventajas significativas, especialmente en entornos de TI donde la eficiencia, la escalabilidad y la automatización son cruciales.

Entre las principales ventajas destacan :

- Automatización del Despliegue: Juju permite automatizar el despliegue de aplicaciones y servicios, reduciendo significativamente el tiempo y el esfuerzo necesarios para configurar entornos complejos. Mediante el uso de charms, Juju encapsula las mejores prácticas y configuraciones necesarias, facilitando un despliegue rápido y eficiente.
- Gestión Simplificada: Con Juju, la gestión de aplicaciones y servicios se simplifica enormemente. Los administradores pueden controlar y supervisar todos los aspectos de sus aplicaciones desde una única interfaz, gestionando dependencias, configuraciones y actualizaciones de manera centralizada.
- **Reducción de Errores Humanos:** Al automatizar muchas de las tareas repetitivas y complejas asociadas con el despliegue y la gestión de aplicaciones, Juju reduce significativamente el riesgo de errores humanos. Esto lleva a una mayor fiabilidad y estabilidad en los entornos de TI.

2.7 Ansible

Ansible es una herramienta de automatización desarrollada por Red Hat. Diseñada para facilitar el despliegue, la gestión y la orquestación de aplicaciones en grandes entornos, Ansible utiliza un enfoque basado en YAML, con archivos llamados playbooks que definen el estado deseado de los sistemas.

Una de las integraciones más destacadas de Ansible es con MAAS (Metal as a Service). La combinación de ambas herramientas proporciona un entorno potente y flexible que facilita la administración de infraestructuras físicas y virtuales.

Una de las características más importantes de Ansible es su omnipotencia. Esto significa que Ansible evita la repetición de tareas previamente realizadas. Antes de ejecutar cualquier acción, Ansible verifica si esta ya se ha realizado, lo que evita redundancias ahorrando tiempo y recursos.

Entre las principales ventajas de este destacan:

- **Automatización:** Ansible permite automatizar el despliegue de aplicaciones y servicios, reduciendo significativamente el tiempo y el esfuerzo necesarios para configurar entornos complejos.
- **Reducción de Errores:** Al automatizar muchas de las tareas repetitivas y complejas asociadas con el despliegue y la gestión de aplicaciones, Ansible reduce significativamente el riesgo de errores humanos.
- **Escalabilidad:** Ansible es altamente escalable, lo que permite gestionar desde unos pocos servidores hasta miles de nodos. Esto es especialmente útil en entornos de nube y centros de datos donde la escalabilidad es una necesidad constante.
- **Integración con otras herramientas:** Ansible se integra fácilmente con una amplia variedad de herramientas y servicios. Es común que muchas empresas proporciones recetas de ejemplo para instalar sus aplicaciones .

3. Preparación del escenario

A continuación, se detallarán los pasos y configuraciones necesarios para establecer el entorno en el que se desarrollará este proyecto.

3.1 Creación de la maquina virtual en Proxmox

Para llevar a cabo este proyecto, crearemos una máquina virtual que cumpla con los requisitos mínimos necesarios para que MAAS funcione correctamente. Estos requisitos incluyen al menos un procesador de 4.5 GHz, 4 GB de RAM y 50 GB de almacenamiento.

Para asegurarnos de que esta tenga un rendimiento óptimo, asignaremos a la máquina anfitriona 8 GB de RAM, 2 núcleos de procesador y 100 GB de almacenamiento.

Además le añadiremos 2 tarjetas de red ya que utilizaremos la segunda para crear nuestro entorno y aislarlo de la red del instituto .



Omitiré el proceso de instalación ya que no requiere ningún paso especifico para el proyecto , esta llevara un Ubuntu Server 22.04.

3.2 Configuración de red de la maquina virtual

A continuación, configuraremos ambas tarjetas de red utilizando Netplan. Una interfaz se conectará a la red externa, la cual utilizaremos para acceder a la máquina y para la conexión a Internet. La segunda interfaz será utilizada para la conexión de nuestros clientes.

Te muestro el fichero de configuración de red de la maquina :

```
javiercruces@maas-fjcd:~$ sudo cat /etc/netplan/00-installer-
config.yaml
# This is the network config written by 'subiquity'
network:
   version: 2
   ethernets
   ens18:
      dhcp4: true
   ens19:
      dhcp4: no
      addresses:
      - 192.168.77.1/24
```

Continuaremos configurando el bit de forwarding de nuestra maquina para que esta actué como router en nuestra red interna :

```
javiercruces@maas-fjcd:~$ echo "net.ipv4.ip_forward = 1" | sudo
tee -a /etc/sysctl.conf
net.ipv4.ip_forward = 1
javiercruces@maas-fjcd:~$ sudo sysctl -p
net.ipv4.ip_forward = 1
```

Para finalizar con este apartado configuraremos el SNAT y guardaremos la regla de forma permanente :

```
javiercruces@maas-fjcd:~$ sudo iptables -t nat -A POSTROUTING -s
192.168.77.0/24 -o ens18 -j MASQUERADE
javiercruces@maas-fjcd:~$ sudo iptables-save >
/etc/iptables/rules.v4
```

3.3 Instalación de MAAS

3.3.1 Añadir repositorios

Lo primero que haremos sera añadir el repositorio de MAAS a nuestra maquina virtual.

```
javiercruces@maas-fjcd:~$ sudo apt-add-repository ppa:maas/3.4
```

A continuación actualizaremos los repositorios de nuestro sistema para que este nuevo que acabamos de añadir comience a actualizarlo nuestro sistema .

javiercruces@maas-fjcd:~\$ sudo apt update -y

Ahora instalaremos MAAS con el siguiente comando .

javiercruces@maas-fjcd:~\$ sudo apt install -y maas

Este comando se encargara de instalar todas las dependencias necesarias para el paquete y comenzara con la instalación del mismo .

3.3.2 Creación de un usuario administrador

Ahora procederemos a crear un usuario administrador para MAAS. Durante este proceso, elegiremos un nombre de usuario, una contraseña, un correo electrónico y una clave pública SSH.

javiercruces@maas-fjcd:~\$ sudo maas createadmin

Username: javiercruces

Password:

Again:

Email: contacto@javiercd.es

Import SSH keys [] (lp:user-id or gh:user-id): gh:javierasping

Este usuario sera el que utilicemos posteriormente para autenticarnos en la interfaz web asi como en la CLI.

3.3.3 Configuración inicial web

La herramienta nos solicitara una configuración inicial la cual podemos realizar desde la interfaz web . Si queremos acceder a la misma el formato para acceder es el siguiente "http://IP:5240/MAAS" .

← → C ▲ No es seguro 172.22.6.174:5240/MAAS/r/	🖙 ☆ む 💷 🌒 :
😥 WhatsApp ອ PLEDIN 3.0 🔥 Moodle centros 🦳 Redmine IES 🧰 Vista general d	» 🗅 Todos los marcadores
MAAS	
Login	
* Username	
javiercruces	
* Password	
Login	

Una vez iniciada la sesión, se nos solicitará realizar una configuración inicial del servicio. El primer paso será configurar el nombre de nuestro servidor MAAS.



A continuación, configuraremos algunos parámetros de la red. Utilizaremos dos servidores DNS: el DNS local del instituto como servidor principal y el DNS de Cloudflare como servidor secundario. Los repositorios de archivos de Ubuntu los mantendré con la configuración predeterminada.

En el caso de que queramos utilizar otros repositorios para encontrar los paquetes cambiaremos ambos parámetros , en mi caso utilizare los servidores de Ubuntu .

Si en nuestra red contamos con un servidor proxy podremos configurarlo para que este busque en este servicio los paquetes y no sea necesario que salga a internet , pero no contamos con uno en nuestra infraestructura , así que lo dejaremos sin configurar .



MAAS METAL AS A SERVICE

Continuaremos configurando las imágenes que queremos tener en nuestro MAAS, por defecto nos traerá instalada la imagen de Ubuntu 20.04 LTS. Las imágenes se irán sincronizando automáticamente con la ultima versión disponible en los repositorios.

← → C ▲ No es seguro 17	2.22.6.174:5240/MAAS/r/intro/images									
🚫 WhatsApp @ PLEDIN 3.0 🗛 M	100dle centros 🛛 🙆 Redmine IES 🧧 Vista ç	general d 🛕 APUNT	ES 💠 SLAAC	C MATES.SCHOOL In Mood	le Incibe 🛛 🖷 Epic Games St	🛥 Prime Gaming 🛛 🛚 Iniciar s	sesión:			
MAAS	Images									
Ø javiercruces Log out	Showing images synced from ma	as.io								
	Select images to be imported and kept in sync daily. Images will be available for deploying to machines managed by MAAS.									
	Ubuntu releases				Architectures for 22	.04 LTS				
	O 24.04 LTS	○ 23.	0		amd64					
	• 22.04 LTS	○ 23.0)4		arm64					
	O 20.04 LTS	○ 22.1	10		🗌 armhf					
	O 18.04 LTS	O 21.	10		🗌 i386 🛈					
	O 16.04 LTS	○ 21.0)4		ppc64el					
	O 14.04 LTS	○ 20.1	0		s390x	s390x				
	12.04 LTS									
	RELEASE ^	ARCHITECTURE	SIZE	STATUS		LAST DEPLOYED				
	24.04 LTS	amd64	425.9 MB	O Downloading 859 Sun, 19 May. 2024 1	% 17:29:59	-				
	22.04 LTS	amd64	1.2 GB	O Downloading 0% Sun, 19 May. 2024 1	17:29:59	-				
	20.04 LTS	amd64	1.2 GB	Sun, 19 May. 2024 1	17:29:59	-				
	maas-fjcd MAAS: 3.4.2									

En mi caso he seleccionado descargar las versiones de Ubuntu 24.04 LTS y 22.04 LTS .

Adicionalmente podremos descargarnos imágenes de CentOS, en mi caso me descargare la ultima versión :

Other images			
CentOS 8			
CentOS 7			
RELEASE ^	ARCHITECTURE	SIZE	STATUS
CentOS 8	amd64	655.8 MB	O Downloading 72%
			Sun, 19 May. 2024 17:42:26

MAAS METAL AS A SERVICE

Si continuamos con la configuración inicial , nos indicara que una vez que habilitemos el DHCP y el PXE podremos listar automáticamente los hosts de nuestra red y comenzar con las instalaciones automáticas .



Por ultimo nos indicara si queremos añadir alguna clave publica adicional a nuestro usuario , en mi caso solo dejare una añadida .

÷ +		▲ No es seguro	0 172.22.6.174:5240/1									¢	∞ ☆ ᡗ = ₹ 🔍 :
😒 Wł	atsApp	PLEDIN 3.0	A Moodle centros	n Redmine IES	🗖 Vista general d	🏠 APUNTES 🛛 💠 SLA	AC 🛛 MATES.SCHOOI	. 🚮 Moodle In	ibe 🖉 Epic Games St	😐 Prime Gamin	g 🛚 🕏 Iniciar sesión:		» 🗅 Todos los marcadores
	MAAS		SSH keg	ys for javiercru	ices								
0	javiercr Log oul		Add multiple	keys from Launch	pad and Github or ent	er them manually.							
			Keys										
			SOURCE			ID			,	EYACTIONS			
			GitHub			javieras	ping		2	sh-rsa AAAAB3Nza(C1y		ŵ
			Source					1	Before you can deploy a n to the deployed machine	nachine you must impo can be accessed.	rt at least one public SS	H key into MAAS,	
			Select source	e				~					
			8 Error: Sour	ce is required									
			About SSH ke	eys									
													Import SSH key
												Skip us	er setup Finish setup

Con esto hemos finalizado la configuración inicial de MAAS.

3.4 Configuración del servicio DHCP

A continuación vamos a proceder a configurar el DHCP para nuestra red interna , en el proceso de instalación tendremos instalado el servidor isc-dhcp-server , ya que viene contenido en el metapaquete de MAAS .

Lo primero que haremos sera decirle al servicio en que interfaces queremos que funcione, en mi caso la interfaz ens19.

```
javiercruces@maas-fjcd:~$ sudo cat /etc/default/isc-dhcp-server |
grep -v '#'
INTERFACESv4="ens19"
INTERFACESv6=""
```

Lo siguiente sera hacer la propia configuración para nuestra red , en mi caso es la siguiente :

```
javiercruces@maas-fjcd:~$ sudo cat /etc/dhcp/dhcpd.conf | grep -v
'#'
default-lease-time 600;
max-lease-time 7200;
ddns-update-style none;
subnet 192.168.77.0 netmask 255.255.255.0 {
  range 192.168.77.100 192.168.77.200;
  option routers 192.168.77.1;
  option subnet-mask 255.255.255.0;
  option domain-name-servers 172.22.0.1, 1.1.1.1;
  option broadcast-address 192.168.77.255;
  filename "pxelinux.0";
  next-server 192.168.77.1;
```

}

En la configuración anterior hemos declarado el servicio DHCP en la red 192.168.77.0/24 , en la cual he decidido que el rango de direcciones que se repartirán sera de la 192.168.77.100 a la 192.168.77.200.

La puerta de enlace de esta red es la 192.168.77.1 y los DNS configurados son 172.22.0.1 y 1.1.1.1 .

Lo mas importante son los 2 últimos parámetros ya que con estos nos permitirá indicar en nuestra red donde se encuentra el servidor TFTP y el nombre del fichero que tenemos que pedir , lo cual nos permitirá arrancar por red y utilizar MAAS .

Por ultimo solo nos queda reiniciar el servicio para que comience a aplicarse la nueva configuración en la red :

javiercruces@maas-fjcd:~\$ sudo systemctl restart isc-dhcp-server

Una vez reiniciado, vamos a comprobar que el servicio este corriendo:

javiercruces@maas-fjcd:~\$ sudo systemctl status isc-dhcp-server

• isc-dhcp-server.service - ISC DHCP IPv4 server

Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)

Active: active (running) since Mon 2024-05-20 17:13:10 UTC; 17h ago

Docs: man:dhcpd(8)

Main PID: 32279 (dhcpd)

Tasks: 4 (limit: 9389)

4. Despliegue de imágenes de MAAS en Proxmox

4.1 Introducción

En esta sección, realizaremos el despliegue de una imagen ubuntu de MAAS utilizando una máquina virtual en Proxmox. El proceso abarcará desde la creación de la máquina virtual, su incorporación al inventario de MAAS, hasta la personalización de la instalación con cloud-init. Finalmente, verificaremos que la instalación y la personalización se han realizado correctamente.

Como estamos en un escenario "ficticio", voy a crear una maquina que posteriormente sera el controlador juju que utilizaremos en posteriores apartados.

Puedes encontrar este apartado en vídeo pulsando aquí.

4.2 Creación de la maquina virtual en proxmox

Comenzaremos creando la maquina en proxmox y añadiéndola a la red en la que esta funcionando nuestro servidor MAAS, con la peculiaridad de que no le seleccionaremos ningún medio de instalación.

La llamare juju , para identificarla posteriormente . Es importante que recordemos el valor de VM ID ya que lo utilizaremos posteriormente :

Create: Virtual Machine									
General OS									
Node:	proxmox1				Resource Po	ool;	Proyecto_javier.cruces_iesgn		
VM ID:	122								
Name:	juju						2		

MAAS METAL AS A SERVICE

No seleccionaremos ningún medio de instalación ya que arrancaremos la maquina usando PXE :

Create: Virtual Machine									
General OS System Disks									
Use CD/DVD disc image file (iso)	\triangleright	Guest OS:							
Storage: local		Туре:	Linux						
ISO image:		Version:	6.x - 2.6 Kernel						
OUse physical CD/DVD Drive									
Do not use any media									

Le añadiré un único disco de 50GB a la maquina virtual :

Create: Virtual Mach	nine				\otimes
General OS S	ystem Disks	CPU Memory Netwo			
virtio0 🛱	Disk Bandwi	idth			
	Bus/Device:	VirtlO Block V 0	Cache:	Default (No cache)	
	Storage:	local-lvm \vee	Discard:		
N	Disk size (GiB):	50 🗘	IO thread:		
45					

Le pondré 2 núcleos ya que no es necesario que tenga mas potencia :

Create: Virtual Machine									
General	os	System	Disks	CPU	Memory	Network			
Sockets:		1				С } Туре:			
Cores:		2				Total cores:	2		

Le pondremos 4GB de memoria RAM :

MAAS METAL AS A SERVICE

Create: Virtual Machine								
General OS	S System	Disks	CPU	Memory	Network			
Memory (MiB):		4096						

Por ultimo la añadimos en la misma red que esta funcionando nuestro servidor MAAS :

Create: Virtual Machine										
General OS	System	Disks	CPU	Memory	Network	Con	firm			
No network device										
Bridge:	vmbr210				Model:		VirtlO (paravirtualized)			
VLAN Tag:					MAC address	5:				
Firewall:										

4.3 Añadir la maquina a MAAS

El siguiente paso a seguir es añadir nuestra maquina a maas , para realizar esto simplemente encenderemos la maquina y veremos que arrancara por PXE :

S WhatsApp 💮 PLEDIN 3.0 🛕 Mood	odle centros 🛛 🦰 Redmine IES 🧧 Vista g	general d 🔞 APUNTES 🎄 SLAAC 🛛 🖉 MATES.SCHOOL \min Moodle Incibe 👼 Epic Games St 🛥 Prime Gaming 🛚 S	Iniciar sesión:		» D	Todos los marcadores
	ent 7.4-17 Search		Documentation	Create VM	Create CT	🔺 javier.cruces@iesgn 🗸
Server View 🗸 🕴	Virtual Machine 122 (juju) on node 'proxmox1'	No Tags 🖋		🖒 Shutdown 🗸 🗸		More V 😧 Help
> Boutacenter ↓ Bo promoti ↓ 122 (may) ↓ 22 (may) ↓ 22 (may) ↓ 22 (may) ↓ 122 (may) ↓	Summary Console Console A Cloud rink Options Tak Heary Kakup Replication S Replication Firewall Firewall P Permissions	SeaBIOS (version rel-1.16.1-0-g3208b098f51a-prebuilt.gemu. Machine UUID 4ea26c74-6215-46a2-a2c5-beaf7909b29a Booting from Hard Disk Boot failed: not a bootable disk Booting from DUD/CD Boot failed: Could not read from CDROM (code 0003) Booting from ROM iPXE (PCI 00:12.0) starting executionok iPXE initialising devicesok iPXE 1.20.1+ (g4bd0) Open Source Network Boot Firmware Features: DNS HTTP iSCSI TFTP AOE ELF MBOOT PXE bzImage Me Press Ctrl-B for the iPXE command line	.org) http enu PXEX	: // i pxe . F	org	
\$						

En este momento, MAAS está inventariando la máquina con todas sus características de hardware y la añadirá a nuestro inventario. Este proceso se realiza automáticamente, sin necesidad de intervención por nuestra parte, y tomará unos pocos minutos en completarse. Una vez finalizado, la máquina se apagará y podremos verla en nuestro inventario.

MAAS METAL AS A SERVICE

Cuando finalice la podremos ver en la sección de machines :

							Para satir de la	pantatta com	olela, pulsa Esc							
÷		▲ No es segu												\$	ទា	≕ i
😒 V	/hatsApp	PLEDIN 3.0	A Moodle centros	n Redmine IES	🗖 Vista general d.	. 🔞 APUNTES	👙 SLAAC 🛛 🗠 M	IATES.SCHOOL	in Moodle Incibe	🖷 Epic Games St	t 😐 Prime Gam	ing 🗴 Inicia	r sesión:	»	🗅 Todos lo	s marcadores
	MAAS		1 machines	in 1 pool Filt	iers v	Search					C	Group	by status 🛛 🗸	Add hardware	~ @ Cr	olumns
8	HARDWA	ARE														
			Showing 1 out	of 1 machines									< Page	1 of 1 >	50/page	~
		s Ilers	FQDN ^ M	IAC	POW	/ER	STATUS		OWNER NAME TAGS	POOL NOTE	ZONE SPACES	ABRIC /LAN	CORES ARCH	RAM	DISKS	STORAGE
&	кvм LXD		Commissi 1 machine	ioning												-
0	organis Tags	SATION	awake	-camel .maas wake-camel.maas	? Uni	nown 🗸 👘	Commissioning	1 ~	- V virtual	default 🗠	default 🗸	a bric-1 Default VLAN	2 amd64	4 GIB	1	53.7 GB
.8	Pools CONFIGU	JRATION	Local document	ation • Legal infor	mation • Give feedb	ack								CAN	IONI	CAL

Vemos que la ha nombrado de forma aleatoria como "awake-camel.maas", en mi caso le cambiare el nombre a juju :

← → C ▲ No es segu					☆ む 🖛 🔍 🗄
😒 WhatsApp 💮 PLEDIN 3.0	🛕 Moodle centros 🏾 🧑 Redmine IES 🗧 Vista general d	I 🔞 APUNTES 🌵 SLAAC 🛛 MATES.SCHOOL 🔢 Mood	le Incibe 🛱 Epic Games St 🛥 Prime Gaming	S Iniciar sesión: »	🗅 Todos los marcadores
MAAS B HARDWARE Machines	juju.maas Commissioning ? Power unknown \vee Actio	ons V Categorise V 🗟 Delete			
Devices	Summary Network Storage PCI devices	s USB O Commissioning Tests Logs Conl	ïguration		
Concrollers KVM LXD ORGANISATION Tags	virtual machine status Commissioning	CPU amd64/generic 2 cores Common KVM processor Test CPU	MEMORY 4 GIB Test memory	STORAGE 53.7 GB over 1 disk Test storage	
AZs Pools	Owner -	Zone > Resource default default	pool>	Power type > None	Tags > virtual
B CONFIGURATION Images ✓ NETWORKING Subnets DNS	HARDWARE INFORMATION System Vendor QEMU	1 NUMA NODE Node 0 CPU cores 2 (0-1)	NETWORK / Red Hat, Inc. NAME IP ADDRESS MAC ADDRESS SUBNET	LINK SPEED FABRIC ① DHC VLAN	P SR-IOV
Network discovery	Product Standard PC (i440FX + PIIX, 1996)	Memory 4 GIB Storage 53.69 GB over 1 disk	ens18 192.168.77.111 fe:d1:b9:c9:67:ae	0 Mbps Fabric-1 No Default VLAN	DHCP No
Settings	Version pc-i440fx-7.2 Serial Unknown	Network 1 interface	Information about tagged traffic can be see	n in the Network tab.	
ø javiercruces Log out	Mainboard Vendor Unknown				
	Firmware SeaBIOS				

Como puedes observar en la captura superior, tenemos acceso a toda la información referente al hardware de esta máquina, incluyendo el número de núcleos, la cantidad de memoria RAM, el número de discos, las interfaces de red y su configuración.

Además tenemos la posibilidad ver con mas detalle los mismos.

4.4 Explorando la interfaz gráfica de MAAS

A continuación vamos a recorrer las distintas opciones que nos ofrece MAAS sobre una maquina que tenemos en nuestro inventario .

4.4.1 Network

Si accedemos al apartado de network podremos ver la configuración que tienen las distintas tarjetas de red . En la cual podremos saber la dirección IP que tienes las mismas , la dirección MAC , a que VLAN pertenece cada interfaz entre otras configuraciones .



Además podemos incluso cambiar la configuración de las mismas , aunque esto solo podremos hacerlo si son maquinas virtuales y tenemos configurada el acceso al hipervisor , en el caso de que sean maquinas físicas solo nos dejara cambiar parcialmente esta configuración .

÷	→ C ▲ No es seguro	17.	2.22.6.174:5240												☆ ⊅	₹ 🌒 :
😒 Wi	atsApp 💮 PLEDIN 3.0	<mark>А</mark> М	loodle centros	n Redmine	IES 🧧	Vista general d	🔊 APUN	TES 💠 SLAAC 🤉	Z MATES.SCHOOL	n Moodle Incibe	e 👼 Epic Games S	St 🛥 Prime Gaming	S Iniciar sesión:		🗅 Todos la	s marcadores
	MAAS		Summary	Network	Storage	PCI devices	USB	Commissioning	g 🔮 Tests 🔮	Logs Conf	iguration					
8	HARDWARE Machines		 Interface 	ce configurati	ion cannot b	e modified unle	s the mach	nine is New, Ready	, Allocated or Broke	n.						
0	Devices Controllers		Edit Phys	ical												
œ	LXD		NAME MAC			PXE	LINK	/INTERFACE SPEED	TYPE NUMA NODE	FABRIC VLAN	C	SUBNET NAME	IP ADDRESS STATUS	DH	CP	
0	organisation Tags		ens18 fe:d1:b9:c	:9:67:ae		~	0 M	bps/0 Mbps	Physical 0	fabric untage	c-1 ged	192.168.77.0/24 192.168.77.0/24	192.168.77.4	No	DHCP	
	Pools		Physical del	tails						Net	work					
.8	CONFIGURATION		Name							* Fat	bric					
	lmages NETWORKING		ens18							fab	oric-1					~
			MAC addres	is						* VL	AN					
	DNS		fe:d1:b9:c9	9:67:ae						unt	tagged					~
~	Settings		Tags							Subr	net					
~	secongs		Tags							192	2.168.77.0/24					~
8	javiercruces Log out		Link speed ((Gbps)						IP m	iode					
			0							Aut	to assign					~
			Interface sp	eed (Gbps)												
			0													

4.4.2 Storage

En esta sección podremos ver los distintos dispositivos de bloques de las que dispone nuestra maquina , obteniendo toda la información referente a los mismos :

÷	→ C	▲ No es segur	o 172.22.6.174:5240/1	MAAS/r/machine/ex4	lhw8/storage	Scree	nshot aborted.				☆ 🖸 🛡 🤱 🗄
<mark>(S</mark>) W	'hatsApp	PLEDIN 3.0	\Lambda Moodle centros	n Redmine IES	. 🗧 Vista general d	🕅 APUNTES 🛛 🙀 S	.AAC 🛛 MATES.SCHOO	DL 🛭 in Moodle Incibe 👼 Epic Gam	es St 👛 Prime Gaming	S Iniciar sesión:	» 🗅 Todos los marcadores
	MAAS		juju.maas								
8	HARDWA	ARE	Deployed (り Poweron 〜	Actions V Power	cycle 🗸 Trouble	shoot ~ Categorise	✓ ○ Lock			
	Machir	nes	6	National Cha	DCI de las			• Loss Configuration			
			Summary	Network Sto	rage PCI devices	USB 🔮 Comm	ssioning 🔮 lests	Logs Configuration			
&	KVM		③ Storage	configuration can	not be modified unless	the machine is Read	<i>.</i>				
0	ORGANI	SATION									
			Filesystems								
	AZs		NAME		SIZE		FILESYSTEM	MOUNT POINT	MOUNT O	PTIONS	ACTIONS
.8	CONFIG	JRATION	vda-part2		53.67 GB		ext4	/			

Además podremos ver el numero de particiones que tenemos así como su ruta de montaje :

Used disks and pa	rtitions					
NAME SERIAL	MODEL FIRMWARE	BOOT	SIZE	TYPE NUMA NODE	HEALTH TAGS	USED FOR
vda		√	53.68 GB	Physical 0	 OK 1rpm, rotary 	GPT partitioned with 1 partition
vda-part2	_	_	53.67 GB	Partition	_	ext4 formatted filesystem mounted at /

Como puedes ver, hay un apartado de salud del sistema que nos indicará si algún disco está dañado. Esto se determina mediante un componente del disco llamado SMART.

Además, en secciones posteriores, veremos que disponemos de pruebas adicionales para evaluar el estado de los discos.

En esta sección , no podremos realizar ninguna modificación como podíamos hacer en el apartado anterior .

4.4.3 PCI Devices

En este apartado, simplemente podemos listar los distintos componentes de la placa base de nuestra máquina que utilizan esta interfaz. Esto nos puede ser útil a la hora de diagnosticar problemas de hardware, planificar actualizaciones o asegurarnos de que todos los componentes están correctamente reconocidos y funcionando.

÷	\rightarrow C \triangle No es seguro	172.22.6.174:5240/MA	AAS/r/machine/ex4hw8/pci-devices					± 1	Ď ₹ 🌒 :
😒 W	hatsApp 💮 PLEDIN 3.0	🗚 Moodle centros	🗅 Redmine IES 🧧 Vista general d	🔞 APUNTES 💠 SLAAC	C MATES.SCHOOL In Moodle Incibe	👼 Epic Games St 👛 Prime G	aming 🛚 🕏 Iniciar sesión:	» DT	odos los marcadores
	MAAS	juju.maas							
8	HARDWARE	Deployed 😃	Power on $ \lor $ Actions $ \lor $ Power	r cycle 🗸 🛛 Troubleshoot	✓ Categorise ✓ ○ Lock ₫	້ງ Delete			
	Machines Devices	Summary N	Network Storage PCI devices	USB 🔮 Commissioni	ng 🛛 Tests 🖉 Logs Configu	ration			
&	Controllers кvм		VENDOR ID		PRODUCT		DRIVER	NUMA NODE	PCI ADDRESS
0	LXD organisation	Network 1 device	Red Hat, Inc. 1af4		Virtio network device		virtio-pci	0	0000:00:12.0
	Tags AZs	Storage 1 device	Red Hat, Inc. 1af4		Virtio block device		virtio-pci	0	0000:00:0a.0
	Pools configuration	GPU 1 device	1234					0	0000:00:02.0
	lmages NETWORKING	Generic 9 devices	Intel Corporation 8086		440FX - 82441FX PMC [Natoma] 1237			0	0000:00:00.0
	Subnets DNS		Intel Corporation 8086		82371SB PIIX3 ISA [Natoma/Triton II] 7000			0	0000:00:01.0
	Network discovery		Intel Corporation 8086		82371SB PIIX3 IDE [Natoma/Triton II] 7010		ata_piix	0	0000:00:01.1
۲			Intel Corporation 8086		82371SB PIIX3 USB [Natoma/Triton II] 7020		uhci_hcd	0	0000:00:01.2
0			Intel Corporation 8086		82371AB/EB/MB PIIX4 ACPI 7113		piix4_smbus	0	0000:00:01.3

Además nos lo divide por categoría y nos muestra el driver que se esta utilizando .

4.4.4 USB

Tendremos otra sección para listar los dispositivos USB conectados a nuestras maquinas :

÷	\rightarrow G	A No es segu	ro 172.22.6.174:5240/1	MAAS/r/machine/ex4h	w8/usb-devices									± ۲) ₹ 🌒 E
<u>9</u> v	WhatsApp	PLEDIN 3.0	A Moodle centros	n Redmine IES	🖬 Vista general d	APUNTES	💠 SLAAC 🛛	Z MATES.SCHOO	DL 🛛 in Moodle Incibe	🍯 Epic Games St	😐 Prime Gaming	S Iniciar sesión:		» 🗅 T	odos los marcadores
	MAAS		juju.maas												
8	HARDW	ARE	Deployed (D Power on 🗸	Actions \vee Powe	r cycle 🗸 🛛 Tr	oubleshoot 、	 Categorise 	✓ ◯ Lock	🕆 Delete					
	Machir			Nationali Char	DCI devices			T h.	• I						
			Summary	Network Store	ige PCI devices	USB 🔮 (ommissioning	j 🔮 Tests	Confi	guración					
ھ	KVM			VENDOR ID			PRODU ID	ICT			DRIVER		NUMA NODE	BUS ADDRESS	DEVICE ADDRESS
			Conoris	Adamay Tr	shaalaay Co. 11d		OFM				uchhid		0	1	2
0	ORGANI	SATION	1 device	0627	crinology co., Lto		0001	J OSB TADIEL			usphila		0	1	2
														~ • • • ~	
-8	CONFIG	URATION	Local documen	tation • Legal infor	mation • Give feedba	ck								LANO	NICAL

Lo cual nos ayudara a tener un control de los dispositivos que hemos conectado a la maquina o para asegurarnos de que nadie añada ningún dispositivo adicional .

4.4.5 Commissioning

El "commissioning" es un proceso clave en MAAS (Metal as a Service) que se encarga de detectar, identificar e inventariar el hardware de una máquina. Este proceso se realiza cuando se añade una nueva máquina al entorno gestionado por MAAS o cuando se desea actualizar la información de una máquina existente.

juju.maas Deployed ひ Power on 〜 Actions 〜 Po	wwer cycle $ \lor $ Troubleshoot $ \lor $ Ca	ategorise V 💭 Lock 🝵 Delete			
Summary Network Storage PCI device	S USB Commissioning	Tests 🛛 Logs Configuration			
NAME	TAGS	RESULT	DATE	RUNTIME	ACTIONS
20-maas-01-install-lldpd	node	Passed	Fri, 07 Jun. 2024 16:11:18	0:00:00	~
20-maas-02-dhcp-unconfigured-ifaces	node	Passed	Fri, 07 Jun. 2024 16:11:23	0:00:05	\sim
20-maas-03-machine-resources	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:24	0:00:00	\sim
30-maas-01-bmc-config	bmc-config, node	🛕 Skipped	Fri, 07 Jun. 2024 16:11:30	0:00:00	\sim
40-maas-01-machine-config-hints	node	Passed	Fri, 07 Jun. 2024 16:11:30	0:00:00	~
50-maas-01-commissioning	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:30	0:00:00	\sim
maas-capture-lldpd	node	Passed	Fri, 07 Jun. 2024 16:12:18	0:00:46	\sim
maas-get-fruid-api-data	node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	\sim
maas-kernel-cmdline	node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	\sim
maas-list-modaliases	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	\sim
maas-lshw	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:32	0:00:01	\sim
maas-serial-ports	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	~
maas-support-info	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	~

Como puedes observar, tenemos la capacidad de ver el tiempo que han tardado en ejecutarse los distintos scripts durante el proceso de commissioning. Esta información es crucial para asegurarnos de que cada script se ha ejecutado correctamente y que el proceso en su totalidad se ha completado sin errores. Además de la duración de la ejecución, también se registra la fecha y la hora de inicio y finalización de cada script.

juju.maas				
Deployed $$ Power on \checkmark	Actions \checkmark Power cycle \checkmark Troubleshoot \checkmark	Categorise V 💭 Lock 🗊 Delete		
Summary Network St	orage PCI devices USB OCommissioning	Tests Logs Configuration		
20-maas-01-install-lldpd	details			< Back to test results
Status Exit status Tags	Passed 0 node	Start time End time Runtime	Fri, 07 Jun. 2024 16:11:17 Fri, 07 Jun. 2024 16:11:18 0:00:00	
Output combined stdout stde	rr yaml			
INFO: Configuring lldpd. INFO: Restarting lldpd	 •			
Local documentation • Legal inf	ormation • Give feedback			CANONICAL

4.4.6 Test

MAAS (Metal as a Service) nos proporciona una serie de tests integrados para verificar el estado del hardware, además permite la adición de tests personalizados para satisfacer necesidades específicas. Estos tests son esenciales para asegurar que todo el hardware de las máquinas gestionadas está en óptimas condiciones y funciona correctamente.

Juju.maas					
Deployed 🖞 Power on 🗸 🛛 Actions 🗸 Power cycl	e ~ Troubleshoot ~ Categoris	e 🗸 🔘 Lock 🗄 Delete			
Summary Network Storage PCI devices USI	3 O Commissioning O Tests	Logs Configuration			
Storage					
/dev/vda					
SUPPRESS NAME	TAGS	RESULT	DATE	RUNTIME ACTION	S
smartctl-validate	commissioning, storage	A Skipped	Fri, 07 Jun. 2024 16:12:22	0:00:00	,
smartctl-validate	commissioning, storage	A Skipped	Fri, 07 Jun. 2024 16:12:22	0:00:00 ~	/
smartctl-validate Local documentation • Legal information • Give feedback	commissioning, storage	A Skipped	Fri, 07 Jun. 2024 16:12:22		-

El test que trae por defecto verifica el estado de los discos duros y SSDs utilizando SMART. Este sistema de monitoreo permite detectar y predecir fallos en los discos .

juju.maas				
Deployed () Power on \checkmark	Actions \checkmark Power cycle \checkmark Troubleshoot \checkmark	Categorise ∨ ◯ Lock		
Summary Network St	orage PCI devices USB 🥥 Commissioning	Tests Configuration		
smartctl-validate details				< Back to test results
Status Exit status Tags	 Skipped o commissioning, storage 	Start time End time Runtime	Fri, 07 Jun. 2024 16:12:22 Fri, 07 Jun. 2024 16:12:22 0:00:00	
Output combined stdout stdo	err yaml			
INFO: Verifying SMART s INFO: Running command: INFO: Unable to run tes	upport for the following drive: /dev/vda sudo -n smartctlall /dev/vda t. The following drive does not support SMAR	f: /dev/vda		
Local documentation • Legal in	formation • Give feedback			CANONICAL

En nuestro ejemplo al tratarse de un disco virtual este no cuenta con la tecnología SMART, ya que esta es propia de los discos físicos, así que se ha saltado el test.

4.4.7 Logs

Los logs proporcionan información detallada sobre el comportamiento del sistema, permitiendo a los administradores identificar y resolver problemas de manera eficiente. Cada evento, error y advertencia se registra, facilitando la localización de fallos y su posterior corrección.

juju.maas	
Deployed 🖰 Power on 🗸 Actio	ns 🗸 Power cycle 🗸 Troubleshoot 🗸 Categorise 🗸 💭 Lock 🔮 Delete
Summary Network Storage	PCI devices USB O Commissioning O Tests O Logs Configuration
Event log Installation output	Download \sim
Search event logs	Q. Show <u>25 v</u> < >
TIME	EVENT
⅔ Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-final_message with frequency always
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-keys_to_console with frequency once-per-instance
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-scripts_user with frequency once-per-instance
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-scripts_per_boot with frequency always
资 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-scripts_vendor with frequency once-per-instance
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-package_update_upgrade_install with frequency once-per-instance
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-ntp with frequency once-per-instance
涨 Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-grub_dpkg with frequency once-per-instance
⅔ Fri, 07 Jun. 2024 16:20:59	Node status event - 'cloudinit' running config-locale with frequency once-per-instance

Los logs de la instalación los tendremos en un apartado separado de los logs que se han generado por el proceso de inventariado y test .En concreto veremos los comandos que se han ejecutado durante la instalación del sistema operativo .

juju.m	aas
Deploye	d O Power on V Actions V Power cycle V Troubleshoot V Categorise V O Lock 🗊 Delete
Summai	y Network Storage PCI devices USB 📀 Commissioning 📀 Tests 💽 Configuration
Event lo	g Installation output
1 0	rurtin: Installation started. (23.1.1-1099-g505dd3a9-Oubuntu1-ubuntu22.04.1)
2 1	hird party drivers not installed or necessary.
3 5	itari: cmd-install/stage-partitioning/builtin/cmd-block-meta: curtin command block-meta
4 9	<pre>let_path_to_storage_volume tor volume vaa({ grub_device: irue, `td: 'vda', `name: `vda', `path:: 'dev/vda', `ptable': 'gpt', `type: 'disk', wipe': 'superblock'})</pre>
51	unning command [udevadm , info ,query=property ,export , /dev/vda] with allowed return codes [0] (capture=Irue) (dau/ud-is: multianth dauice metheds Eales
7 1	devy va is multipali device memori : raise unning command ['astronsha' : /du/uda'] with allowed rature codes [0, 1] (canture-Ealse)
8 6	Namina (joho na service) a vice a consected in consected in consected a consecte
9 1	The Dulevalm settle(): 0.067
10 0	levsync happy - path /dev/vda now exists
11	eturn volume path /dev/vda
12 0	et path to storage volume for volume vda-part1({'device': 'vda', 'flag': 'bios_grub', 'id': 'vda-part1', 'number': 1, 'offset': '4194304B', 'size': '1048576B', 'type': 'par
13	et_path_to_storage_volume for volume vda({'grub_device': True, 'id': 'vda', 'name': 'vda', 'path': '/dev/vda', 'ptable': 'gpt', 'type': 'disk', 'wipe': 'superblock'})
14 F	Running command ['udevadm', 'info', 'query=property', 'export', '/dev/vda'] with allowed return codes [0] (capture=True)
15 /	'dev/vda is multipath device member? False
16 F	Running command ['partprobe', '/dev/vda'] with allowed return codes [0, 1] (capture=False)
17 F	Running command ['udevadm', 'settle'] with allowed return codes [0] (capture=False)

4.4.8 Configuración

En esta sección podemos encontrar información general sobre nuestras máquinas, como la arquitectura, la zona y el pool de recursos asignado. Además, podemos añadir etiquetas personalizadas a cada máquina para identificarlas y gestionarlas con mayor facilidad. Estas etiquetas permiten una organización más eficiente, facilitando la búsqueda y la administración de los recursos según las necesidades específicas del entorno.

juju.maas	
Deployed () Power on \vee Actions \vee Power	er cycle → Troubleshoot → Categorise →) Lock 🗇 Delete
Summary Network Storage PCI devices	USB O Commissioning O Tests O Logs Configuration
Machine configuration	Architecture
	amd64/generic
	Minimum kernel
	-
	Zone
	default
	Resource pool
	default
	Note
	-
Tags	virtual

Por último, en esta sección encontraremos información sobre la "Configuración de energía" donde indicaremos el tipo de máquina que estamos configurando. Esta herramienta ofrece integración con diversas plataformas como Proxmox, OpenStack, PDUs y IPMs entre otros . Esta nos da la °opción de encender y apagar la maquina desde MAAS .

Power configuration	Power type	Edit
	Proxmox	
	Proxmox host name or IP	
	proxmox.gonzalonazareno.org	
	Proxmox username, including realm	
	javier.cruces@iesgn	
	Proxmox password, required if a token name and secret aren't given	
	Proxmox API token name	
	-	
	Proxmox API token secret	
	-	
	Node ID	
	122	
	Verify SSL connections with system CA certificates	
	No	

4.5 Commissioning

Aunque en el apartado anterior hemos añadido la máquina a MAAS, esta solo ha pasado por un proceso preliminar más liviano para identificar su hardware. Ahora es necesario llevar a cabo el commissioning completo, que incluye una serie de pruebas más exhaustivas. Para ello, iniciaremos el proceso de commissioning completo desde la interfaz gráfica de MAAS.

Dado que hemos configurado previamente nuestro usuario de Proxmox, MAAS encenderá la máquina automáticamente cuando lancemos la orden de commissioning. Este proceso detallado permitirá a MAAS recopilar información más precisa y configurar adecuadamente el hardware, asegurando que la máquina esté completamente preparada para su uso en nuestra infraestructura.

	MAAS	juju.maas			
8	HARDWARE Machines	New ? Power unknown	 Actions ^ Commission 	Power cycle \checkmark Troubleshoot \checkmark	Categorise
	Devices	Summary Network S	Allocate Deploy	USB 🥝 Commissioning	Tests Log:
&	кум LXD	VIRTUAL MACHINE STATUS		CPU 2 COFES	amd6-
0	organisation Tags		Clone from	Test CPU	
	AZs	Owner		Zone	

A continuación nos indicara los distintos script que se van a lanzar tanto de commisioning como de test :

		Commission
竜 D	elete	Allow SSH access and prevent machine powering off
Conf	iguration	Skip configuring supported BMC controllers with a MAAS generated username and password
eneric	MEMORY 4 GIB	 Retain network configuration Commissioning scripts
	Test memory	20-maas-01-install-lldpd 20-maas-02-dhcp-unconfigured-ifaces 20-maas-03-machine-resources 30-maas-01-hmc-config
source pool » fault		40-mas-01-machine-escultes 30-mas-of-home-coming 50-mas-01-machine-config-hints
	NETWORK> Red Hat, Inc. NAME IP ADDRESS SUBJECT	maas-get-truid-api-data maas-kernet-cmdline maas-list-modaliases maas-lshw maas-serial-ports maas-support-info Select additional scripts
	ens18 Auto assigr fe:d1:b9:c9:67:ae	Testing scripts smartctl-validate × Solort additional scripts
	Information about tagged traffic can be Test network	Retain storage configuration Update firmware
		Configure HBA
		Cancel Start commissioning for machine

Mientras se ejecuta el proceso, podrás observar en esta sección cómo se actualizan progresivamente los distintos resultados a medida que se completan las pruebas y configuraciones.

juju.maas					
Deployed 😃 Power on 🗸 🛛 Actions 🗸 Po	ower cycle $ \lor $ Troubleshoot $ \lor $ Co	ategorise 🗸 💭 Lock 🗊 Delete			
Summary Network Storage PCI device	es USB Ocommissioning	Tests 🕑 Logs Configuration			
NAME	TAGS	RESULT	DATE	RUNTIME	ACTIONS
20-maas-01-install-lldpd	node	Passed	Fri, 07 Jun. 2024 16:11:18	0:00:00	~
20-maas-02-dhcp-unconfigured-ifaces	node	Passed	Fri, 07 Jun. 2024 16:11:23	0:00:05	\sim
20-maas-03-machine-resources	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:24	0:00:00	\sim
30-maas-01-bmc-config	bmc-config, node	🔺 Skipped	Fri, 07 Jun. 2024 16:11:30	0:00:00	~
40-maas-01-machine-config-hints	node	Passed	Fri, 07 Jun. 2024 16:11:30	0:00:00	\sim
50-maas-01-commissioning	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:30	0:00:00	\sim
maas-capture-lldpd	node	Passed	Fri, 07 Jun. 2024 16:12:18	0:00:46	~
maas-get-fruid-api-data	node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	~
maas-kernel-cmdline	node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	~
maas-list-modaliases	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	~
maas-lshw	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:32	0:00:01	~
maas-serial-ports	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	\sim
maas-support-info	deploy-info, node	Passed	Fri, 07 Jun. 2024 16:11:31	0:00:00	\sim

Si deseas examinar detalladamente algún resultado en particular, puedes obtener información más detallada haciendo clic en la columna de "Actions" y seleccionando "Ver detalles". Esto te permitirá acceder a una vista más completa y específica de cada etapa del proceso :

juju.maa	as.									
Deployed	ථ Power on	~ [Actions $ \smallsetminus $ I	Power cycle 🚿	Troubleshoot ✓	Categorise	· ()	Lock 🌐 Delete		
Summary	Network	Stor	age PCI devi	ces USB	Commissioning	🖉 Tests	Logs	Configuration		
maas-capl	ture-lldpd d	letail	S							< Back to test results
Status Exit status Tags			Passed 0 node					Start time End time Runtime	Fri, 07 Jun. 2024 16:11:31 Fri, 07 Jun. 2024 16:12:18 0:00:46	
Output										
combined	stdout	stder	yaml							
xml ver<br <lldp lat<="" td=""><td>rsion="1.0" bel="LLDP ne</td><td>encod eighbo</td><td>ing="UTF-8"?> rs"/></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></lldp>	rsion="1.0" bel="LLDP ne	encod eighbo	ing="UTF-8"?> rs"/>							

Adicionalmente recuerda que tienes una sección de logs para que en el caso de que haya un error leer la salida completa del mismo .

Cuando finalicen todos la maquina se apagara automáticamente y si los resultados han sido correctos podrás proceder a la instalación del sistema operativo .

4.6 Instalación del sistema operativo

Ahora vamos a elegir el sistema operativo que vamos a desplegar , asi que para ello vamos a Actions > Deploy , ya que una vez hecho el paso anterior de commisioning podremos desplegar nuestro sistema .

			Release
✓ ⊕ Delete			Ubuntu 20.04 LTS "Focal Fossa" 🗸
Logs C		-	Kernel
Logs			No minimum kernel 🗸
amd64/generic	MEMORY		
	4 GIB		Customise options
	Test memory		Register as MAAS KVM host. KVM docs
Resource	pool>		Only Ubuntu 18.04 LTS and Ubuntu 20.04 LTS are officially supported
default			Cloud-init user-data Cloud-init docs
	NETWORK >	-	→ Upload script
	Red Hat, Inc.		YsSwCcc0qTG1bjWLp8ZHr3/z40/ejPCpqtVyeKgLHpi05yveMm
	NAME MAC ADDRESS	IP ADDRESS SUBNET	CgcWricETTw4WVvSSJMoochUJS1siLbPr1S1vu1uH1a9B63rT
	ens18	Auto assigr	7F8I5Riainm5XhZnx/8PKtvVyuAlmTV6uuw5jZqAw==
	fe:d1:b9:c9:67:ae	192.168.77.0	javierasping@gmail.com shell: /bin/bash
	Information about tagge	ed traffic can be	sudo: ALL=(ALL:ALL) NOPASSWD:ALL
	Test network		fqdn: juju-fjcd.es

A continuación le indicare un fichero cloud-init.yaml para que la maquina se configure acorde al mismo , en mi caso el fichero tiene la siguiente configuración :

- Creara el usuario javiercruces con una contraseña hasheada
- Pertenecerá al grupo sudo y no pedirá contraseñas
- La maquina tendrá un FQDN juju-fjcd.es
- Instalara el paquete snap

Aquí te dejo el fichero que he utilizado :

! cloud-ir	iit-config.yaml ×
	viercruces > Documentos > 2ºASIR > SRI > PRIMER TRIMESTRE > openstack > 🕴 cloud-init-config.yaml
	cloud-config
	ackage_upgrade: true
	vackages:
	- snap
	- name: javiercruces
	passwd: \$6\$0mzQBNDo.VzQ6d3B\$Q0c.FeToLuBo/TRYFhpx.wjM9CsMA392ntCFCCU6Acx5pFR7DCrlxIJhqMUeUsvfFZhCzgT1/fGeOwtgoNo3e1
	- ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQC7D77owxzTExSK+jymXOYGUn5GrjB1SPvOh8lJHM29WØKViIFJzumfHFvwfGPcDgVR8DyrZYHTWAmyH2E19Sw
	shell: /bin/bash
	sudo: ALL=(ALL:ALL) NOPASSWD:ALL
17 f	ʻqdn: juju-fjcd.es

MAAS METAL AS A SERVICE

Francisco Javier Cruces Doval

Al igual que en pasos anteriores, solo tendremos que esperar. Una vez que iniciemos la instalación, la máquina se encenderá automáticamente gracias a la configuración previa de nuestro usuario de Proxmox en MAAS en la sección de energía de la máquina.

La máquina arrancará mediante PXE e instalará Ubuntu 22.04, el sistema operativo que le hemos indicado. A continuación, se procederá con la configuración especificada en nuestro archivo cloudinit.yaml, asegurando que todos los ajustes y parámetros necesarios se apliquen correctamente durante la instalación. Este proceso automatizado facilita enormemente la configuración inicial y garantiza que la máquina esté lista para su uso con la mínima intervención manual.

Una vez finalizado el estado de la maquina habrá cambiado a desplegado :

Deployed O Power on \checkmark Actions \checkmark	Power cycle 🗸	Troubleshoot 🗸	Categorise 🗸	O Loo	ck
Summary Network Storage PCI de	vices USB	Commissioning	🕑 Tests 🛛 🖉	Logs	Configuration
VIRTUAL MACHINE STATUS	CPU		amd64	/generic	MEMORY
Deployed	2 cores	2 cores			4 GiB
Ubuntu 22.04 LTS "Jammy Jellyfish"	Common	KVM processor			
	Test CPL	J			Test memory
Owner	Zone>			Resource	pool>

4.7 Comprobación de la instalación

Vamos a probar la nueva instalación que hemos realizado. Lo primero será obtener la IP de nuestra máquina. Esta información se puede consultar directamente en nuestro MAAS, dado que esta máquina también actúa como servidor DHCP y almacena los datos correspondientes.

NETWORK					
Red Hat, Inc.		1			
NAME MAC ADDRESS	IP ADDRESS SUBNET	LINK SPEED	FABRIC ① VLAN	DHCP	SR-IOV
ens18 fe:d1:b9:c9:67:ae	192.168.77.4 192.168.77.0/24	0 Mbps	fabric-1 Default VLAN	No DHCP	No
Information about tagge Test network	ed traffic can be seer	n in the Network t	ab.		

Como esta maquina esta en una red interna me tendré que conectar desde la maquina MAAS, para ello previamente me he conectado a esta utilizando el parámetro -A para llevarme mi clave privada :

javiercruces@maas-fjcd:~\$ ssh javie	ercruces@192.168.77.4	
Welcome to Ubuntu 22.04.4 LTS (GNU	/Linux 5.15.0-112-generi	x86_64)
* Documentation: https://help.ubu	untu.com	
* Management: https://landsca	pe.canonical.com	
* Support: https://ubuntu.o	com/pro	
System information as of Sat Jun	8 23:40:27 UTC 2024	
System load: 0.0	Processes:	97
Usage of /: 14.1% of 48.91GB	Users logged in:	0
Memory usage: 5% Swap usage: 0%	IPv4 address for ens18:	192.168.77.4
Shap asager on		
* Strictly confined Kubernetes main just raised the bar for easy, re	kes edge and IoT secure. esilient and secure K8s o	Learn how MicroK8s cluster deployment.
https://ubuntu.com/engage/secure	e-kubernetes-at-the-edge	
Expanded Security Maintenance for ,	Applications is not enab	led.
0 updates can be applied immediate	ly.	
Enable ESM Apps to receive addition See https://ubuntu.com/esm or run:	nal future security upda sudo pro status	tes.
Last login: Sat Jun 8 21:44:55 20: javiercruces@juju-fjcd:~\$	24 from 192.168.77.1	

Vamos a comprobar que se ha aplicado la configuración del fichero cloud-init.yaml.

• El usuario se ha creado y se ha añadido la clave publica indicada :

javiercruces@juju-fjcd:~\$ cat .ssh/authorized_keys ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQC7D77owxzTExSK+jymX0YGUn5GrjB1SPv0h81 ejLtmQIGILjNMN7glFEmzY7qswb94/06R9A00aTh0+2JB+C3ouzSc41rD8xqhEsszjCjjzZXuU5 WrLp3/dJHtNA6eVBkQnw0XzHE/y+3wcX6SuHd63AVtcstdGNn60Zp5eJdIGE5e39YsSwCcc0qTG

• Podemos usar sudo sin contraseña :

javiercruces@juju-fjcd:~\$ sudo su
root@juju-fjcd:/home/javiercruces#

zfIAMdoEPr2IUG9JWbKmrQAvTs3rJkkT9VAh6mzoUoILwSK77F8I5Riainm5XhZnx/8PKtvVyuA

• Se ha instalado el paquete snap

```
javiercruces@juju-fjcd:~$ whereis snap
snap: /usr/bin/snap /usr/lib/snap /usr/share/snap /usr/share/man/man8/snap.8.gz
javiercruces@juju-fjcd:~$
```

Con esto hemos comprobado de que se ha ejecutado correctamente tanto la instalación como la configuración de la maquina haciendo uso de un fichero cloud-init .

5. Despliegue de cluster de k3s con MAAS y Ansible

5.1 Introducción

A continuación, vamos a realizar la instalación de un clúster de Kubernetes, específicamente de k3s. Este clúster estará compuesto por 4 nodos, los cuales crearemos en Proxmox. Todos los nodos compartirán las mismas especificaciones. Desplegaremos el sistema operativo en cada nodo utilizando MAAS. Durante esta configuración, se establecerá un nombre de usuario con mi clave pública, se actualizarán los paquetes y se asignará un nombre FQDN personalizado a cada nodo.

Posteriormente, utilizando la herramienta Ansible, instalaremos k3s en los 4 nodos. Para verificar que el clúster se ha creado correctamente, utilizaremos kubectl para comprobar que todos los nodos pertenecen al mismo clúster.

Para finalizar, desplegaremos una aplicación de prueba para asegurarnos de la funcionalidad del mismo , además crearemos las reglas de iptables necesarias para poder acceder a esta aplicación desde la red del instituto ya que los nodos pertenecen a una red privada a la que solo la maquina de MAAS esta conectada .

Puedes encontrar este apartado grabado en vídeo pulsando aquí .

5.2 Creación de las maquinas en Proxmox

Lo primero será definir la creación de estos 4 nodos. Dado que no disponemos de una infraestructura física, utilizaremos máquinas virtuales. Para evitar repeticiones, a continuación mostrare las características de una única máquina, ya que todas serán idénticas.

Cada nodo contará con 4 GB de RAM, 32 GB de disco y 2 núcleos de CPU.

Virtual Machine 122 (nodo1) on node 'proxmox1' No Tags 🖋					
🗐 Summary	Add V Remove Edit	Disk Action V Revert			
>_ Console	📟 Memory	4.00 GiB			
🖵 Hardware	Processors	2 (1 sockets, 2 cores)			
📥 Cloud-Init	BIOS	Default (SeaBIOS)			
Options	🖵 Display	Default			
Task History	🕫 Machine	Default (i440fx)			
	SCSI Controller	VirtIO SCSI single			
Monitor	OD/DVD Drive (ide2)	none,media=cdrom			
🖺 Backup	🖨 Hard Disk (virtio0)	local-lvm:vm-122-disk-0,iothread=1,size=32G			
🔁 Replication		virtio=0E:53:E3:27:E6:42,bridge=vmbr210,firewall=1			

A la hora de definirlas no le indicaremos ningún medio de instalación , ya que lo haremos a través de la red .

Esto mismo haremos para tener preparado nuestros 4 nodos que utilizaremos a lo largo de este apartado .



5.3 Añadir los nodos a MAAS

Una vez creadas nuestras máquinas virtuales, el primer paso será encenderlas para que nuestro servidor MAAS las agregue a su inventario. Inicialmente, estas máquinas tendrán un nombre generado aleatoriamente por MAAS, por lo que tendremos que cambiarles el nombre manualmente para mantener un control preciso de las mismas. Podemos distinguir unas de otras mediante sus direcciones MAC.

Finalmente mi inventario a quedado así :

8	HARDWARE					
	Machines	Showing 4 out of 4 machines				
	Devices		POWER	STATUS	OWNER I NAME	ROOL
	Controllers	IP	TOWER	SIAIOS	TAGS	NOTE
&	кум	Deployed				
	LXD	4 machines				
0	ORGANISATION	nodo1.maas	() On	Ubuntu 22.04 LTS	javiercruces	default
	Tags	192.168.77.19 (PXE)	Proxmox		virtual	
	AZs	nodo2.maas	() On	Ubuntu 22.04 LTS	javiercruces	default
	Pools	192.168.77.20 (PXE)	Proxmox		virtual	
.8	CONFIGURATION	nodo3.maas	() On	Ubuntu 22.04 LTS	javiercruces	default
	Images	192.168.77.21 (PXE)	Proxmox		virtual	
	NETWORKING	nodo4.maas	() On	Ubuntu 22.04 LTS	javiercruces	default
	Subnets	192.168.77.22 (PXE)	Proxmox		virtual	

Una vez que tengamos las máquinas identificadas, es importante configurar el tipo de máquina en MAAS. MAAS ofrece una serie de integraciones que permiten manipular las máquinas (encenderlas o apagarlas), lo cual es necesario para algunos procesos en MAAS. Para realizar esta configuración, accederemos a la configuración individual de cada máquina y seleccionaremos el apartado de energía. Aquí podremos establecer los parámetros necesarios para controlar la máquina .

nodo1.m	naas										
Deployed	ථ Power off	~	Actions 🗸	Pow	er cycle 🗸	Troubleshoot $ \smallsetminus $	Categorise	\sim	0	Lock	觉 Delete
Summary	Network	Stor	age PCI o	evices	USB	Commissioning	Tests	0	Logs	Conf	iguration

MAAS METAL AS A SERVICE

Francisco Javier Cruces Doval

Una vez en esta sección accedemos a la configuración de energía y configuraremos nuestra maquina acorde al escenario que tengamos . En mi caso como he comentado anteriormente se trata de maquinas en Proxmox . Aquí dejo el ejemplo del nodo 1 .

Power configuration	Power type
	Proxmox
	Proxmox host name or IP
	proxmox.gonzalonazareno.org
	Proxmox username, including realm
	javier.cruces@iesgn
	Proxmox password, required if a token name and secret aren't given

	Proxmox API token name
	_
	Proxmox API token secret
	Node ID
	122
	Verify SSL connections with system CA certificates
	No

Esto lo haremos para que cuando lancemos una orden en MAAS , este sea capaz de encender la maquina y apagarla cuando sea necesario .

Una vez hayamos configurado esto en nuestros 4 nodos podremos continuar al siguiente paso.

5.4 Commisioning

En este paso se realizara un inventario exhaustivo de cada uno de nuestros nodos en el cual aparecerán todas las características de nuestras maquinas y se realizaran los test que tengamos configurados en MAAS.

Este paso es crucial ya que es el anterior a realizar una instalación y nos aseguramos de que la maquina a nivel de hardware esta operativa para entrar en producción .

Si quieres lanzar las 4 maquinas a la vez a este proceso, selecciónalas desde la interfaz gráfica :

~	nodo1 .maas 192.168.77.19 (PXE)	ு	Off Proxmox
	nodo2 .maas 192.168.77.20 (PXE)	ு	Off Proxmox
	nodo3 .maas 192.168.77.21 (PXE)	ு	Off Proxmox
~	nodo4.maas	Ċ	Off

Y en la parte superior en acciones haz clic sobre commision :

	Actions ^	Po	ower cycle 🗸	Troublest	noot ~
	Commission				
	Allocate				
	Deploy				
STATUS	Release		R NAME	POOL	ZO
	Abort			NOTE	SP
	Clone from				

Una vez lanzado el proceso las 4 maquinas arrancara por PXE y se ejecutaran los scripts de commisioning y test correspondientes .

Ready 5 machines			
nodo1 .maas	じ Off	Ready	-
192.168.77.120 (PXE) (+1)	Proxmox		virtual
nodo2 .maas	じ Off	Ready	-
192.168.77.122 (PXE) (+1)	Proxmox		virtual
nodo3 .maas	じ Off	Ready	-
192.168.77.124 (PXE) (+1)	Proxmox		virtual
nodo4 .maas	じ Off	Ready	-
192.168.77.125 (PXE) (+1)	Proxmox		virtual

Cuando estos finalicen el estado de las maquinas cambiara a ready :

Y si accedemos a cualquiera de ellas , veremos que han pasado satisfactoriamente todos los test :

nodo1.maas Ready () Power off ~ Actions ~ Trout	oleshoot ~ Categorise ~ 音 Del	ete			
Summary Network Storage PCI device	s USB O Commissioning	Tests Logs Configuration			
NAME	TAGS	RESULT	DATE	RUNTIME	ACTIONS
20-maas-01-install-lldpd	node	Passed	Wed, 12 Jun. 2024 13:46:53	0:00:00	\sim
20-maas-02-dhcp-unconfigured-ifaces	node	Passed	Wed, 12 Jun. 2024 13:46:59	0:00:05	\sim
20-maas-03-machine-resources	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:46:59	0:00:00	\sim
30-maas-01-bmc-config	bmc-config, node	A Skipped	Wed, 12 Jun. 2024 13:47:06	0:00:00	\checkmark
40-maas-01-machine-config-hints	node	Passed	Wed, 12 Jun. 2024 13:47:06	0:00:00	\sim
50-maas-01-commissioning	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:47:07	0:00:00	~
maas-capture-lldpd	node	Passed	Wed, 12 Jun. 2024 13:47:53	0:00:45	\sim
maas-get-fruid-api-data	node	Passed	Wed, 12 Jun. 2024 13:47:08	0:00:00	\sim
maas-kernel-cmdline	node	Passed	Wed, 12 Jun. 2024 13:47:07	0:00:00	\sim
maas-list-modaliases	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:47:09	0:00:00	\sim
maas-lshw	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:47:09	0:00:01	\sim
maas-serial-ports	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:47:08	0:00:00	~
maas-support-info	deploy-info, node	Passed	Wed, 12 Jun. 2024 13:47:08	0:00:00	\sim

Con este paso completado ya tenemos la certeza de que nuestra maquina esta preparada para instalarle un sistema operativo .

5.5 Instalación personalizada con cloud-init

Vamos a comenzar la instalación del sistema operativo en los cuatro nodos, optando por Ubuntu 22.04. Utilizaremos un archivo de configuración cloud-init para cada máquina, personalizando el FQDN para adaptarlo a cada nodo.

El archivo cloud-init que utilizaremos incluye la actualización de paquetes, la instalación de snap, la creación de un usuario con privilegios de sudo y la configuración de las claves SSH. Para este ejemplo, lo más importante es tener un usuario con privilegios sin necesidad de contraseña al utilizar sudo.

Aquí dejo de ejemplo el fichero cloud-init.yaml del nodo1 , que es idéntico a los otros 3 cambiando el valor del FQDN :



Para desplegar la instalación , puesto que vamos a utilizar ficheros distintos tendremos que ir maquina por maquina indicándole el fichero de cada una .

En la sección de despliegue seleccionamos que queremos un Ubuntu 22.04, no es necesario que especifiquemos una versión mínima del kernel.

OS	
Ubuntu	\sim
Release	
Ubuntu 22.04 LTS "Jammy Jellyfish"	~

En la sección inferior de este menú, especificamos que queremos subir un archivo cloud-init y seleccionamos el correspondiente para cada nodo. Utilizo un archivo base como plantilla y, desde este menú, modifico el valor del FQDN para cada nodo individualmente , ya que todos comparten el mismo fichero a excepcionan de este valor :

Customise options
 Register as MAAS KVM host. KVM docs Only Ubuntu 18.04 LTS and Ubuntu 20.04 LTS are officially supported.
Cloud-init user-data Cloud-init docs
→ Upload script
#cloud-config
package_upgrade: true
packages: - snap
users: - name: javiercruces lock_passwd: false
Periodically sync hardware ③ Hardware sync docs Hardware sync interval: 15 minutes - Admins can change this in the global settings.
Cancel Start deployment for machine

Una vez configuradas las máquinas con el archivo cloud-init adecuado, procederemos con el despliegue. Cada máquina se iniciará y arrancará por red, comenzando así el proceso de instalación del sistema operativo.

Este proceso lo repetiremos para los 4 nodos.

Una vez el despliegue de las mismas haya finalizado el estado en el panel de MAAS pasara a ser deployed :

FQDN ^ MAC	POWER	STATUS	OWNER NAME TAGS
Deployed 4 machines			
nodo1 .maas	On	Ubuntu 22.04 LTS	javiercruces
192.168.77.19 (PXE)	Proxmox		virtual
nodo2.maas	U On	Ubuntu 22.04 LTS	javiercruces
192.168.77.20 (PXE)	Proxmox		virtual
nodo3.maas	ပံ On	Ubuntu 22.04 LTS	javiercruces
192.168.77.21 (PXE)	Proxmox		virtual
nodo4 .maas	ひ On	Ubuntu 22.04 LTS	javiercruces
192.168.77.22 (PXE)	Proxmox		virtual

Una vez desplegados nuestros sistemas operativos ya tendremos nuestras maquinas operativas para poder trabajar con ellas .

5.6 Instalación de k3s con Ansible

Una vez completada la instalación de Ubuntu 22.04 en los 4 nodos, es el momento de desplegar nuestras aplicaciones, específicamente k3s. El primer paso será configurar Ansible en el servidor MAAS, desde donde ejecutaremos nuestro playbook.

Para instalarnos Ansible utilizamos el siguiente comando :

```
sudo apt install ansible
```

A continuación nos clonaremos el repositorio donde contiene la receta de ansible que vamos a desplegar :

```
git clone https://github.com/k3s-io/k3s-ansible.git
```

Nos moveremos dentro del repositorio clonado y crearemos el fichero de inventario aprovechando la plantilla que tenemos :

cp inventory-sample.yml inventory.yml

Ahora sobre el fichero inventory modificaremos las Ips y añadiremos la de nuestros nodos , estas las puedes encontrar desde el panel de MAAS :

```
javiercruces@maas-fjcd:~/k3s-ansible$ cat inventory.yml
.---
k3s_cluster:
    children:
        server:
        hosts:
           192.168.77.19:
        agent:
        hosts:
           192.168.77.20:
           192.168.77.21:
           192.168.77.22:
```

En este caso habrá un nodo que sera controlador de vuelo y los demás serán workers .

En el mismo fichero configuraremos el usuario que tiene que utilizar Ansible para conectarse a las distintas maquinas , así como su puerto :

```
# Required Vars
vars:
   ansible_port: 22
   ansible_user: javiercruces
   k3s_version: v1.26.9+k3s1
```

Por ultimo en este fichero es necesario configurar un token , el propio fichero nos indica dos posibles comandos para obtener uno de forma aleatoria :



Una vez configurado el fichero inventory.yaml , vamos a comprobar si tenemos conectividad con todos los nodos lanzándoles un ping :

```
javiercruces@maas-fjcd:~/k3s-ansible$ ansible -m ping all
[WARNING]: Platform linux on host 192.168.77.22 is using the discovered
See https://docs.ansible.com/ansible-core/2.17/reference_appendices/inte
192.168.77.22 | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3.10"
    },
    "changed": false,
    "ping": "pong"
}
```

Una vez tenemos conectividad con todos los nodos procederemos a lanzar el playbook ansible-playbook playbook/site.yml -i inventory.yml Esto tardara un poco , cuando finalice tenemos que asegurarnos que tenemos una salida similar a esta :

LAT RECAP							
192.168.77.19	: ok=25	changed=13	unreachable=0	failed=0	skipped=52	rescued=0	ignored=2
192.168.77.20	: ok=18	changed=7	unreachable=0	failed=0	skipped=40	rescued=0	ignored=1
92.168.77.21	: ok=18	changed=7	unreachable=0	failed=0	skipped=40	rescued=0	ignored=1
192.168.77.22	: ok=18	changed=7	unreachable=0	failed=0	skipped=40	rescued=0	ignored=1

Como vemos que la ejecución ha sido exitosa , vamos a acceder al nodo 1 que es el controlador de vuelo y comprobaremos que tenemos cuatro nodos en el cluster de kubernetes .

NAME	STATUS	ROLES	AGE	VERSION
nodo1	Ready	control-plane,master	80s	v1.26.9+k3s1
nodo2	Ready	<none></none>	19s	v1.26.9+k3s1
nodo4	Ready	<none></none>	19s	v1.26.9+k3s1
nodo3	Ready	<none></none>	19s	v1.26.9+k3s1
javier	ruces@noo	do1:~\$	100	1112010 100

La instalación ha sido todo un éxito, como podemos ver el cluster se ha formado por cuatro nodos.

5.7 Despliegue de la aplicación de encuestas

Para comprobar la funcionalidad de nuestro cluster de kubernetes vamos a desplegar la famosa aplicación de encuestas , para ello utilizare los ficheros de la practica de kubernetes .

Los cuales puedes encontrar los ficheros necesarios en mi repositorio de GitHub , a continuación te dejo el enlace \rightarrow <u>https://github.com/javierasping/PracticaK8S</u>.

Los pasos para desplegar esta aplicación son sencillos , simplemente nos clonamos el repositorio anterior con el siguiente comando :

```
git clone <a href="https://github.com/javierasping/PracticaK8S.git">https://github.com/javierasping/PracticaK8S.git</a>
```

La aplicación está compuesta por un servidor web que ejecuta Django, una base de datos MariaDB, un volumen persistente para almacenar los datos, dos servicios para facilitar la comunicación entre los contenedores (Django y MariaDB), y finalmente un Ingress para permitir el acceso al servidor web desde el exterior.

Una vez conocemos que hace cada uno de estos ficheros vamos a desplegar la aplicación con los siguientes comandos :

kubectl	apply	-f	pvc-mariadb.yaml
kubectl	apply	-f	deployment-mariadb.yaml
kubectl	apply	-f	<pre>srv-mariadb.yaml</pre>
kubectl	apply	-f	deployment-django.yaml
kubectl	apply	-f	srv-django-web.yaml
kubectl	apply	-f	ingress.yaml

Este seria el orden correcto para desplegar la aplicación correctamente .

Vamos a asegurarnos de que esta se ha desplegado correctamente en todos los nodos con el siguiente comando :

iaviercruces@nodo1:~/PracticaK8S\$ kub	ectl de	t all	-o wide										
NAME	R	EADY	STATUS	RESTA	RTS AGE	IP	NO	DE	NOMIN	ATED NODE	READINESS GATES		
pod/django-tutorial-web-57859bc87f-dz	v2s 1	/1	Running	0	5s	10.4	42.2.4 no	do3	<none< td=""><td>></td><td><none></none></td><td></td><td></td></none<>	>	<none></none>		
pod/mariadb-django-6f95f6c74c-8t6z2	1	/1	Running	0	5s	10.4	42.3.8 no	do2	<none< td=""><td>></td><td><none></none></td><td></td><td></td></none<>	>	<none></none>		
NAME	TYPE		CLUSTER-I	P	EXTERNAL-I	P P(DRT(S)	AG	E	SELECTOR			
service/kubernetes	Cluste	rIP	10.43.0.1		<none></none>	44	43/TCP	6m-	45s -	<none></none>			
service/django-tutorial-web-service	NodePo	rt	10.43.197	.235	<none></none>	80	0:32578/TCF	5s		app=django-	tutorial-web		
service/mariadb-django	Cluste	rIP	10.43.27.	237	<none></none>	33	306/TCP	5s	S	app=mariadb	-django		
NAME	READY	IIP-1	TO-DATE		E AGE	CONT	ATNERS		TMAG	FS		SELEC	TOP
deployment apps/diange_tutevial.ueb	1/1	1	IO DATE	1	Ee Ade	dian	an tutorial	uch	d aud	avacaina (di	anna tutorial is	apped	linnes tuterial uch
deproyment.apps/django-cucorrar-web	1/1	1		1	35	ujang	go-cucorial	-web	Javi	eraspring/uj	ango_cucoriai_ic	app=0	IJango-tutorrar-web
deployment.apps/mariadb-django	1/1	1		1	5s	maria	adb-django		mari	adb		app=m	ariadb-django
NAME		1	DESIRED	CURRENT	READY	AGE	CONTAINER	2S		IMAGES			SELECTOR
replicaset.apps/django-tutorial-web-5	7859bc8	7f 1	1	1	1	5s	django-tu	toria	l-web	javierasp	ing/django_tutoria	l ic	app=django-tutorial-web,pod-template-hash=57859bc87f
replicaset.apps/mariadb-django-6f95f6	c74c	1	1	1	1	5s	mariadb-d	ijango		mariadb			app=mariadb-django,pod-template-hash=6f95f6c74c
javiercruces@nodo1:~/PracticaK8S\$													

Como vemos el despliegue de la aplicación en el cluster parece haber sido exitosa , ahora vamos a configurar una regla de DNAT en la maquina de MAAS que apunte al nodo1 para poder ver la aplicación . Si nos fijamos en la captura de pantalla anterior el puerto en el que el servidor web esta sirviendo la aplicación es el 32578 .

Así quedaría la regla en cuestión :

javiercruces@maas-fjcd:~\$ sudo iptables -t nat -A PREROUTING -p tcp --dport 32578 -j DNAT --to-destination 192.168.77.19:32578

Una vez aplicada , podremos acceder a la IP de nuestra maquina MAAS y ver la aplicación de encuestas :



Para asegurarnos de que la aplicación funciona correctamente voy a crear una pregunta en la misma :

← → C ▲ No es seguro 172.22.6				🖈 🖸 🛛 🗐 Error 🗄
🕦 WhatsApp @ PLEDIN 3.0 🔥 Moodle	e centros 🛛 🧑 Redmine IES 🗧 Vista gen	eral d 🔞 APUNTES 💠 SLAAC 🛚 MATES.SCHO	OL 🛭 in Moodle Incibe 👼 Epic Games St 🛥 Prime Gaming 🛚 S Inic	iar sesión: » 🗅 Todos los marcadores
Django administration				
Home > Polls > Questions				
Start typing to filter				
AUTHENTICATION AND AUTHORIZATION				
Groups + Add				ADD QUESTION +
Users + Add				FILTER
		Search		+ By date published
POLLS		Call O of 2 colorted		Any date
Questions + Add	Action			Today
	QUESTION TEXT	DATE PUBLISHED	PUBLISHED RECENTLY?	Past 7 days
	12 junio	June 12, 2024, 3:36 p.m.		This year
	Hola	June 12, 2024, 2:58 p.m.		
	2 questions			

Voy a comprobar de que puedo contestar a la misma , si accedo a la sección de polls :



Vote again?

Con esta serie de pruebas hemos comprobado que el cluster esta operativo y que la aplicación es totalmente funcional en el mismo .

6. Conclusiones

MAAS es una herramienta muy útil en sitios donde el parque informático es muy grande ,como por ejemplo en centros de datos , ya que nos permite ahorrar una infinidad de tiempo cuando realizamos instalaciones , además nos da una gran ventaja ya que no tenemos que "preocuparnos" de preparar imágenes personalizadas al poder utilizar cloud-init en un entorno físico real .

Gracias a esto nos permite tener una serie de plantillas la cual podemos editar fácilmente y personalizar la instalación a nuestro gusto sin depender de una imagen completa para realizar esta tarea.

Aunque nos cubra la instalación del sistema operativo , esta herramienta se vuelve mas útil si la combinas con herramientas de orquestación para gestionar el software en estos sistemas. Para implementar soluciones complejas como OpenStack o un clúster de Kubernetes, se utilizan herramientas de orquestación como Juju, también de Canonical, o Ansible.

Esta herramienta mejoraría si incluyera de forma nativa la orquestación de software , ya que tendríamos todo automatizado de mano de la misma herramienta lo que simplificaría el proceso .

7. Bibliografía

- <u>Página oficial de MAAS</u>
- Imágenes personalizadas en MAAS con packer
- Integración de Juju y MAAS
- <u>Receta de Ansible para desplegar k3s</u>
- <u>Repositorio con los ficheros para el despliegue de la aplicación de encuestas</u>