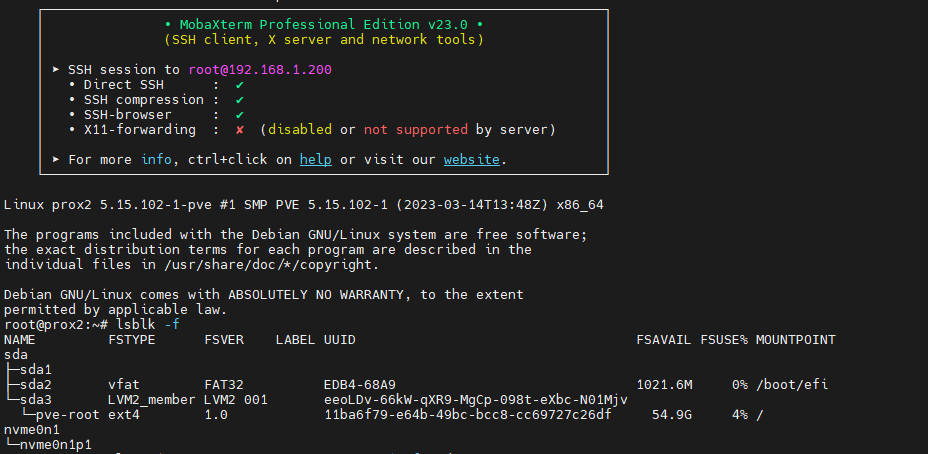
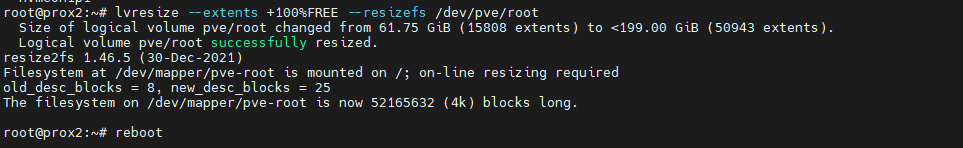
* Install Proxmox from ISO:
* 
* Check disk:

*lsblk*



*lvresize --extents +100%FREE --resizefs /dev/pve/root*

*reboot*



*nano /etc/apt/sources.list.d/pve-enterprise.list*

deb http://download.proxmox.com/debian/pve bookworm pve-no-subscription

*nano /etc/apt/sources.list.d/ceph.list*

deb http://download.proxmox.com/debian/ceph-quincy bookworm no-subscription

*apt update -y*

*apt --fix-broken install*

*apt install unzip -y*

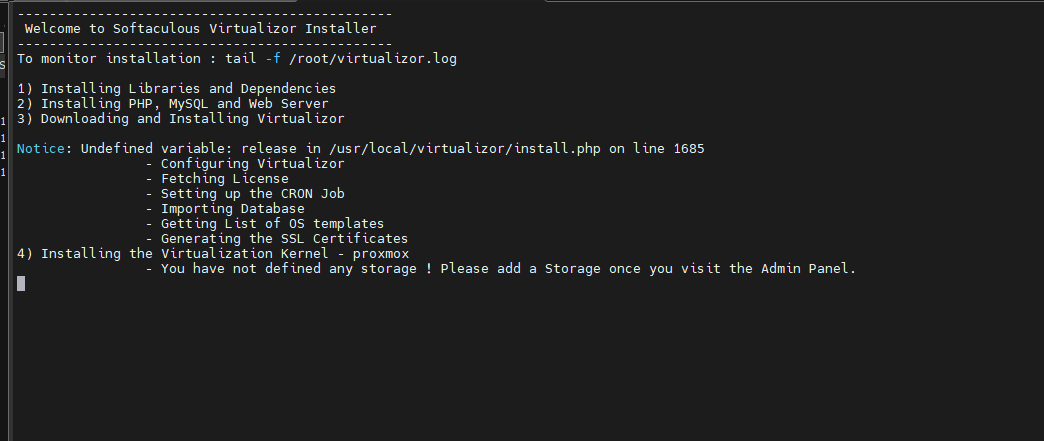
*apt install parted screen net-tools -y*

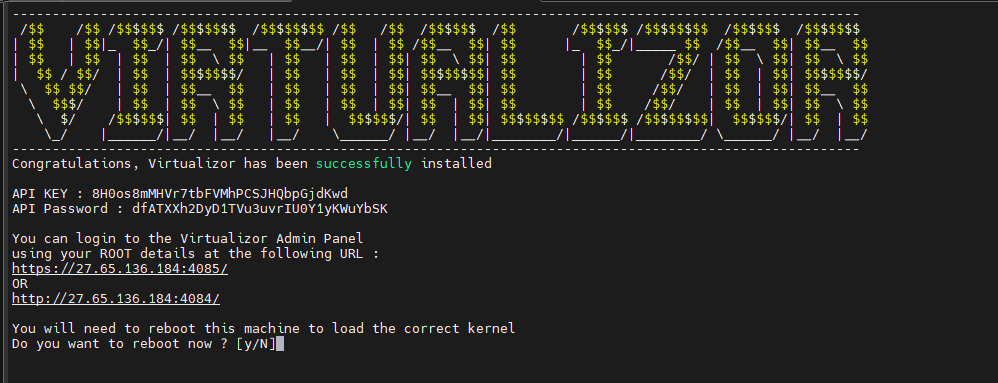
*screen*

*wget -N http://files.virtualizor.com/install.sh*

*chmod +x install.sh*

./install.sh email=kythuat@megahost.vn kernel=proxmox license=VIRTE-80367-83284-46673-91844





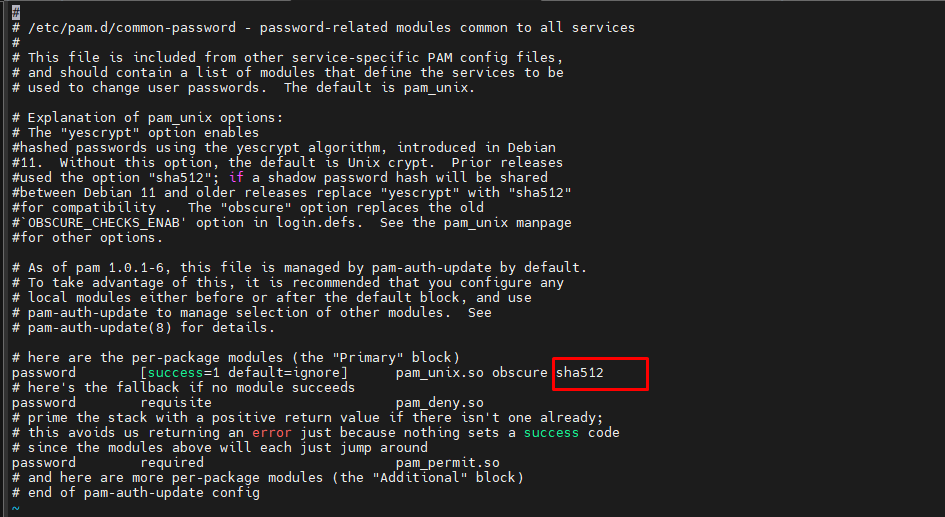
*nano /etc/pam.d/common-password*

Edit:

password [success=1 default=ignore] pam\_unix.so obscure yescrypt

To:

password [success=1 default=ignore] pam\_unix.so obscure sha512



Update password root:

*passwd root*

Test truy cập web contrl và login bằng user root

https://192.168.1.192:4085/

Format ổ đĩa :   
parted -s /dev/sde mklabel gpt

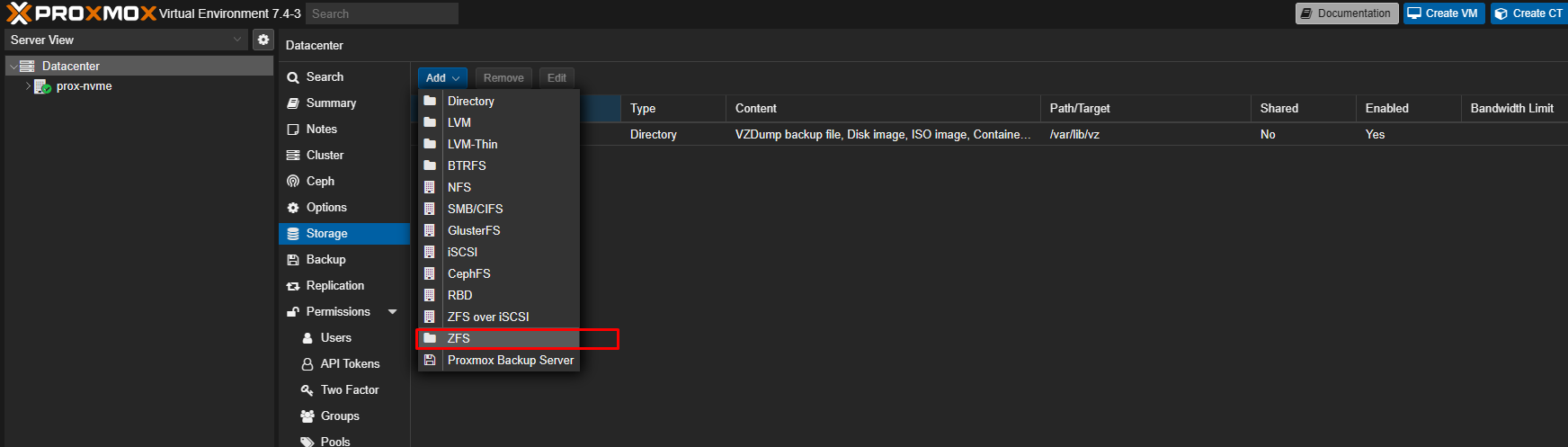
=🡺 vào Proxmox tạo raid cho ổ đĩa

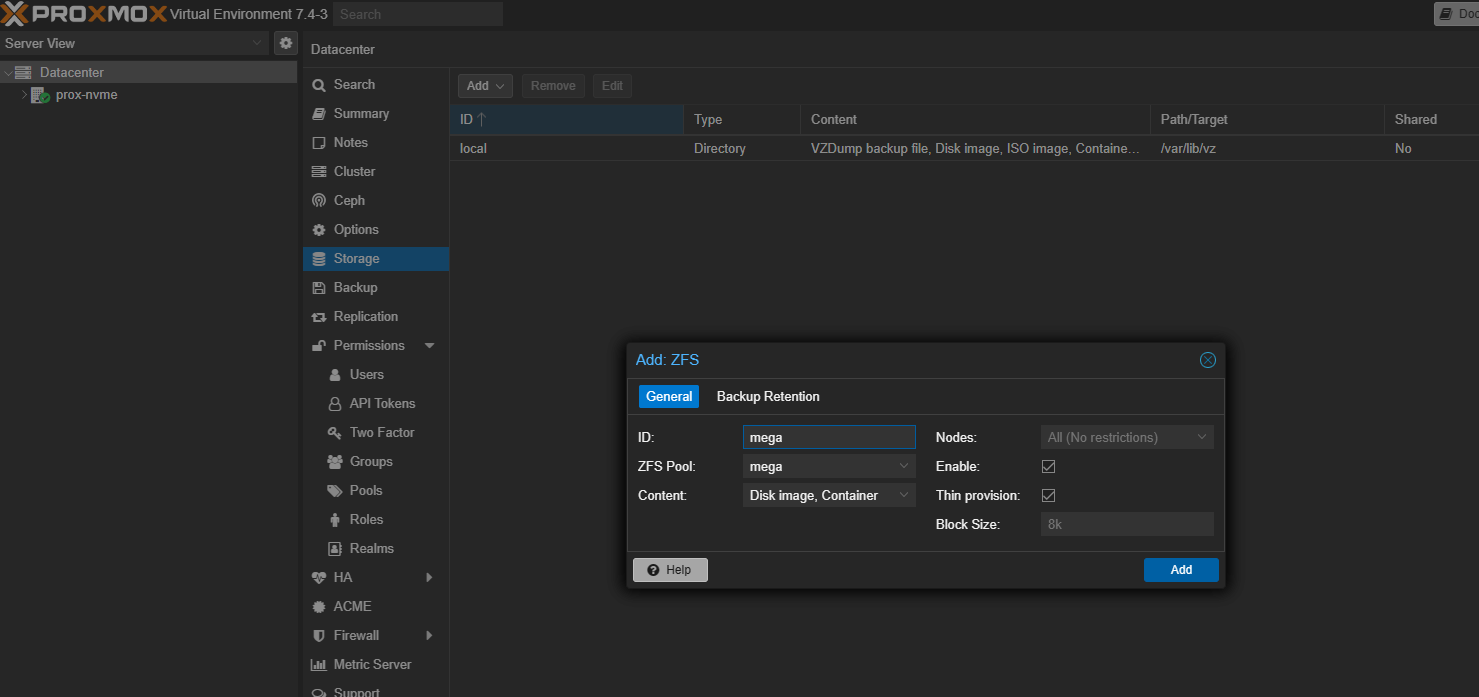
Ảnh có chứa văn bản, phần mềm, Hệ điều hành, máy tính

Nội dung do AI tạo ra có thể không chính xác.

* Truy cập Proxmox add storage:

https://192.168.1.205:8006/





*zfs set atime=off megahost*

*zfs set xattr=sa megahost*

*vi /etc/modprobe.d/zfs.conf*

options zfs zfs\_arc\_min=8589934592

options zfs zfs\_arc\_max=17179869184

echo "$[8 \* 1024\*1024\*1024 - 1]" >/sys/module/zfs/parameters/zfs\_arc\_min

echo "$[16 \* 1024\*1024\*1024]" >/sys/module/zfs/parameters/zfs\_arc\_max

update-initramfs -u -k all

Fix lỗi growpark khi cài OS windown trên Virtual

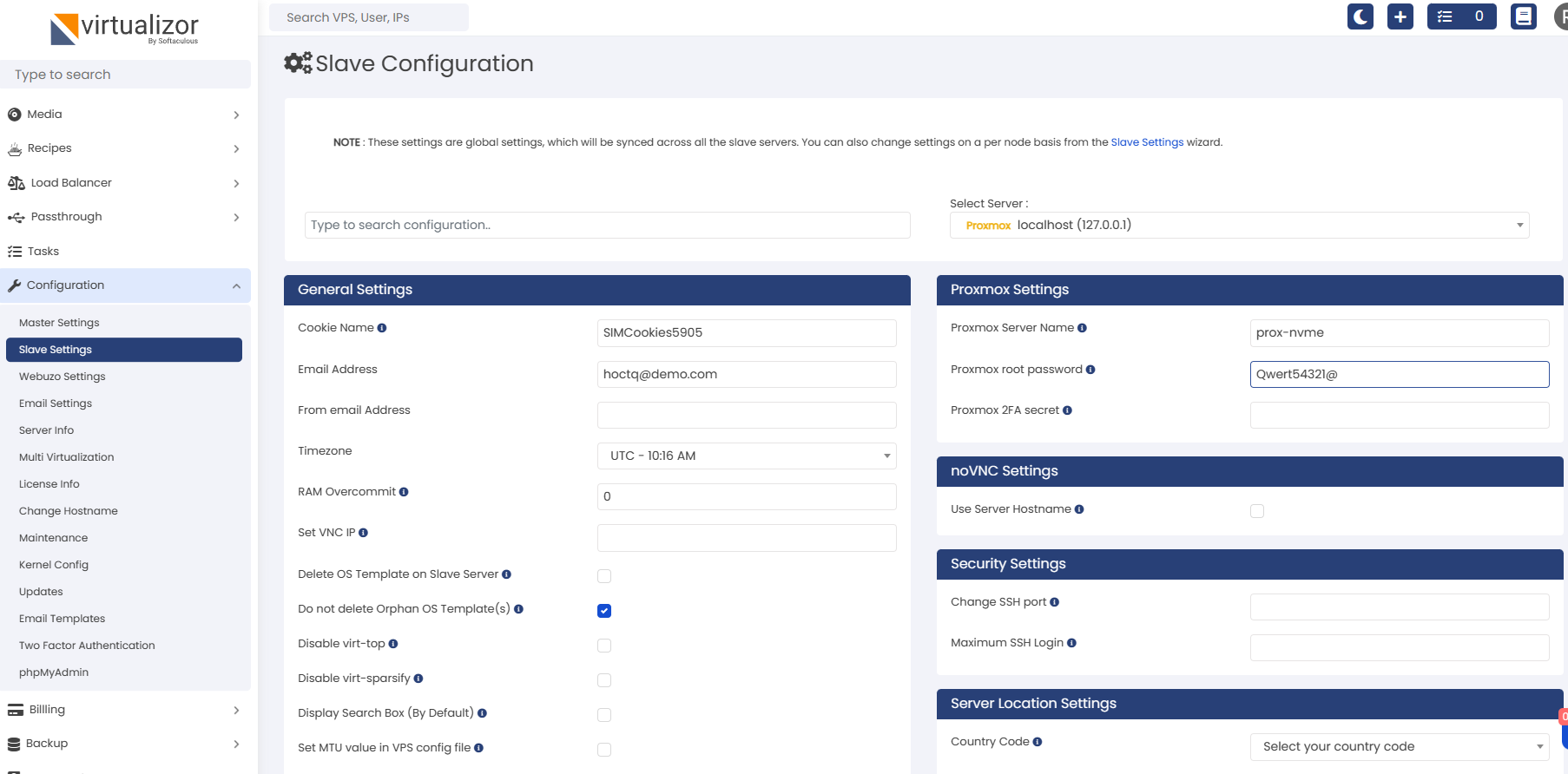
apt install cloud-utils -y

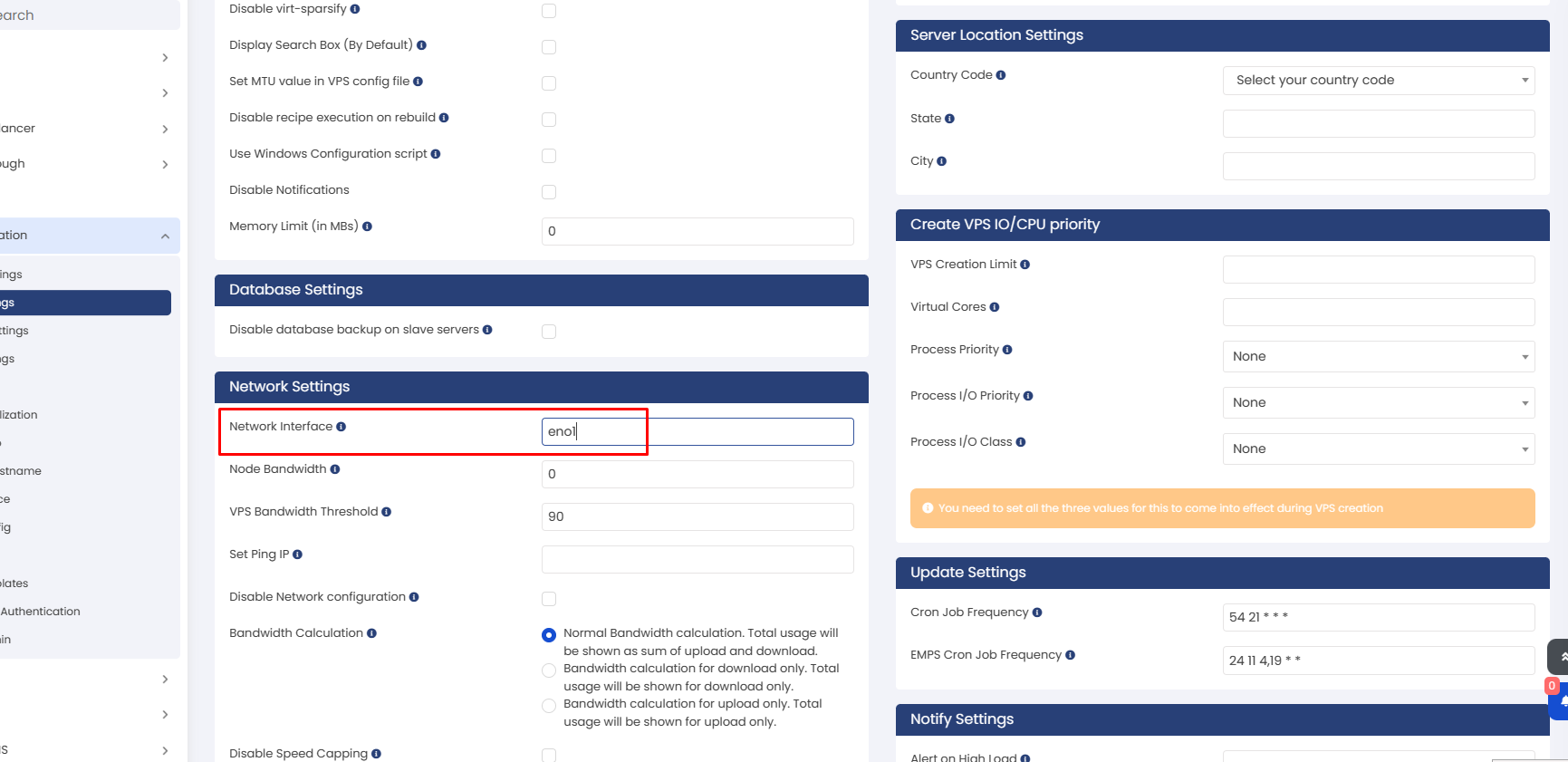
apt install cloud-guest-utilslsblk

apt-get install chntpw

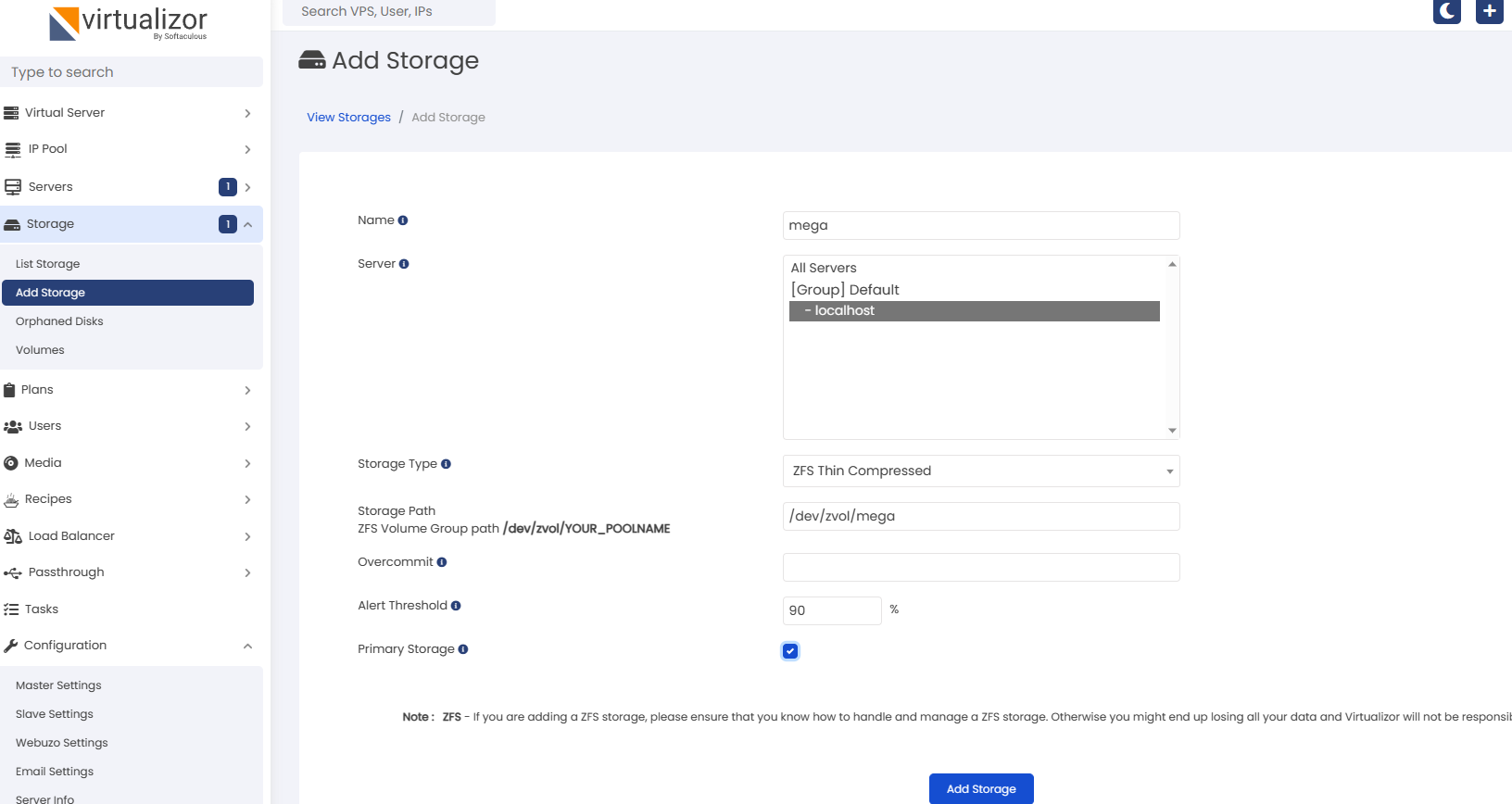
Virtualizor

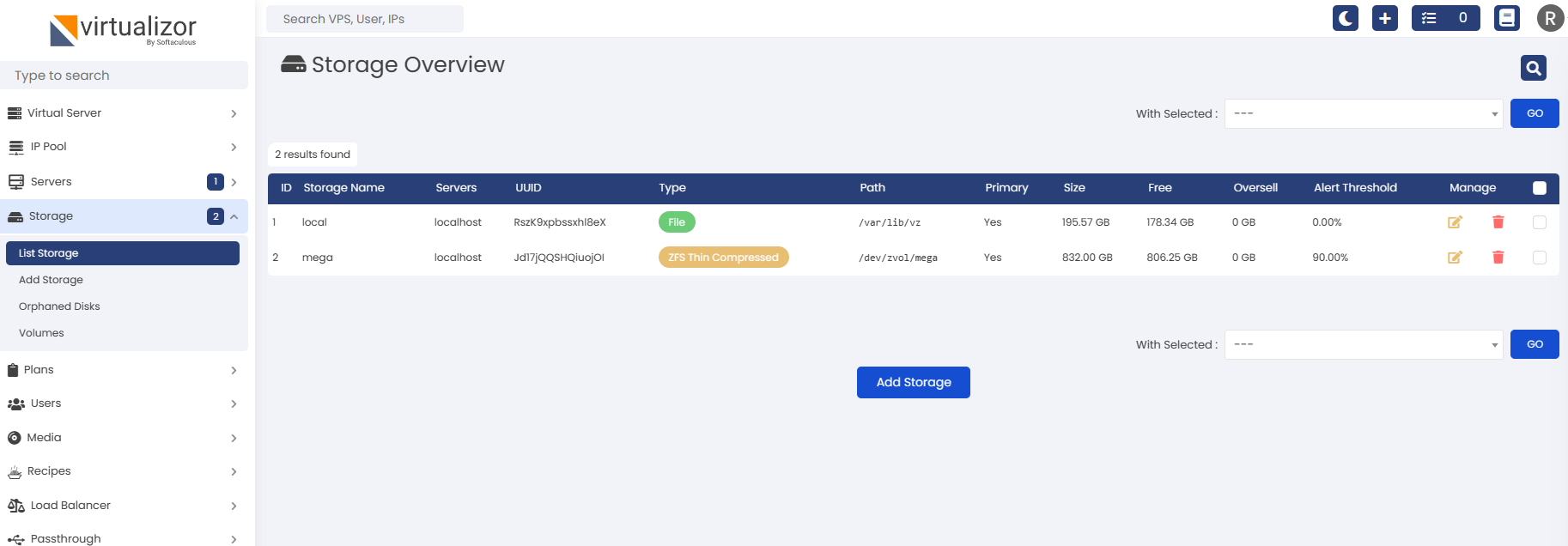
* Khai báo hostname, password proxmox và Network interface to vitualizor:
* Khai báo hostname, password proxmox và Network interface to vitualizor:





* Truy cập control để add storage:

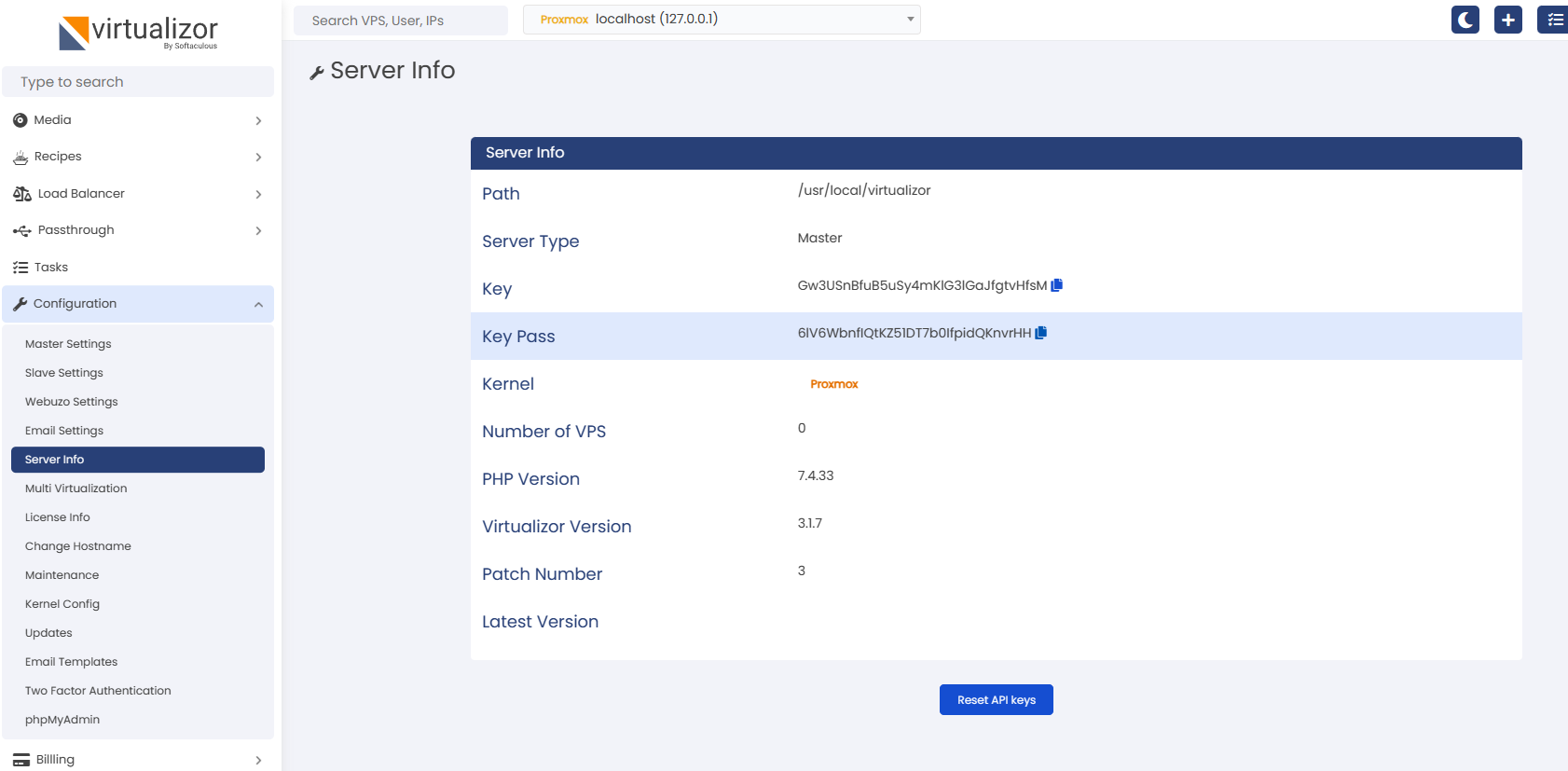


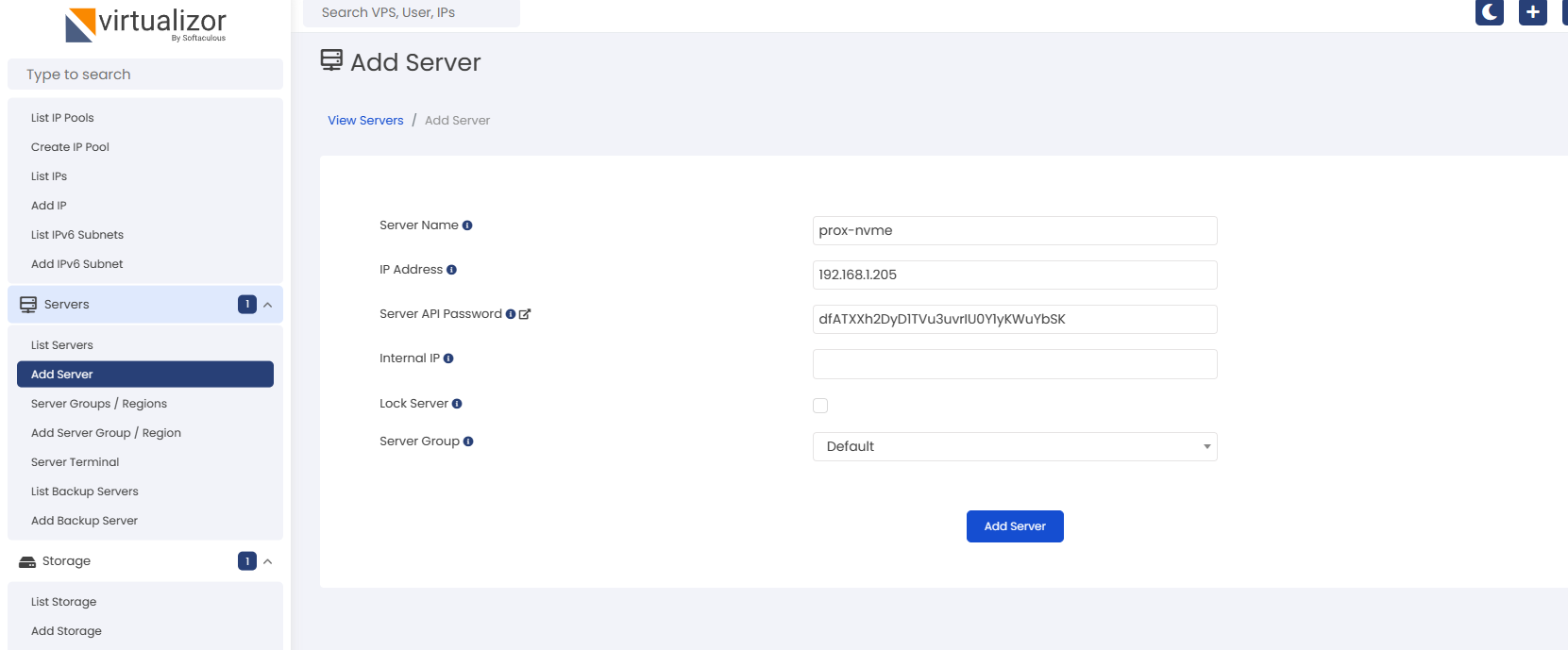


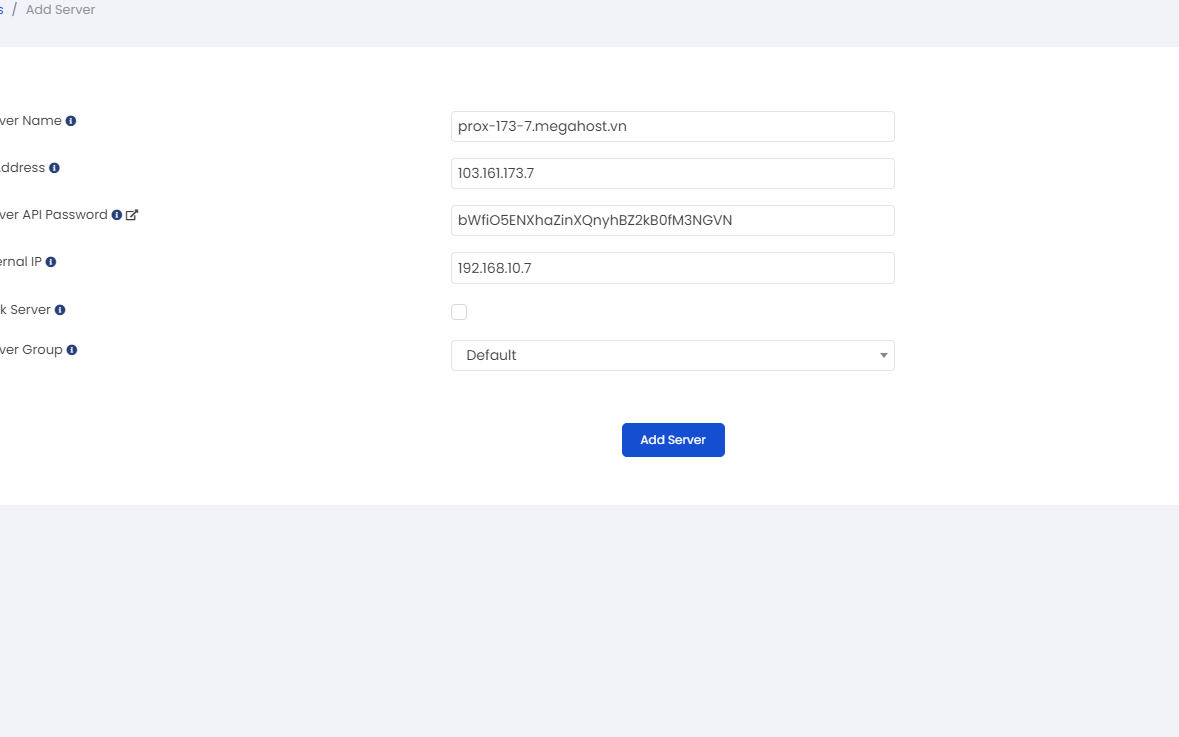
* Add node to master:

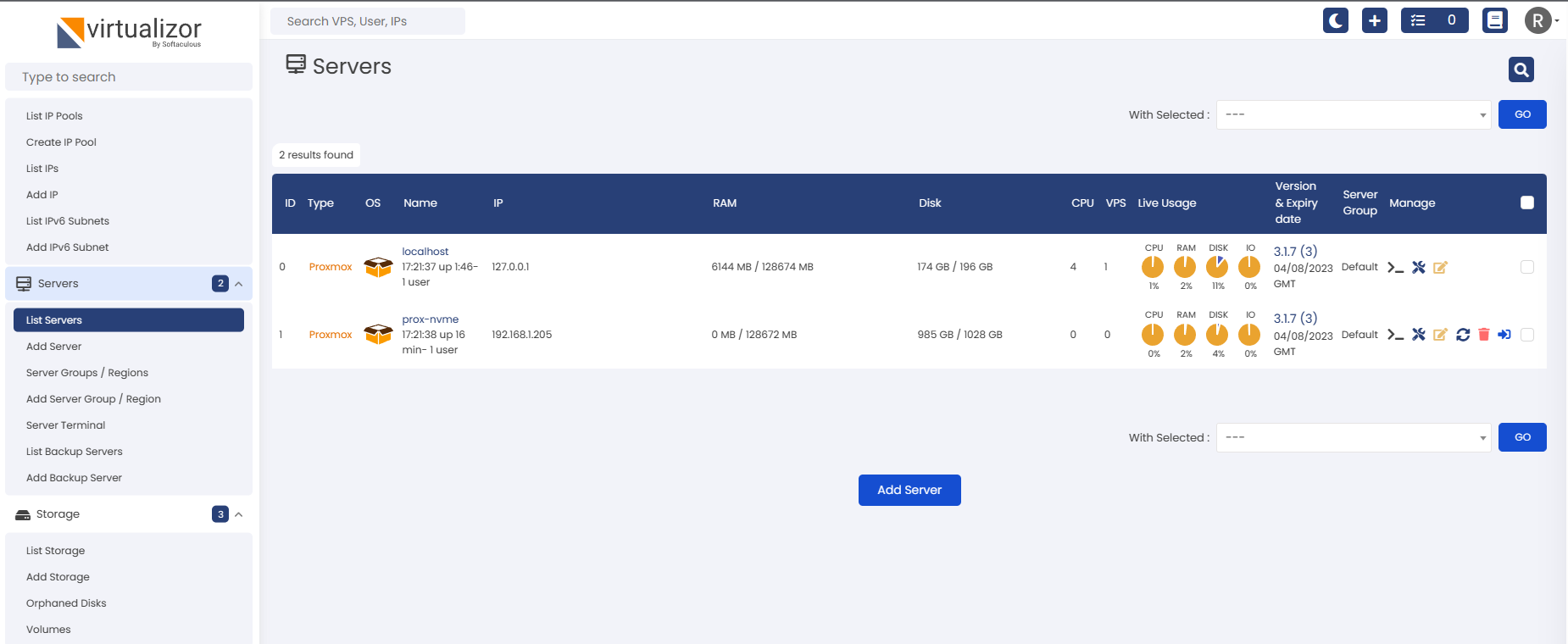
Get server API password for node server:

Cần allow ip cho node trên master







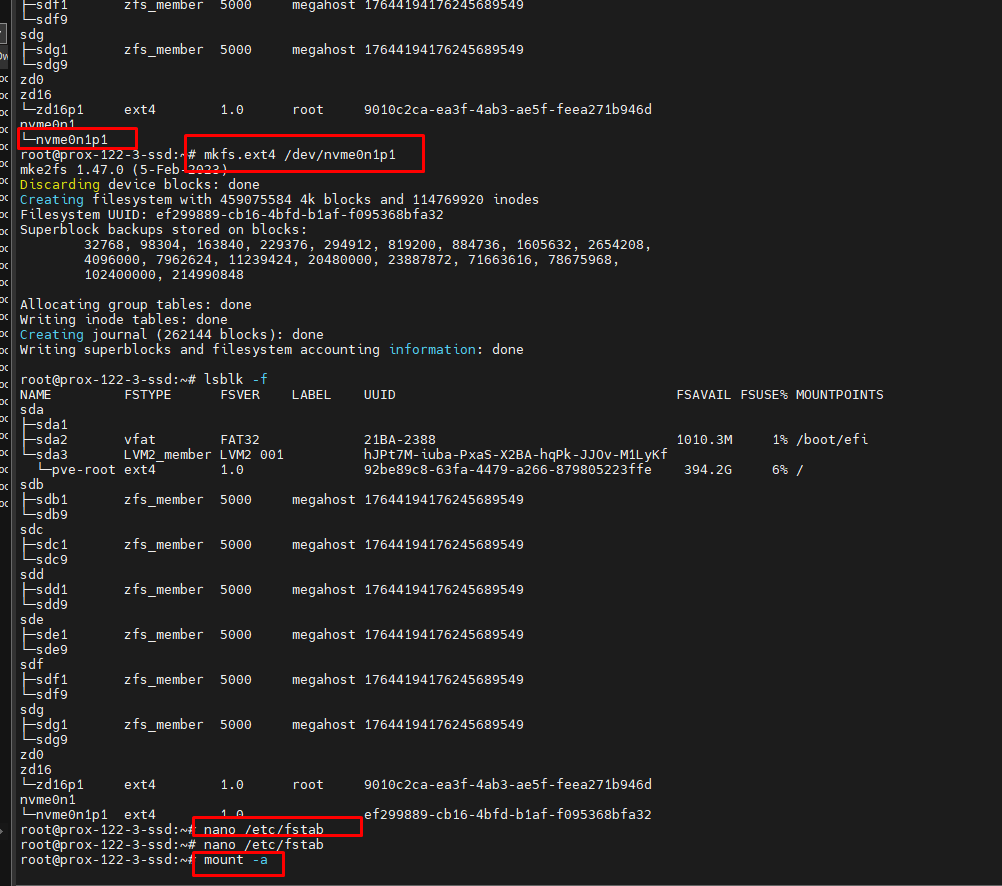


Tạo thư mục backup

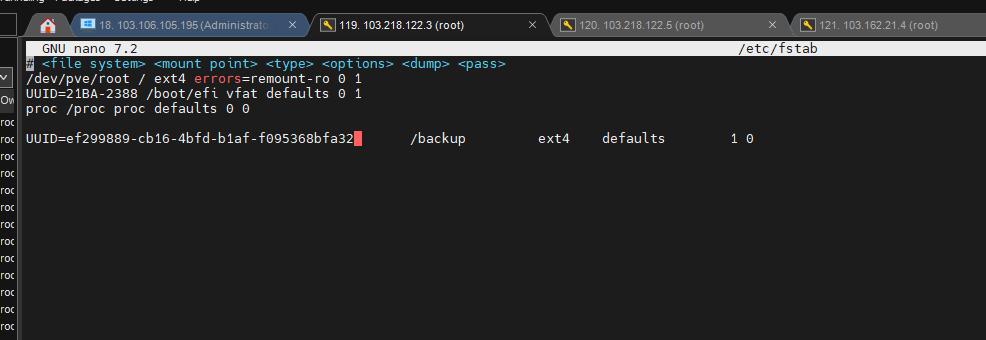
mkdir /backup

mkfs.ext4 /dev/sdX1

format ổ cứng backup



Gắn kết vào fstab



nano /etc/fstab

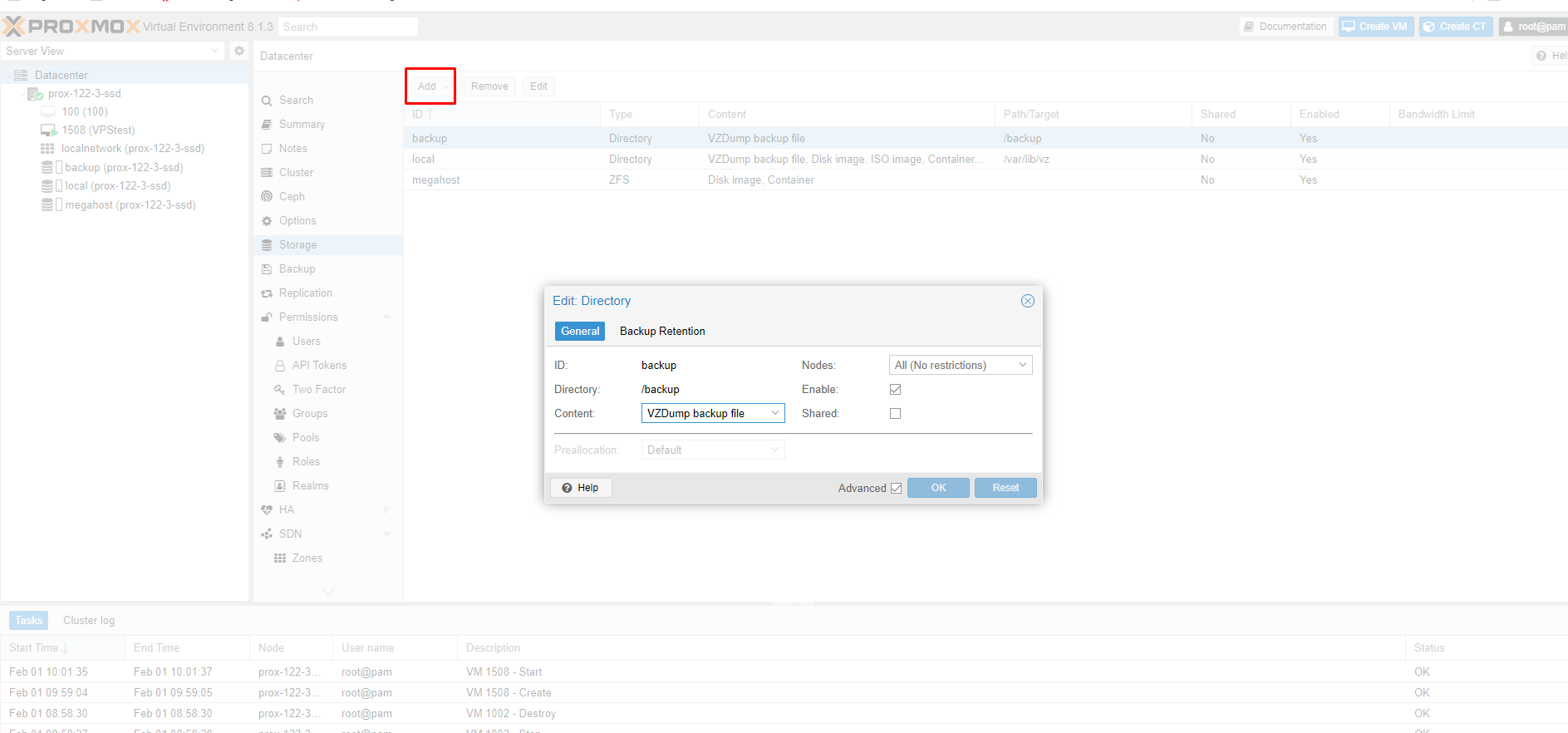
mount -a

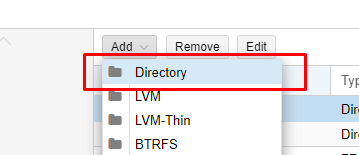
systemctl daemon-reload

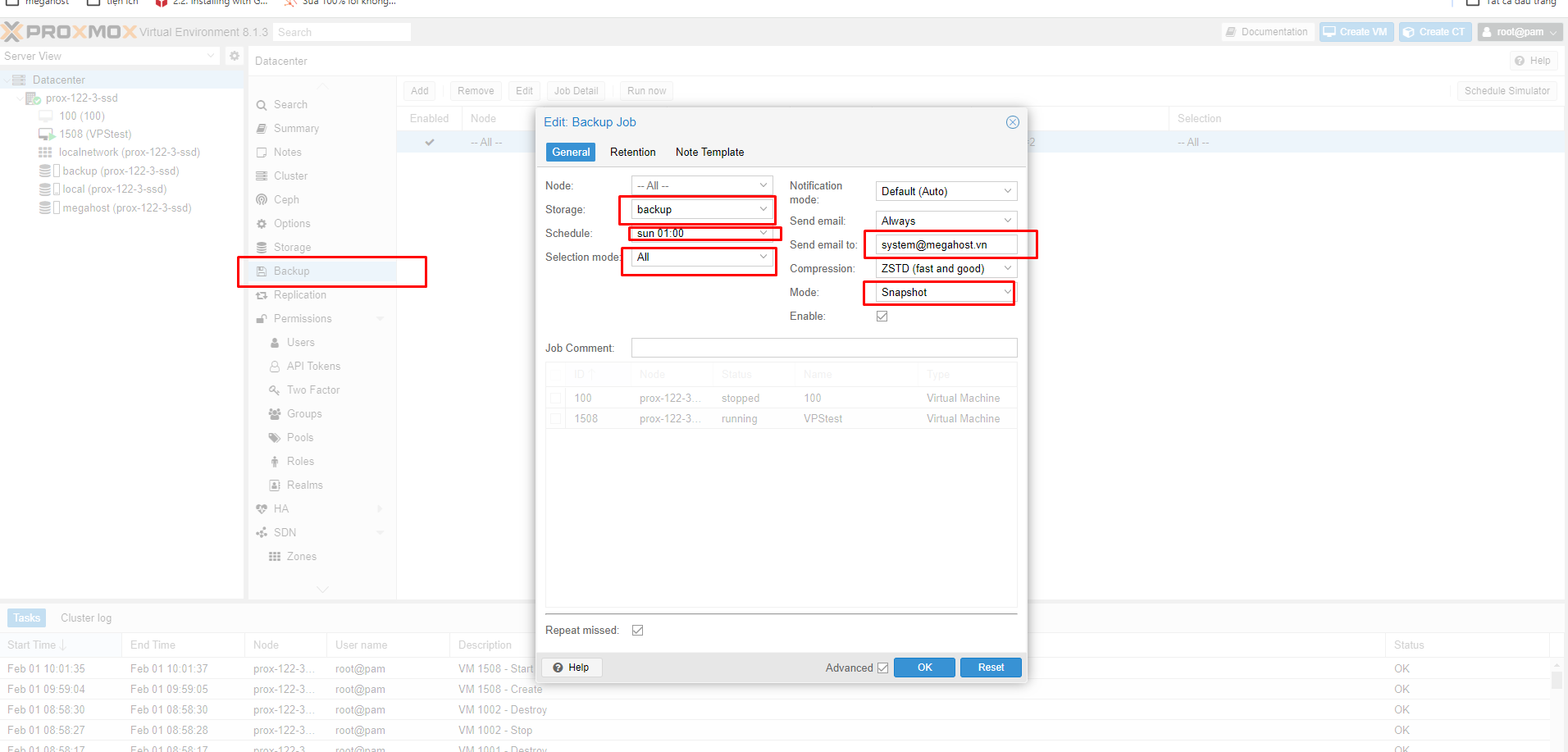
df -h

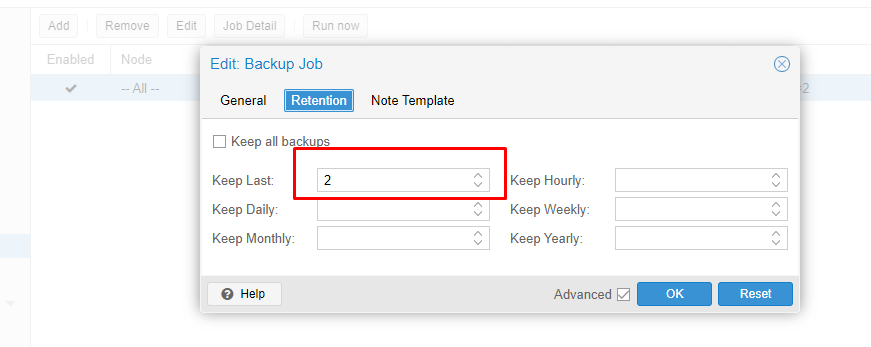
EX: UUID=4bfb69f2-bd25-4c4e-bf1d-d307b778dac2 /backup ext4 defaults 1 0

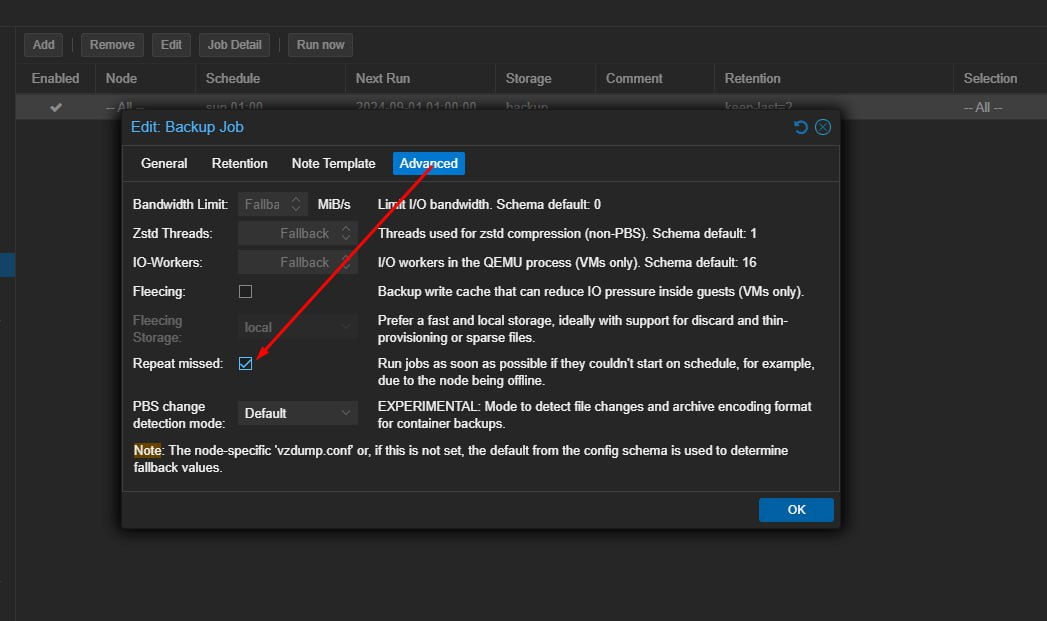
Add storage + config backup cho proxmox



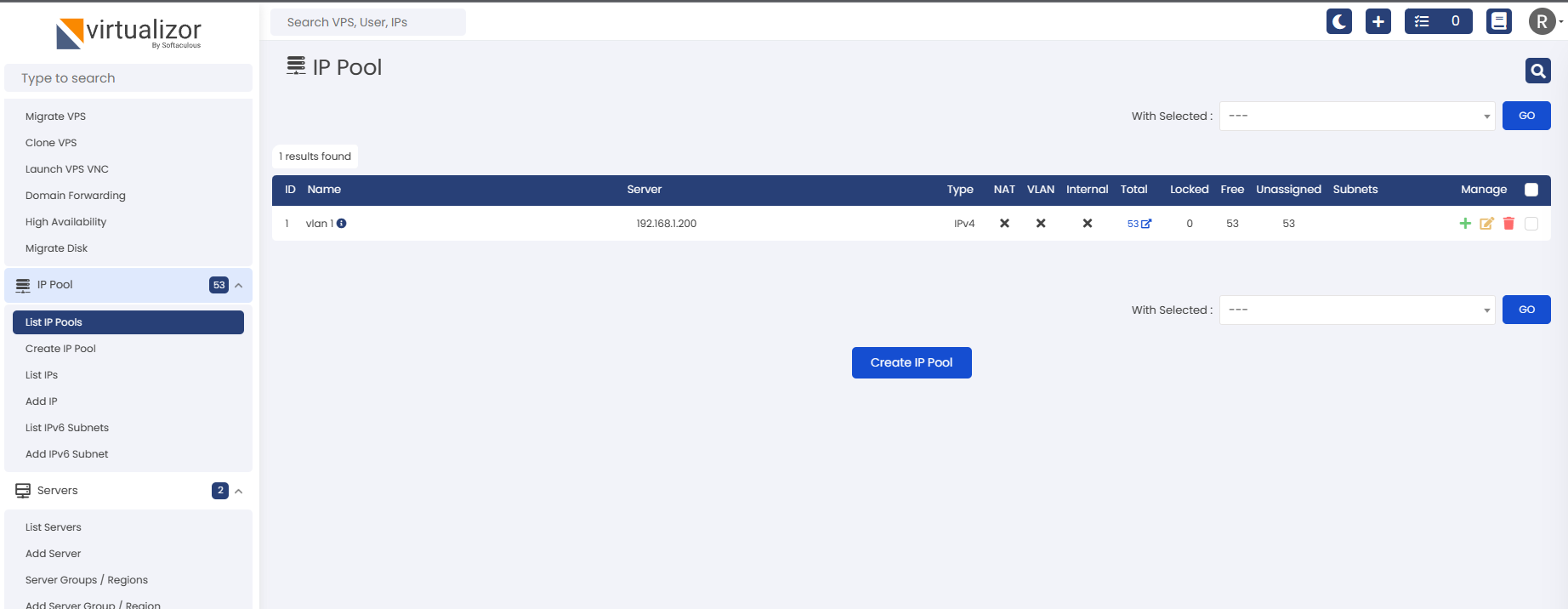




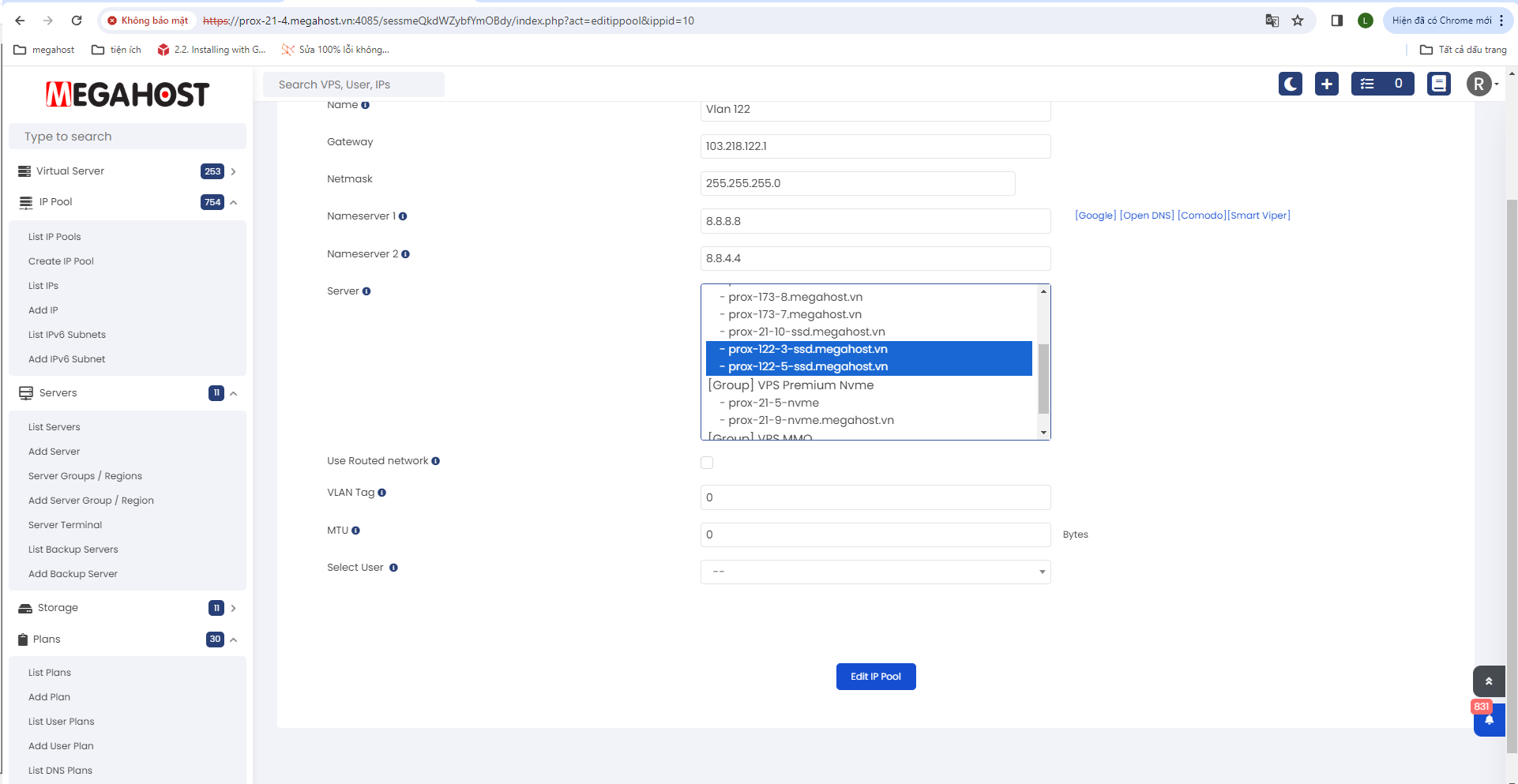




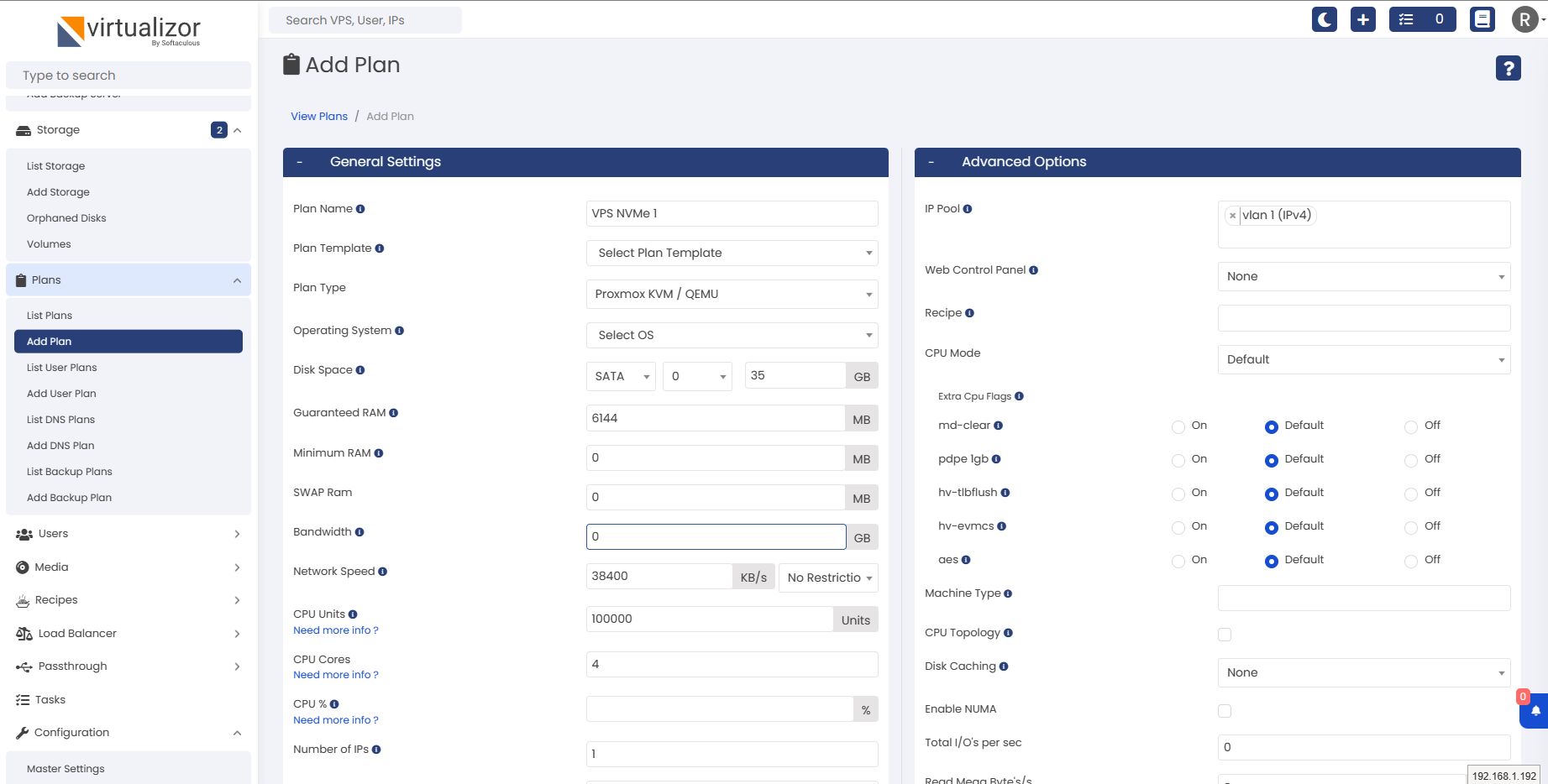
* Tạo IP pool

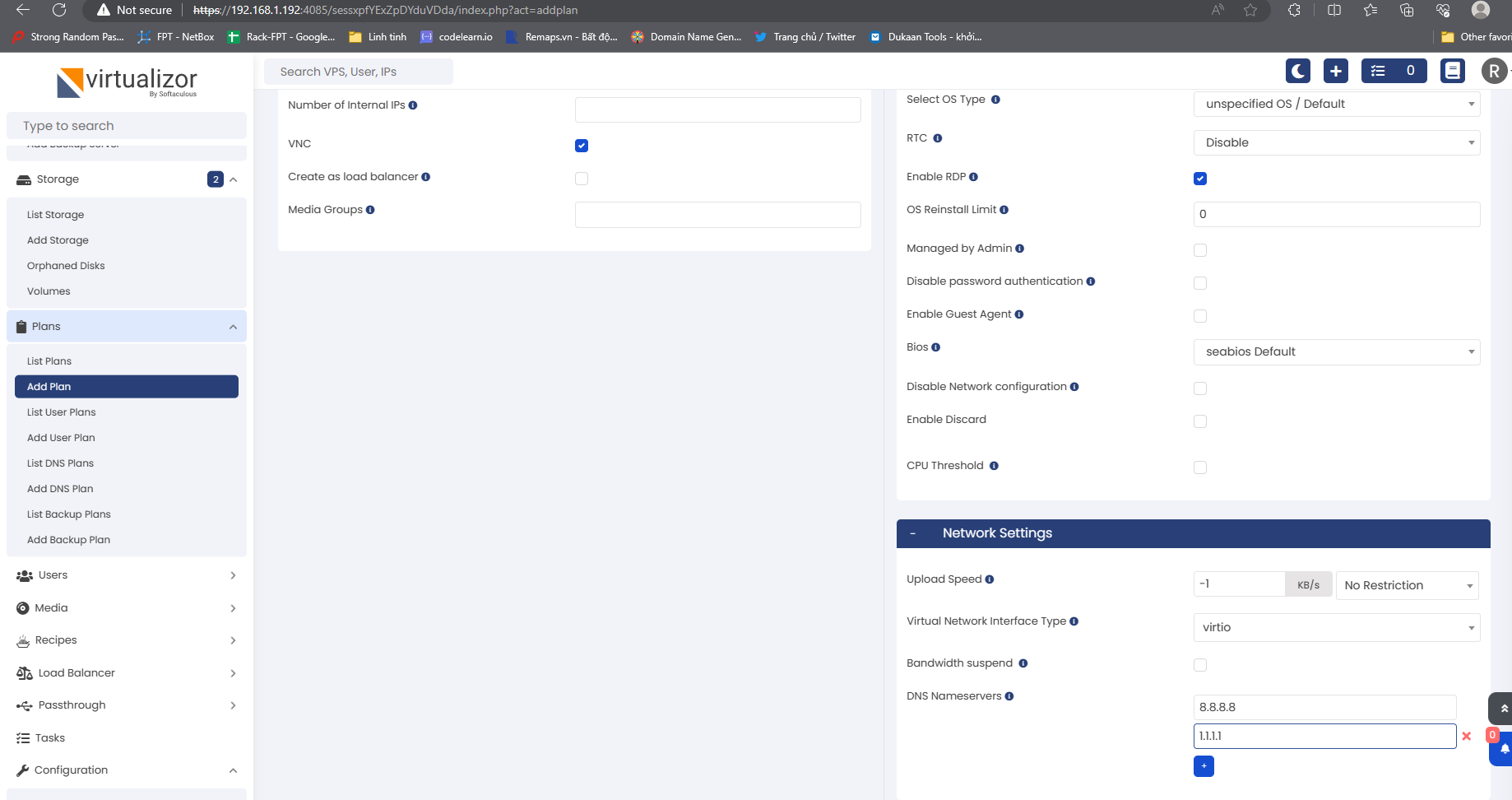


Chỉnh các node dùng chung dải ip

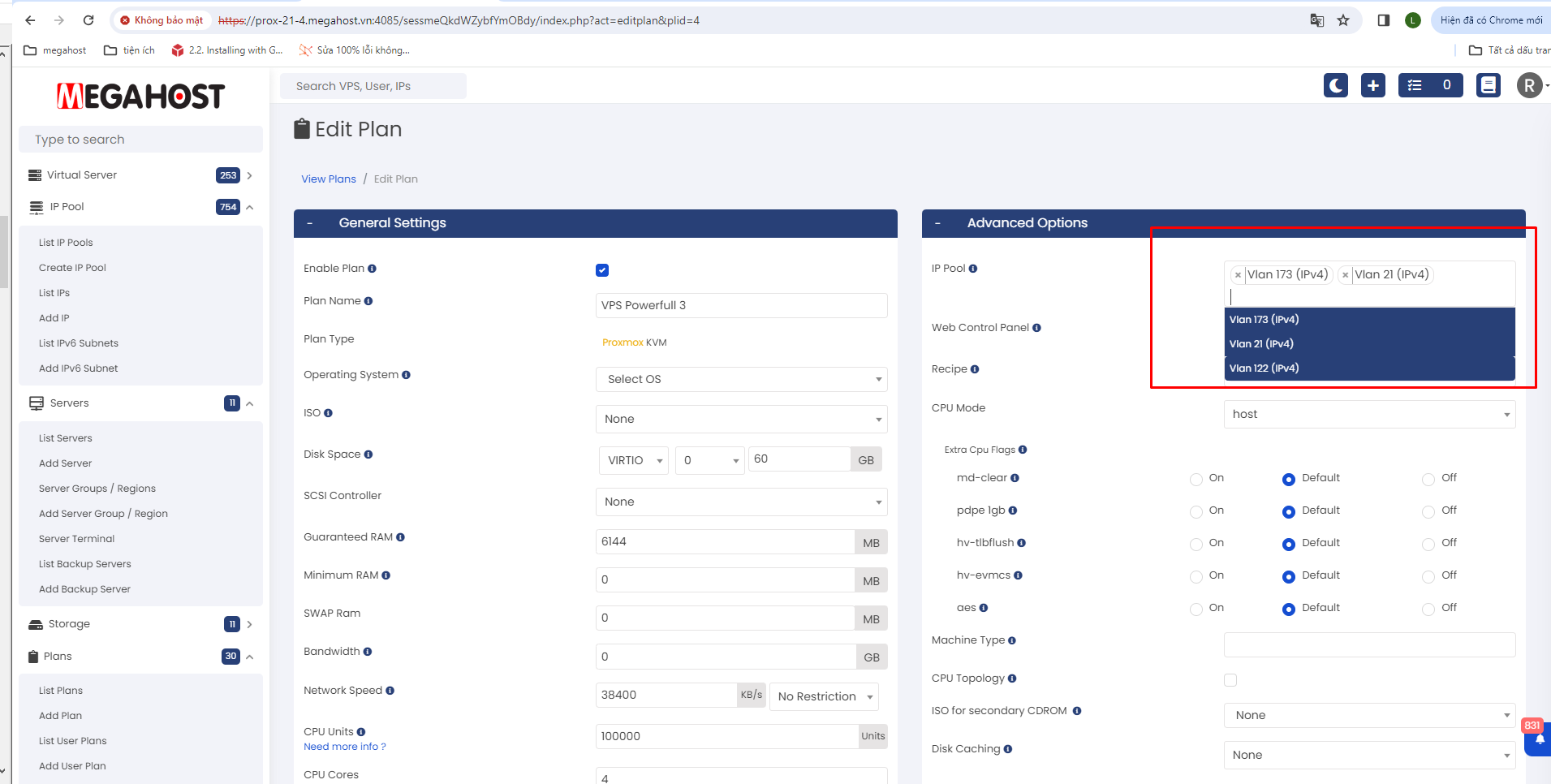


* Tạo Plan





Chinh plan dùng dải ip



SETUP FIREWALL + ZABBIX

systemctl disable pve-firewall.service

pve-firewall stop

apt install perl-libwww-perl -y

cd /tmp

wget https://download.configserver.com/csf.tgz

tar -xzf csf.tgz

cd csf

sh install.sh

cài csf, chỉ allow ip office, ip vps chay whmcs và các range ip

nano /etc/csf/csf.conf

TCP\_IN = " 80,443,4081:4085,5900:7000,8443,2087,2086"

TCP\_OUT = "20,21,22,25,53,853,80,110,113,443,587,993,995"

TESTING = "0"

RESTRICT\_SYSLOG = "3"

nano /etc/csf/csf.allow

103.161.173.0/24 # Manually allowed: 103.161.173.0/24 (-) - Thu Feb 1 11:42:29 2024

103.162.21.0/24 # Manually allowed: 103.162.21.0/24 (-) - Thu Feb 1 11:43:00 2024

103.218.122.0/24 # Manually allowed: 103.218.122.0/24 (-) - Thu Feb 1 11:44:03 2024

103.218.122.28 # VPN-FPT

157.10.198.199 # VPN-CMC

103.162.21.27 # VPN

222.253.82.76 # VNPT MGH

103.162.21.100 # Zabbix

103.162.21.4 # Manually allowed: 103.162.21.4 (VN/Vietnam/static-4-21-162-103.tino.vn) - Tue Mar 12 10:15:50 2024

103.162.21.11 # Manually allowed: 103.162.21.11 (VN/Vietnam/-) - Sat Sep 7 00:34:14 2024

nano /etc/hosts.deny

sshd:All

nano /etc/hosts.allow

sshd: 103.162.21.11

sshd: 103.162.21.27 #VPN megahost

sshd: 157.10.198.199 #VPN-CMC

sshd: 103.218.122.28 #VPN-FPT

sshd: 10.10.10.0/24 #Local

sshd: 192.168.1.0/24

sshd: 127.0.0.1

sshd: [::1]

For Proxmox 8: Zabbix Agent

apt update

apt install zabbix-agent -y

nano /etc/zabbix/zabbix\_agentd.conf

-Server=103.161.173.26 🡪 103.162.21.100

-ServerActive=103.161.173.26 🡪 103.162.21.100

-Hostname=prox-173-7.megahost.vn

systemctl enable zabbix-agent

systemctl start zabbix-agent

tail -f /var/log/zabbix-agent/zabbix\_agentd.log

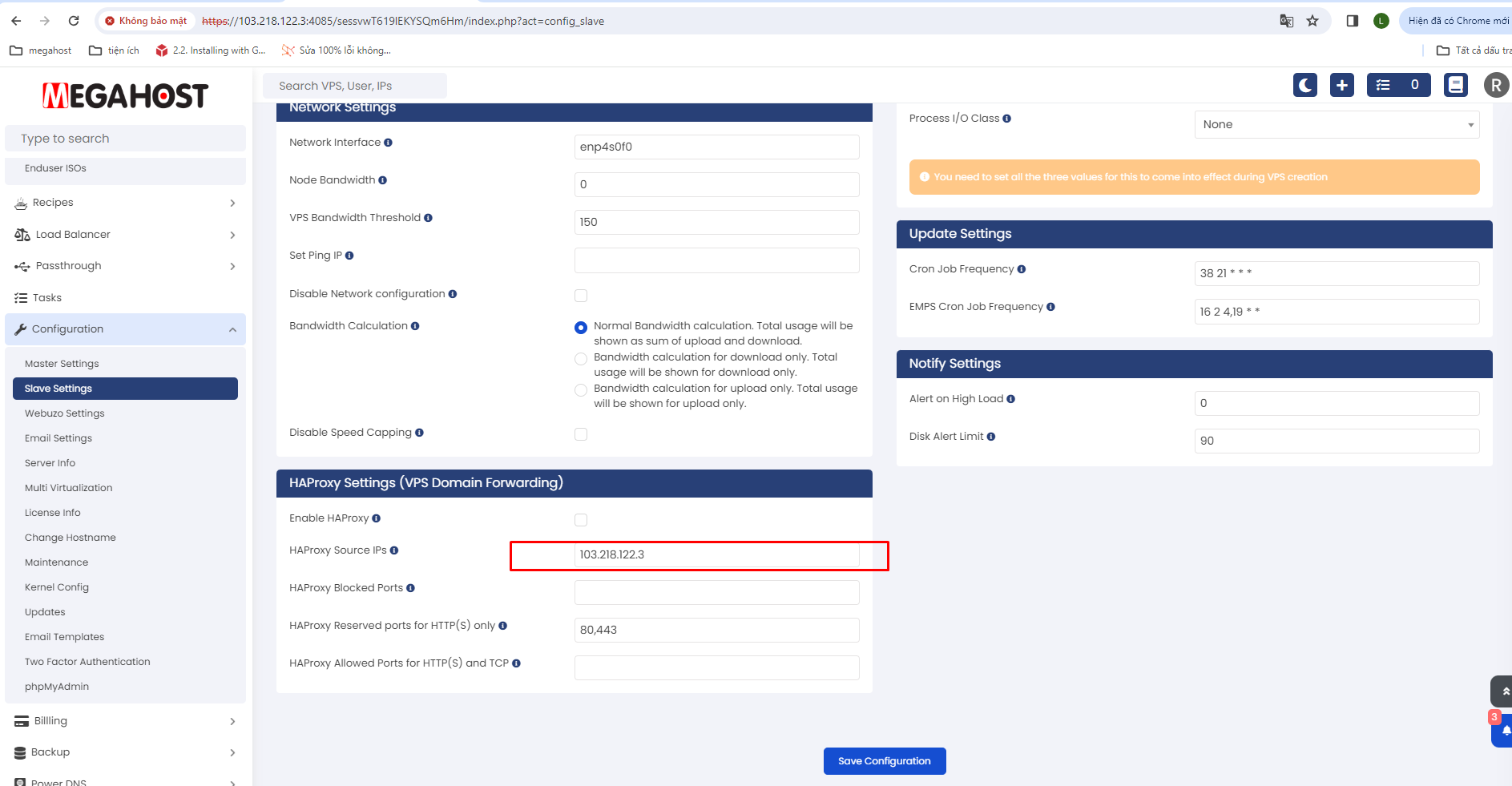
🡪web zabbix -> host 🡪 chọn 1 sv 🡪 full clone (sửa ip + tên sv)

* Zo root sv zabbix Allow ip node .

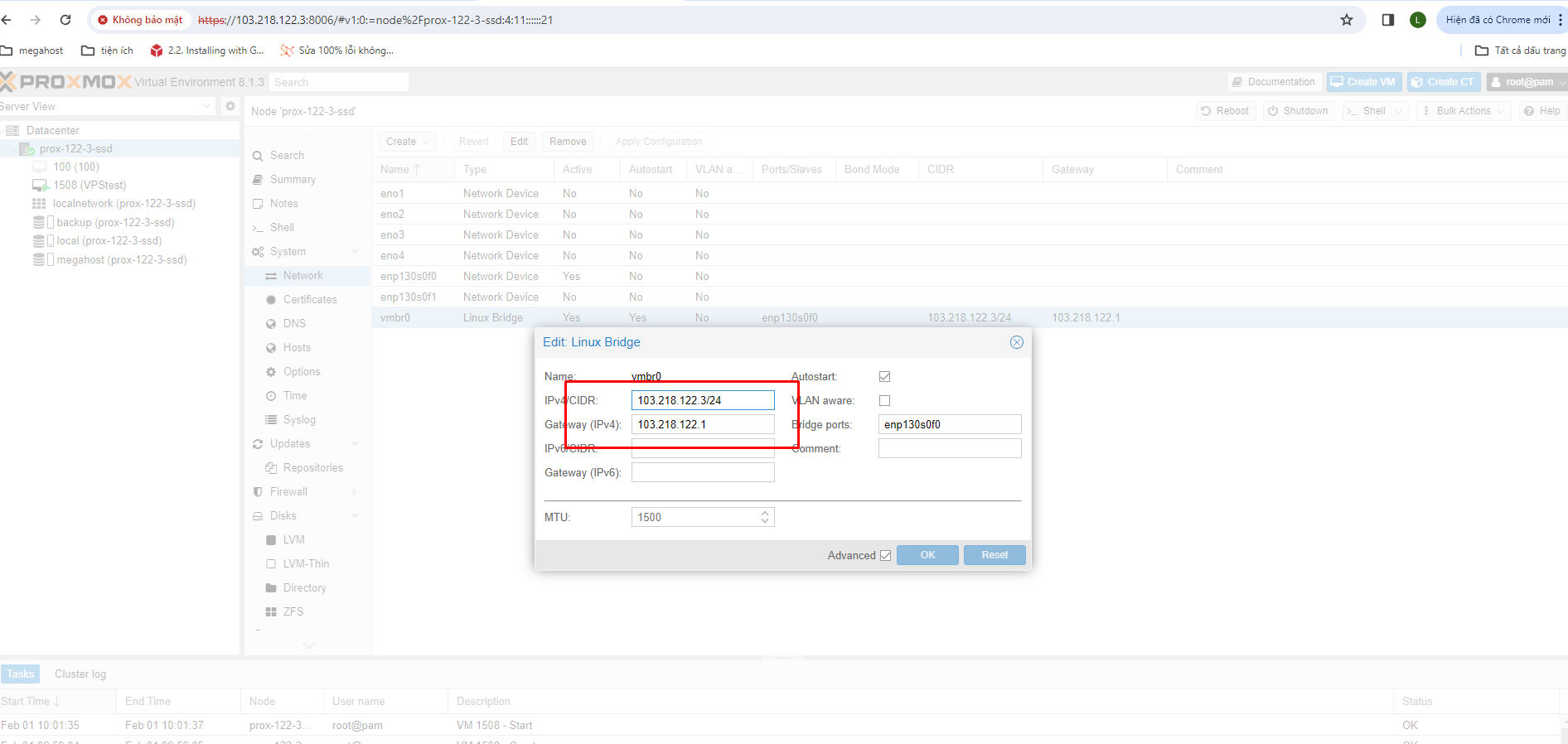
Đổi ip của virtualizor + proxmox

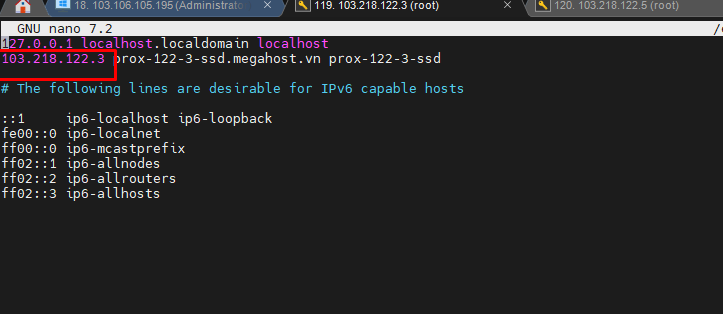
Đổi hostname

1-Virtualizor :

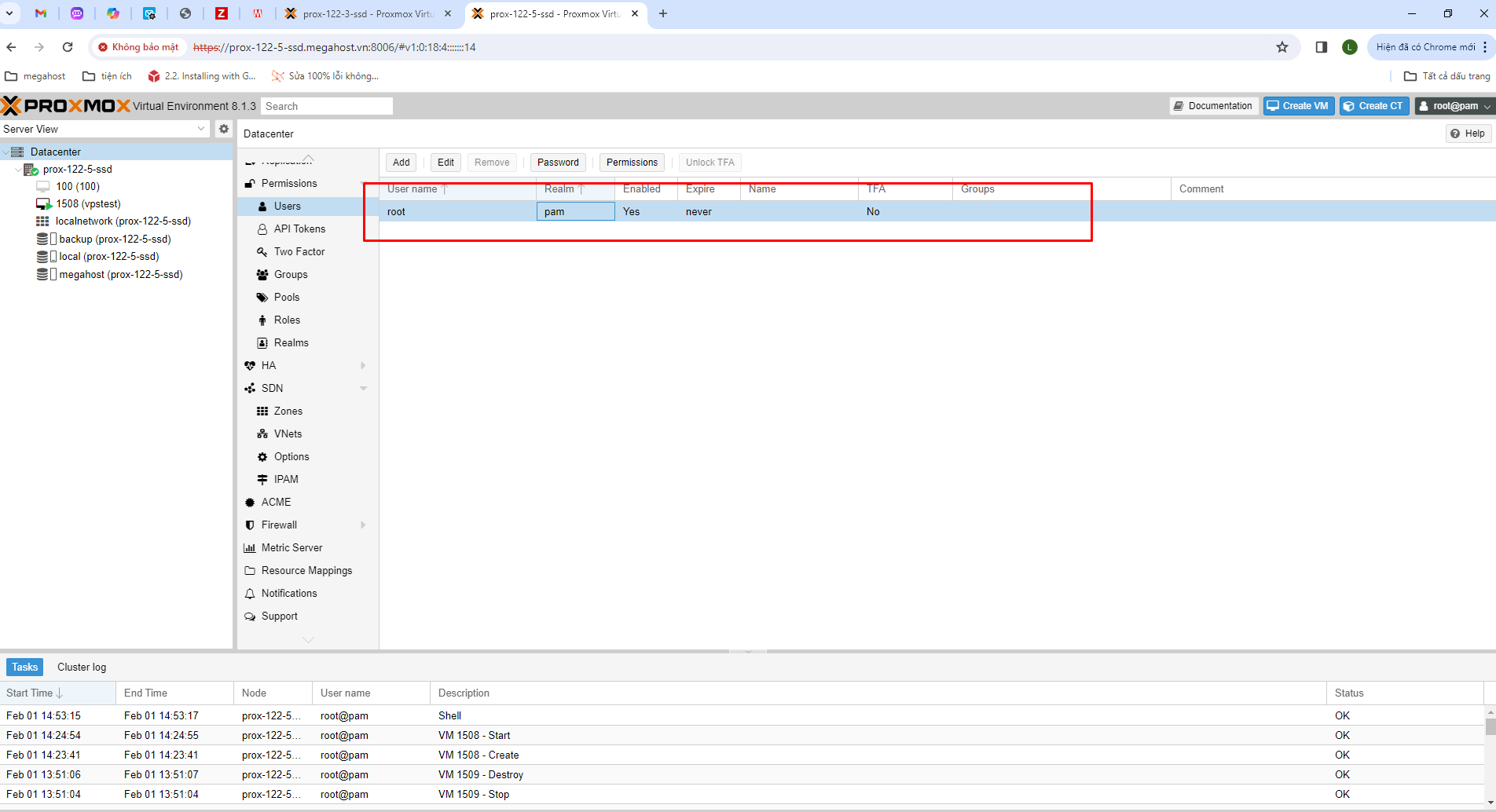


nano /etc/hosts





Xoá gửi mail từ proxmox spam :



Add script check cho NODE

mkdir /script

nano /script/zpool\_status.sh

#!/bin/bash

# Biến chứa thông tin về bot Telegram

TELEGRAM\_BOT\_TOKEN="6578540130:AAGWWi7OAYFQp8mvdCossP3U2T8sCS6TFMI"

TELEGRAM\_CHAT\_ID="-996434742"

# Lấy tên máy chủ

HOSTNAME=$(hostname)

# Kiểm tra trạng thái zpool

ZPOOL\_STATUS=$(zpool status)

ZPOOL\_STATE=$(echo "$ZPOOL\_STATUS" | grep "state:" | awk '{print $2}')

# Kiểm tra nếu trạng thái khác "ONLINE"

if [ "$ZPOOL\_STATE" != "ONLINE" ]; then

# Tạo nội dung thông báo bao gồm tên máy chủ và trạng thái zpool

MESSAGE="🆘 Node: ${HOSTNAME} Status: ${ZPOOL\_STATUS}"

# Gửi thông báo qua Telegram

send\_to\_telegram() {

local message=$1

curl -s -X POST "https://api.telegram.org/bot${TELEGRAM\_BOT\_TOKEN}/sendMessage" \

-d chat\_id="${TELEGRAM\_CHAT\_ID}" \

-d text="${message}" \

-d parse\_mode="Markdown"

}

# Gọi hàm send\_to\_telegram với nội dung là trạng thái zpool

send\_to\_telegram "${MESSAGE}"

fi

nano /script/clear\_backup.sh

#!/bin/bash

# Đường dẫn tới thư mục cần kiểm tra

BACKUP\_DIR="/backup/dump"

#BACKUP\_DIR2="/backup2/dump"

# Số ngày cần kiểm tra

DAYS=30

# Thông tin Bot Telegram

BOT\_TOKEN="6578540130:AAGWWi7OAYFQp8mvdCossP3U2T8sCS6TFMI"

CHAT\_ID="-996434742"

# Lấy giá trị HOSTNAME

HOSTNAME=$(hostname)

# Thông điệp thông báo

MESSAGE="Đã xóa tất cả các file quá $DAYS ngày trong thư mục BACKUP trên $HOSTNAME"

# Xóa các file quá ngày và ghi log

FILES\_DELETED=$(find "$BACKUP\_DIR" -type f -mtime +$DAYS -exec rm -v {} \;)

#FILES\_DELETED=$(find "$BACKUP\_DIR2" -type f -mtime +$DAYS -exec rm -v {} \;)

# Kiểm tra nếu có file bị xóa

if [ -n "$FILES\_DELETED" ]; then

# Gửi thông báo tới Telegram

curl -s -X POST "https://api.telegram.org/bot$BOT\_TOKEN/sendMessage" -d chat\_id=$CHAT\_ID -d text="$MESSAGE"

else

# Gửi thông báo không có file nào bị xóa

curl -s -X POST "https://api.telegram.org/bot$BOT\_TOKEN/sendMessage" -d chat\_id=$CHAT\_ID -d text="Không có file nào quá $DAYS ngày trong thư mục BACKUP để xóa trên $HOSTNAME"

fi

nano /script/checklogin.sh

#!/bin/bash

# Telegram chat ID và token

CHAT\_ID="-4160958075"

TOKEN="6895998420:AAGVV\_plqG9-wsgWO8n5dYON4Mx31saRRXc"

# Hàm gửi tin nhắn qua Telegram

send\_telegram\_message() {

local message="$1"

# Kiểm tra nếu tin nhắn trùng lặp với tin nhắn trước đó thì không gửi

if [[ "$SENT\_MESSAGE" != "$message" ]]; then

curl -s -X POST "https://api.telegram.org/bot$TOKEN/sendMessage" -d "chat\_id=$CHAT\_ID" -d "text=$message"

# Lưu trạng thái của tin nhắn gửi đi

SENT\_MESSAGE="$message"

fi

}

# Hàm kiểm tra sự kiện đăng nhập SSH

checklogin() {

ssh\_log=$(journalctl \_COMM=sshd --since "1 hour ago")

if [[ $ssh\_log =~ "Accepted password for root from" ]]; then

ip\_address=$(echo "$ssh\_log" | grep -oE "\b([0-9]{1,3}\.){3}[0-9]{1,3}\b" | awk 'NR==1')

send\_telegram\_message "New login SSH to 103.218.122.12 from IP: $ip\_address"

fi

}

# Vòng lặp vô hạn

while true; do

checklogin

sleep 10

done

else

echo "Dòng net0: không tồn tại trong $CONFIG\_FILE."

fi

else

echo "File cấu hình $CONFIG\_FILE không tồn tại hoặc không thể đọc được."

fi

done

chmod +x /script/zpool\_status.sh

chmod +x /script/clear\_backup.sh

chmod +x /script/checklogin.sh

crontab -e

0 12 30 \* \* /script/clear\_backup.sh

0 7,19 \* \* \* /script/zpool\_status.sh

service cron restart

--- Check login

nano /etc/systemd/system/checklogin.service

[Unit]

Description=Continuous notification script

After=network.target

[Service]

ExecStart=/script/checklogin.sh

Restart=always

RestartSec=10

[Install]

WantedBy=multi-user.target

chmod +x /etc/systemd/system/checklogin.service

systemctl daemon-reload

systemctl enable checklogin

systemctl start checklogin