

Documentación – Despliegue y administración de una IT basada en CMS - GAR

ALBERTO MORENO GÓMEZ

alberto.moreno15@alu.uclm.es

ALEJANDRO DURAN DE LA
MORENA

alejandro.duran@alu.uclm.es

DAVID MUÑOZ ESCRIBANO

david.munoz19@alu.uclm.es

MARIO SERRANO ROMERO

mario.serrano7@alu.uclm.es

Índice

Variables del sistema.....	3
Tabla resumen de encaminamiento	5
Instrucciones para el despliegue.....	5
Nodos balanceadores de carga	6
Nodos CMS.....	7
Nodos clustering.....	7
Master k8s	7
Slave k8s	7
Monitorización del sistema	7
Ampliar o sustituir nodos	8
Puestos host-desk	8
Nodos CMS.....	8
Nodos BBDD	8
Maqueta redes con packettracer	8

Índice de tablas

Tabla 1. Valores nodo inicial	3
Tabla 2. Valores nodo balanceador de carga.....	3
Tabla 3. Valores nodos CMS	4
Tabla 4. Valores nodos clustering	5
Tabla 5. Resumen de los valores de los nodos	5

Variables del sistema

Nodo 0 (nodo inicial)				
Nodo	Parámetros	Valor	Significado	Observaciones
Nodo 0	IP del servidor	10.0.0.10	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	
	Archivo de configuración de arranque	-	Archivo de configuración para la instalación	
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	PXE, Puppet		Puppet (master)

Tabla 1. Valores nodo inicial

Nodo 1 (Balanceado de carga)				
Nodo	Parámetros	Valor	Significado	Observaciones
Nodo 1	IP del servidor	Dinámico	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	
	Archivo de configuración de arranque		Archivo de configuración para la instalación	Instrucciones de instalación de las herramientas necesarias
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	Ha-proxy, Puppet		Puppet (agent)

Tabla 2. Valores nodo balanceador de carga

Nodos 2 y 3 (CMS)				
Nodo	Parámetros	Valor	Significado	Observaciones
Nodo 2	IP del servidor	Dinámico	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	

	Archivo de configuración de arranque		Archivo de configuración para la instalación	Instrucciones de instalación de las herramientas necesarias
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	Wordpress, Puppet		Puppet (agent)
Nodo 3	IP del servidor	Dinámica	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	
	Archivo de configuración de arranque		Archivo de configuración para la instalación	Instrucciones de instalación de las herramientas necesarias
	SELinux	Habilitado	Otorgará a los administradores más control sobre quién se puede acceder	
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	Wordpress, Puppet		Puppet (agent)

Tabla 3. Valores nodos CMS

Nodos 4 y 5 (Clustering)				
Nodo	Parámetros	Valor	Significado	Observaciones
Nodo 4	IP del servidor	Dinámico	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	
	Archivo de configuración de arranque		Archivo de configuración para la instalación	Tendrá el rol de <i>master</i> en k8s
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	Kubernetes, Puppet		Puppet (agent)

Nodo 5	IP del servidor	Dinámico	Dirección IP que le será asignada al servidor	
	SO	Ubuntu 18.04	El sistema operativo que será utilizado	
	Archivo de configuración de arranque	anaconda-ks-k8s-slave.cfg	Archivo de configuración para la instalación	Tendrá el rol de <i>slave</i> en k8s
	Cortafuegos	Habilitado	Bloqueará accesos no autorizados	
	Servicios instalados	Kubernetes, Puppet		Puppet (agent)

Tabla 4. Valores nodos clustering

Tabla resumen de encaminamiento

Nodo	Dirección IP	MACs	Propósito
Main (10.0.0.0/24)			
Nodo 0	10.0.0.10	-	Aprovisionar
Nodo 1	Dinámica (DHCP)	080027CF4988	Balanceo Carga
Nodo 2	Dinámica (DHCP)	08002783067B	LAMP/CMS
Nodo 3	Dinámica (DHCP)	08002798E599	LAMP/CMS
Internal (10.0.1.0/24)			
Nodo 4	Dinámica (DHCP)	0800276705D1	Clustering
Nodo 5	Dinámica (DHCP)	08002723FB48	Clustering

Tabla 5. Resumen de los valores de los nodos

***En todos los nodos se instalará puppet junto a los servicios correspondientes de cada uno, para cumplir el propósito de cada uno de los nodos.**

Instrucciones para el despliegue

Para llevar a cabo el despliegue se debe de tener de un nodo inicial con TFTP y PXE para instalar desatendidamente los otros nodos ya que esto será lo que nos permitirá el aprovisionamiento de cada uno de los nodos. Para tener todo configurado, recomendamos importar la máquina virtual que se encuentra en el repositorio de la práctica, la cual lleva todo lo que he mencionado anteriormente para el correcto funcionamiento.

Sugerimos que se siga el orden de esta sección del documento para llevar a cabo el despliegue e ir creando cada uno de los nodos. De manera general, todos los nodos necesitamos configurarlos para administrarles el sistema operativo y para eso tenemos un archivo llamado *preseed.cfg* en la ruta */var/www/html/* y dependiendo de los servicios y prestaciones que queramos configurar nos seleccionará un archivo puppet que tendrá diferentes configuraciones específicas y ese archivo se selecciona de manera automática según la MAC.

Es importante tener en cuenta que, para todos los tipos de nodos se realizará una configuración muy similar. Ahora se detallarán configuraciones que deben complementar a la general en caso de que los servicios incorporados en los distintos tipos de nodos así lo requieran.

La instalación desatendida de cada uno de los nodos varía según el archivo que utilicemos como hemos explicado antes, que todos los archivos están en esa ruta que hemos mencionado. La instalación desatendida general y para todos los nodos es la siguiente:

- **Modificar el archivo de configuración de DHCP:** En el archivo *dhcpd.cfg* que se encuentra en */etc/dhcp/* lo que deberemos hacer es asignar en función de la interfaz de red una dirección IP correspondiente a la red a la que pertenezca el nodo. Para crear diferentes redes deberemos crear diferentes pools de direcciones, en nuestro caso habrá 2, 10.0.0.0/24 y la 10.0.1.0/24

- **Configuración apache:** Hemos configurado un servidor apache para hacer accesibles los archivos de configuración y el archivo *preseed*, mencionado anteriormente, a la hora de la instalación.

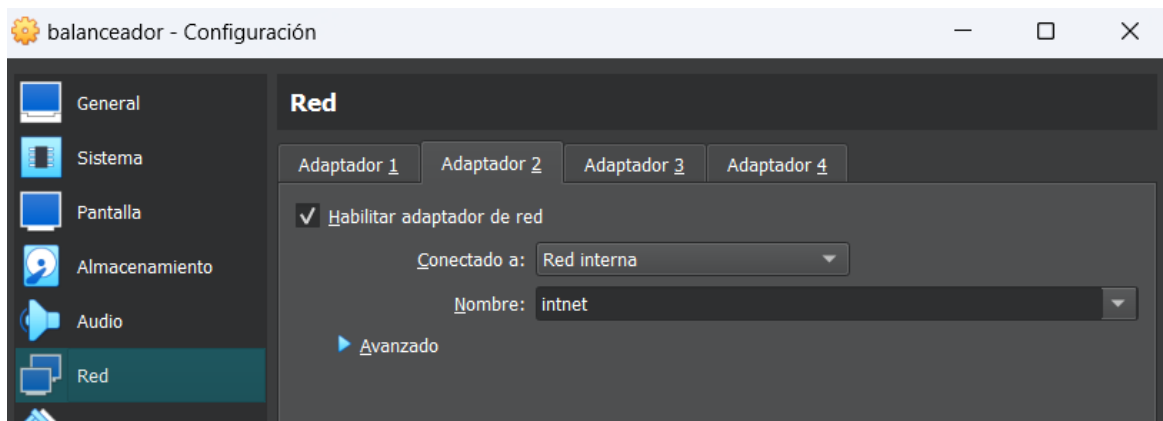
- **Configurar TFTP:** Para todos los nodos existe un solo archivo llamado *txt.cfg* con el cual podrá instalar de forma personalizada la imagen del sistema operativo con la configuración que se desea (*/var/lib/tftpboot/Ubuntu-installer/amd64/boot-screens*), aparte en este directorio también configuramos el archivo *syslinux.cfg*. En estos archivos hemos realizado una serie de cambios que nos permitían completar la instalación automática, por ejemplo, hemos añadido un timeout de 5s para que se seleccione por defecto la **instalación desatendida**.

- **Configuración de las interfaces de red:** Para permitir el *forwarding*, que lo que hace es permitirnos la conexión a internet desde los diferentes nodos (pasando todos por el nodo 0), hemos tenido que realizar una serie de cambios que son los siguientes:

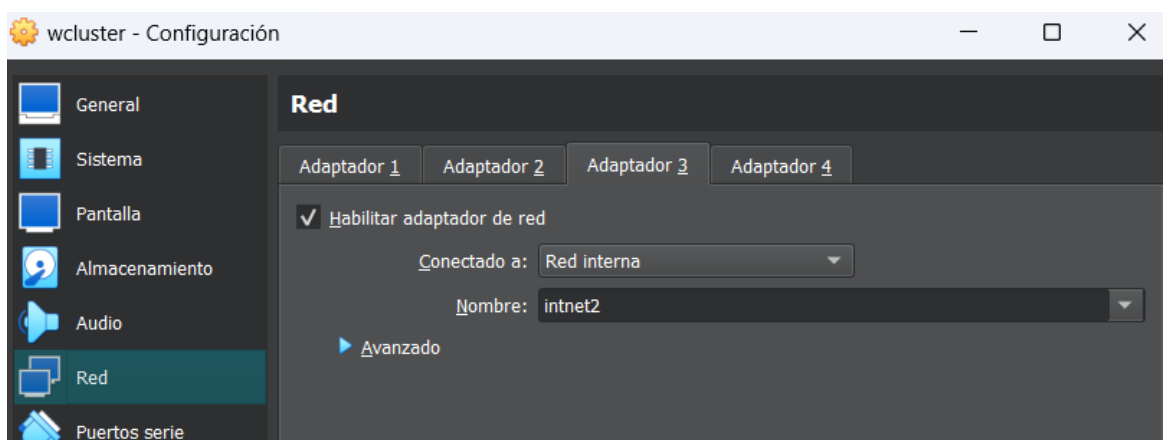
1. “echo 1 > /proc/sys/net/ipv4/ip_forward”, habría que ejecutar primero de todo este commando para activar el *forwarding*.
2. “sudo nano /etc/sysctl.conf”, abríamos este archivo para descomentar la siguiente línea: “net.ipv4.ip_forward = 1”, para hacerlo persistente.
3. Para habilitar el NAT debemos agregar una norma nueva a IPTABLES que sería la siguiente: “iptables -t nat POSTROUTING -o enp0s3 -j MASQUERADE”
4. Para hacer persistente esta norma nueva de IPTABLES debemos instalar “iptables-persistent” con un comando sencillo de instalación y ahora poner en la terminal “iptables-save > /etc/iptables/rules.v4”.

- **Configuración de ejemplo en los nodos:**

Para configurar correctamente el nodo balanceador, como pertenece a la red *main*, lo conectamos a la interfaz de red *intnet*, la cual esta en la red 10.0.0.0/24.



Para configurar correctamente el nodo balanceador, como pertenece a la red *internal*, lo conectamos a la interfaz de red *intnet2*, la cual está en la red 10.0.1.0/24.



Nodos balanceadores de carga

Una vez completa la instalación general del nodo, comprobaremos que el servicio está activo utilizando el comando siguiente (que debería estar activo si el archivo puppet es correcto):

```
$ systemctl status haproxy
```

En caso de no estarlo, completaremos la configuración de los nodos balanceadores de carga iniciando el servicio:

```
$ systemctl start haproxy
```

Nodos CMS

Una vez finalizada la instalación general, haremos uso de una herramienta llamada *wp-cli* ya incorporada durante la fase anterior de la instalación. Usaremos el siguiente comando dentro de la carpeta **/var/www/html/wordpress** que debe retornar “*Success*”:

```
$ wp core is-installed
```

En caso contrario deberemos ejecutar el archivo **wp-database.sh** que se encuentra en el mismo directorio. Volveremos a ejecutar el comando anterior para verificar que se haya instalado correctamente.

Nodos clustering

Master k8s

Cuando la instalación general se haya completado, buscaremos los comandos que están en el archivo `/var/kube.sh` y los ejecutaremos en la terminal todos excepto el comando

```
helm install mariadb bitnami/mariadb
```

que será ejecutado una vez se haya unido, al menos, un nodo *slave* correctamente a nuestro clúster.

Para comprobar que el nodo se haya creado correctamente usaremos el comando

```
$ kubectl get nodes
```

en caso de obtengamos una entrada con nuestro nodo master, significará que hemos realizado la configuración correctamente y que los nodos *slave* ya pueden instalarse.

Cuando se hayan añadido los nodos *slave* al clúster ejecutaremos el comando excluido anteriormente y comprobaremos que el despliegue de MariaDB funciona correctamente con el siguiente comando:

```
$ kubectl get pods
```

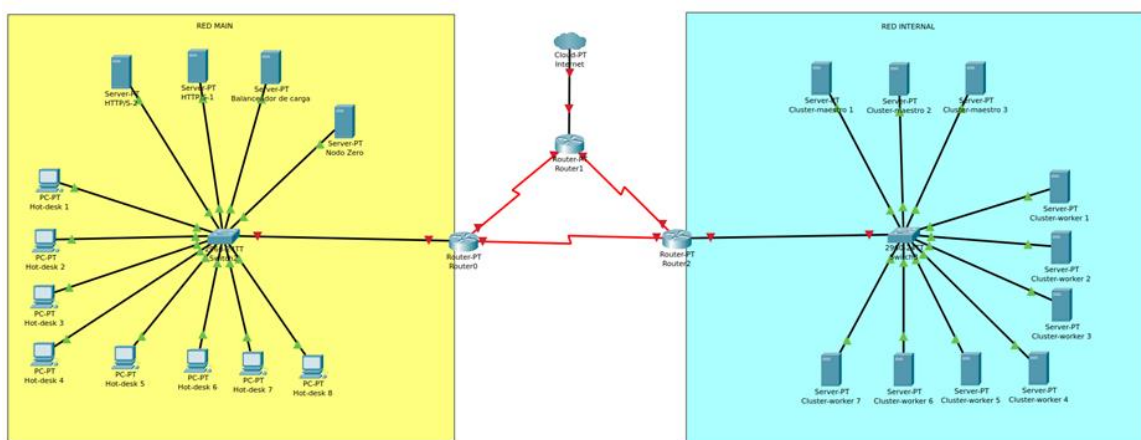
Slave k8s

Una vez instalada la configuración general, esperaremos a que el nodo *master* haya completado la creación del clúster para unirnos al mismo. Lo haremos ejecutando el archivo `/var/kube_slave.sh`.

Para comprobar que el nodo se ha añadido correctamente al clúster deberemos ejecutar desde un nodo *master* el siguiente comando, obteniendo una entrada correspondiente a este nodo *slave*:

```
$ kubectl get nodes
```

Maqueta redes con packettracer.



Enlace para el nodo 0: https://pruebasaluuclm-my.sharepoint.com/:u:/g/personal/alberto_moreno15_alu_uclm_es/EQgHUrcpXSBHstutQuy7a2sB7iG4kX4SK31Ai0DdV5qbUA?e=mhnTvJ