

Установка Proxmox VE на ZFS на примере Hetzner

https://www.youtube.com/watch?v=Szt_TE5AS68&t=162s

<https://community.hetzner.com/tutorials/proxmox-docker-zfs>

Rescue System

selecting "Linux x64"

Скачать ISO

<https://www.proxmox.com/en/downloads/proxmox-virtual-environment/iso>

```
wget https://enterprise.proxmox.com/iso/proxmox-ve_8.4-1.iso
```

Запустить виртуальную машину на Rescue System

Команда для запуска VM будет отличаться для каждого сервера. Она в основном зависит от количества имеющихся у вас дисков. Чтобы увидеть имена всех ваших дисков, выполните команду `lsblk`.

BIOS

nvme

```
qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./proxmox-ve_8.4-1.iso -drive file=/dev/nvme0n1,format=raw,media=disk -drive file=/dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1
```

sda,sdb

```
qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./proxmox-ve_8.4-1.iso -drive file=/dev/sda,format=raw,media=disk -drive file=/dev/sdb,format=raw,media=disk -vnc 127.0.0.1:1
```

Аккуратно с пробелами!!!

Для 4х дисков

```
qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./proxmox-ve_8.3-1.iso -drive file=/dev/sda,format=raw,if=virtio -drive file=/dev/sdb,format=raw,if=virtio -drive file=/dev/sdc,format=raw,if=virtio -drive file=/dev/sdd,format=raw,if=virtio -vnc 127.0.0.1:1
```

UEFI

Для использования режима UEFI с QEMU нужна OVMF прошивка, которая включается в команду запуска через `-bios`:

```
apt install ovmf
```

Прошивка появится здесь: `/usr/share/ovmf/OVMF.fd`

```
qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./proxmox-ve_8.4-1.iso -drive file=/dev/nvme0n1,format=raw,media=disk -drive file=/dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1 -bios /usr/share/ovmf/OVMF.fd
```

После того как Proxmox будет установлен, проверьте, что система действительно загружается в UEFI-режиме, введя:

```
ls /sys/firmware/efi
```

Без DVD

```
qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -drive  
file=/dev/nvme0n1,format=raw,media=disk -drive  
file=/dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1 -bios  
/usr/share/ovmf/OVMF.fd
```

[Подключиться к виртуальной машине](#)

```
ssh -L 8888:127.0.0.1:5901 root@188.40.102.187
```

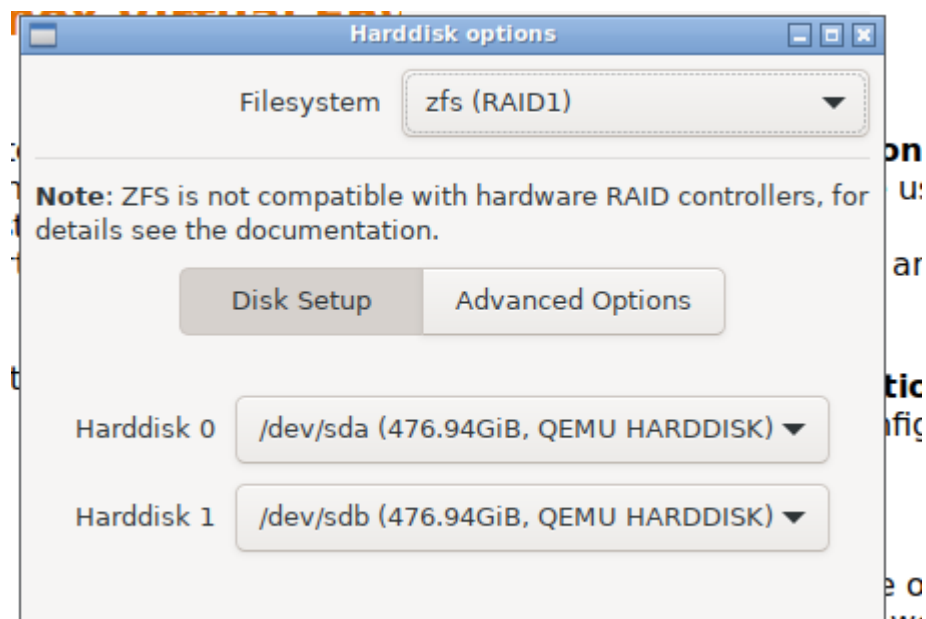
Затем с помощью VNC-клиента по вашему выбору подключитесь к хосту 127.0.0.1 и порту 8888. Вы сможете увидеть свою виртуальную машину.

Если сервер перенисталивал, то на Windows 10/11

```
del \Users\exch_adm\.ssh\known_hosts
```

[Installing Proxmox](#)

Выбираем Raid1



Country

Time zone

Keyboard Layout

Можно сразу написать правильный IP

На rescue

1. Выяснить IP и маску

```
ip a | grep eth0 | grep inet
```

```
root@rescue ~ # ip a | grep eth0 | grep inet
inet 144.76.118.246/27 scope global eth0
```

2. Выяснить Шлюз

```
ip route | grep default
```

```
root@rescue ~ # ip route | grep default
default via 144.76.118.225 dev eth0
```

Network Interface ☒ ens3 - 52:54:00:12:34:56 (e1000)

Hostname (FQDN)

IP Address (CIDR) /

Gateway

DNS Server

После установки Виртуалка опять начнет заражаться с CD-ROM. Нам нужно ее остановить и отключить CD-ROM

Переключается в SSH

```

└─md2          9:2      0 443.8G  0 raid1
root@rescue ~ # ls
configs hwcheck-logs images proxmox-ve_8.2-1.iso
root@rescue ~ # qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./
proxmox-ve_8.2-1.iso -drive file=/dev/nvme0n1,format=raw,media=disk -drive file=
/dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1
qemu-system-x86_64: -cdrom ./: 'file' driver requires './' to be a regular file
root@rescue ~ # qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -cdrom ./
roxmox-ve_8.2-1.iso -drive file=/dev/nvme0n1,format=raw,media=disk -drive file=
dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1

```

Ctrl+C

```

qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -drive
file=/dev/nvme0n1,format=raw,media=disk -drive
file=/dev/nvme1n1,format=raw,media=disk -vnc 127.0.0.1:1

```

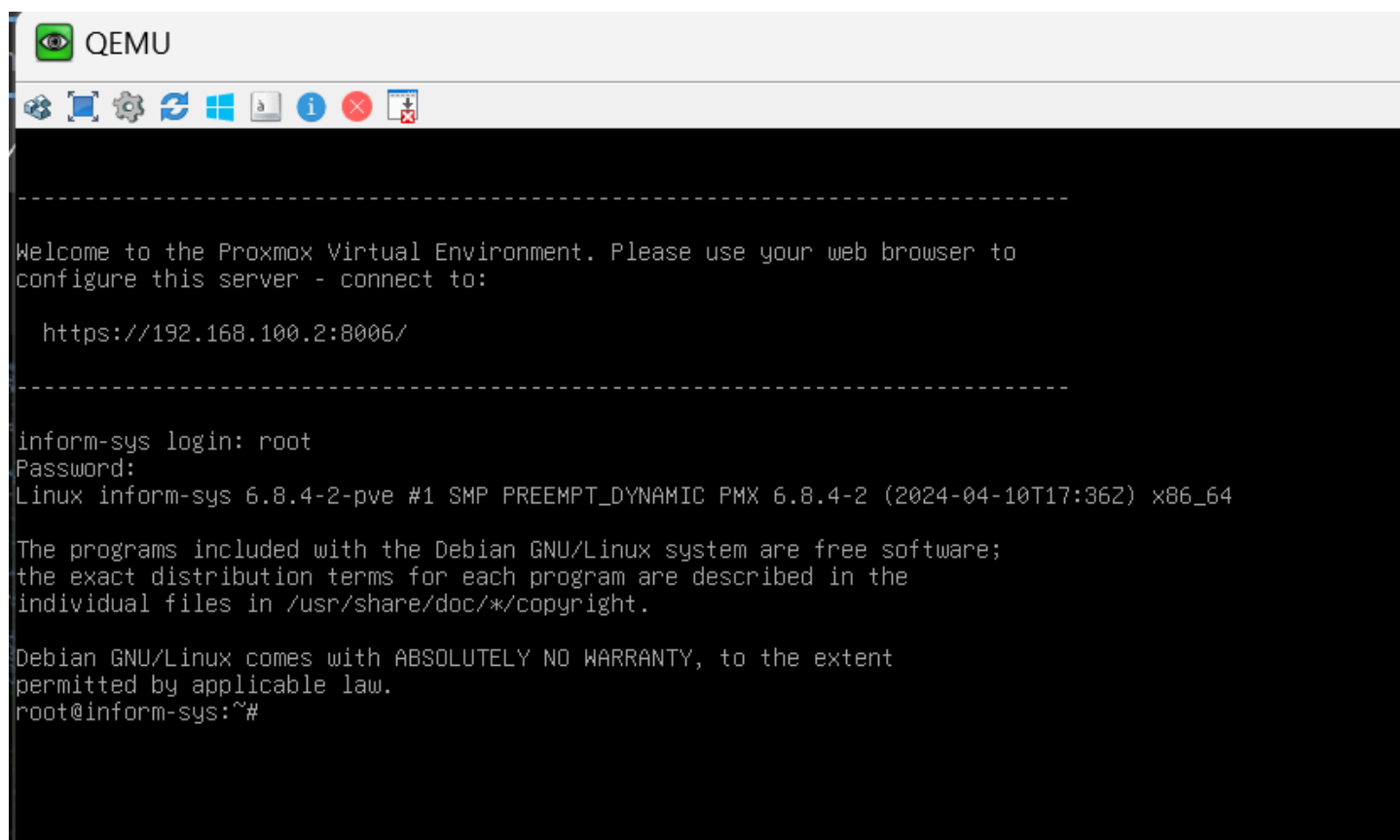
или

```

qemu-system-x86_64 -enable-kvm -smp 4 -m 4096 -boot d -drive
file=/dev/sda,format=raw,media=disk -drive file=/dev/sdb,format=raw,media=disk -
vnc 127.0.0.1:1

```

Вижу в VNC



Сконфигурировать сеть Proxmox:

На rescue.

Выяснить Имя сетевого интерфейса.

```

root@rescue ~ # ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast
    link/ether 90:1b:0e:b6:e7:05 brd ff:ff:ff:ff:ff:ff
    altname enp0s31f6
    inet 138.201.250.74/26 scope global eth0
        valid_lft forever preferred_lft forever

```

У нас он eth0

Выясняем, какое имя интерфейса будет на Proxmox

То есть

```

udevadm info -q all -p /sys/class/net/eth0 | grep ID_NET_NAME

```

```

root@rescue ~ # udevadm info -q all -p /sys/class/net/eth0 | grep ID_NET_NAME
E: ID_NET_NAME_MAC=enx901b0eb6e705
E: ID_NET_NAME_PATH=enp0s31f6
E: ID_NET_NAME=eth0

```

То есть имя интерфейса **enp0s31f6**, который мы будем конфигурировать в Proxmox

Идем в Proxmox и конфигурируем сеть nano /etc/network/interfaces для **enp0s31f6** (там будет другой интерфейс, в данном случае ens3)

Для серверов supermicro используем ID_NET_NAME_ONBOARD=**eno2**

```

root@rescue ~ # udevadm info -q all -p /sys/class/net/eth0 | grep ID_NET_NAME
E: ID_NET_NAME_MAC=enx0cc47ae7d2f2
E: ID_NET_NAME_ONBOARD=eno2
E: ID_NET_NAME_PATH=enp6s0
E: ID_NET_NAME=eth0

```

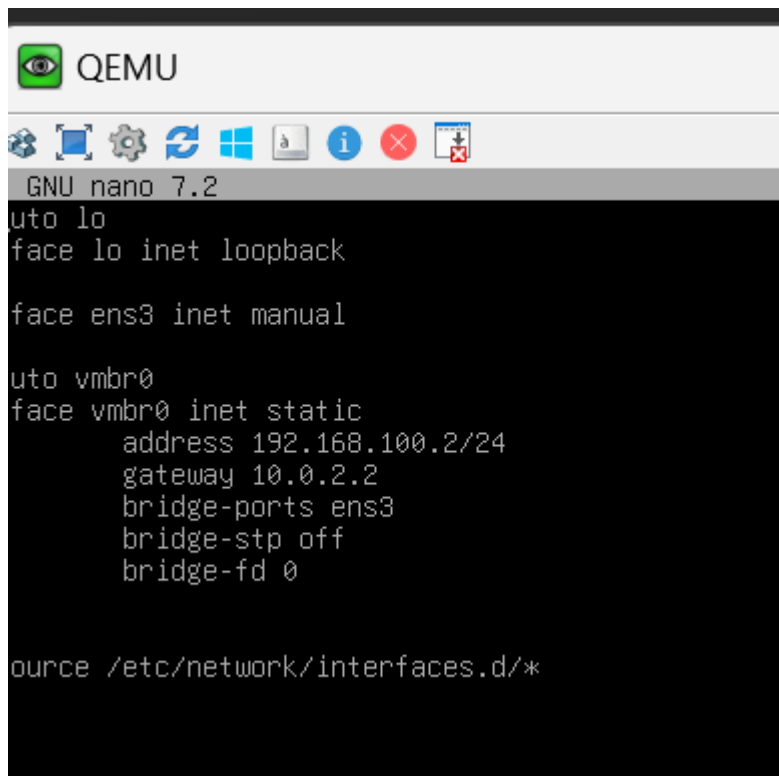
или

```

root@rescue ~ # ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 0c:9d:92:c3:fb:b8 brd ff:ff:ff:ff:ff:ff
    altname enp5s0
    inet 95.216.229.41/26 scope global eth0
        valid_lft forever preferred_lft forever
    inet6 2a01:4f9:2b:2268::2/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::e9d:92ff:fec3:fbb8/64 scope link
        valid_lft forever preferred_lft forever

```

Мы видим



```
QEMU
GNU nano 7.2
auto lo
face lo inet loopback

face ens3 inet manual

auto vbr0
face vbr0 inet static
    address 192.168.100.2/24
    gateway 10.0.2.2
    bridge-ports ens3
    bridge-stp off
    bridge-fd 0

source /etc/network/interfaces.d/*
```

Меняем ens3 на **enp0s31f6** в 2х местах!!!

Ошибка failed to import pool rpool

```
Begin: Sleeping for ... done.
Begin: Importing ZFS root pool 'rpool' ... Begin: Importing pool 'rpool' using defaults ... Failure:
1
Failure: 1

Command: /sbin/zpool import -N 'rpool'
Message: cannot import 'rpool': no such pool available
Error: 1

Failed to import pool 'rpool'.
Manually import the pool and exit.

BusyBox v1.30.1 (Debian 1:1.30.1-4) built-in shell (ash)
Enter 'help' for a list of built-in commands.

(initramfs)
```

<https://mertviyprorok.com/articles/instructions/sistemy-virtualizacii/proxmox-failed-to-import-pool-rpool-na-zfs-2-0-i-ranshe/>

Система не может импортировать rpool и, соответственно, не может загрузить Proxmox

Чтобы импортировать пул и загрузить систему нам нужно выполнить команду:

```
zpool import -N rpool
```

Чтобы выйти из initramfs нажимаем ctrl+D и система загрузится.

Проблема в том, что при следующей загрузке системы ситуация повторится.

Для решения проблемы нам нужно отредактировать файл по пути /etc/default/zfs:

```
nano /etc/default/zfs
```

Нужно найти строчку

```
ZFS_INITRD_PRE_MOUNTROOT_SLEEP=' 0 '
```

Изменить 0 на 5 к примеру.

Эта переменная показывает сколько секунд ожидать перед импортом пула.

В Proxmox версии 6.3-4, zfs обновляется до версии 2.0.3 и файл zfs меняется.

В нём нет строчки `ZFS_INITRD_PRE_MOUNTROOT_SLEEP='0'`, но она по-прежнему работает. Поэтому нам нужно просто добавить строчку

```
ZFS_INITRD_PRE_MOUNTROOT_SLEEP=' 5 '
```

После этого нам нужно обновить initramfs для ядра, чтобы изменения применились:

```
update-initramfs -u
```