**MANUAL DE COM AFEGIR UN NOU NODE DE IA A PROXMOX**

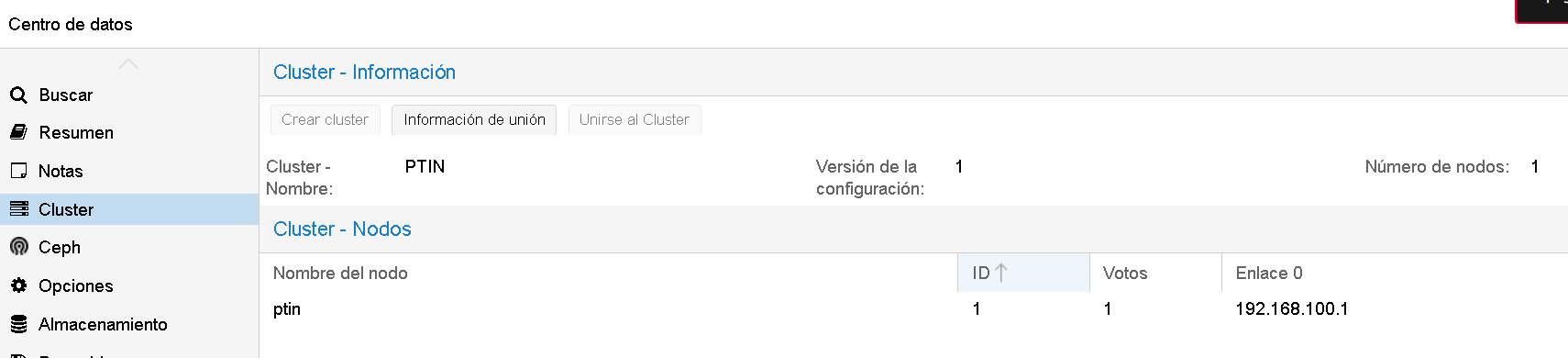
# Objectiu i possibilitats de les màquines:

1.Tenim un servidor principal amb proxmox instalar y operatiu , aquest té moltes entrades ethernet , el que es usable per a tenir una bona connexió directa amb l'altre node nou.

2.Tenim un portàtil amb una gráfica dedicada NVIDIA el cual es el nou node a incorporar per als grup encarregat de IA.

# Procediment:

Creem un Cluster per a tenir més d’un node:

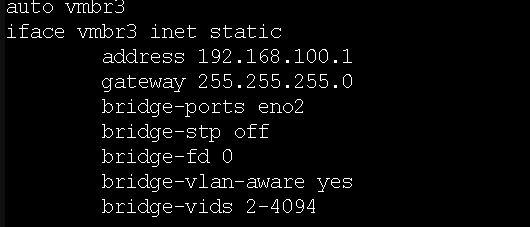


Tenim que configurar la connexió entre nodes , tenim lliure el dispositiu eno2 per el que podem comunicar via cable ethernet una subred pair to pair.

Posem una Ip fixa amb una subred virtual per a la connectivitat entre nodes.

Aquest és el principal:

nano /etc/network/interfaces



I afegir les regles de comunicació amb el rang on es troba el nou node.



Instalem el Mateix sistema operatiu al nou node.

Configurem el sistema igual que al altre , les interfaces:

nano /etc/network/interfaces

I afegir les regles de comunicació amb el rang on es troba el nou node.

Executem la comanda per a afegir la al cluster :

pvecm add 192.168.100.1

# MIGRACIÓ DE LA VM:

Per a començar mirarem la màquina de ia , tenint en compte les prestacions de hardware del nou node i si es el cas modificarla .

Per a migrar pararem la VM de IA i amb clic dret farem un MIGRATE al nou node.

# PROBLEMES AMB EL NODE PORTÀTIL I SOLUCIÓ:

La arquitectura del portátil al tenir un mix de component integrats en la placa i no se a través de pci Express ,proxmox no permet compartir la grafica amb la VM, s’ha obtat per a instalar directament un ubuntu server, fent que la VM sigui directament la maquina.

# 

# INSTAL·LACIÓ DE DRIVERS I CONFIGURAR GPU:

DESACTIVAR el secure mod de la BIOS del portatil.

Activa IOMMU en el GRUB:

* nano /etc/default/grub
* Editar

GRUB\_CMDLINE\_LINUX\_DEFAULT="quiet intel\_iommu=on"

* Guarda y actualiza grub:

update-grub

Crea o edita /etc/modules

vfio

vfio\_iommu\_type1

vfio\_pci

vfio\_virqfd

Regenera el initramfs

update-initramfs -u -k all

reboot

Verifica que la GPU esté “secuestrada” por vfio

lspci -nnk -d 10de:1c82

tiene que decir Kernel driver in use: vfio-pci

En la parte de Hardware de la VM , Anyadir , PCIespres i seleccionamos la GPU.

DRIVERS:

1. Actualiza el sistema

sudo apt update && sudo apt upgrade -y

2. Habilita el repositorio de controladores adicionales

sudo add-apt-repository ppa:graphics-drivers/ppa -y

sudo apt update

3. Instala el controlador propietario de NVIDIA

sudo apt install nvidia-driver-535 -y

4. Reboot

sudo reboot

5. Instalar CUDA Toolkit (Recomendado: v12.2 o superior)

A. Instala dependencias básicas:

sudo apt install build-essential dkms -y

B. Añade el repositorio de NVIDIA:

Ve al sitio oficial: https://developer.nvidia.com/cuda-downloads

Selecciona:

Linux Ubuntu Versión (22.04 o 20.04, según tu sistema)

wget https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86\_64/cuda-ubuntu2204.pin

sudo mv cuda-ubuntu2204.pin /etc/apt/preferences.d/cuda-repository-pin-600

sudo apt-key adv --fetch-keys https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86\_64/3bf863cc.pub

sudo add-apt-repository "deb https://developer.download.nvidia.com/compute/cuda/repos/ubuntu2204/x86\_64/ /"

sudo apt update

sudo apt install cuda -y

6. Añadir CUDA al PATH (importantísimo)

Edita tu bash profile:

nano ~/.bashrc

Agrega al final:

export PATH=/usr/local/cuda/bin:$PATH

export LD\_LIBRARY\_PATH=/usr/local/cuda/lib64:$LD\_LIBRARY\_PATH

Guarda y recarga:

source ~/.bashrc