

Proxmox Virtual Environment (VE)

Diegimas, konfigūravimas ir pirmieji žingsniai

Turinys

1.	Kas yra Proxmox VE	2
2.	Proxmox VE diegimas	2
2.1	Reikalavimai	2
2.2	Diegimo ISO paruošimas	2
2.3	Diegimo eiga	2
3.	Pradinė PVE konfigūracija po diegimo	3
3.1	Prisijungimas prie Web GUI	3
3.2	Enterprise repo išjungimas (lab environment)	3
3.3	Host bazinė konfigūracija	3
4.	PVE GUI struktūros supratimas	3
4.1	Kairysis medis (Datacenter view)	3
4.2	Datacenter lygis	3
4.3	Node lygis	3
5.	Diskų ir Storage valdymas	4
5.1	Diskai (Physical)	4
5.2	Storage tipai	4
5.3	Naujo storage pridėjimas (Directory)	4
6.	Network konfigūravimas PVE	4
6.1	Linux bridge koncepcija	4
6.2	Network per GUI	4
6.3	CLI konfigūracija	4
7.	Pirmojo LXC konteinerio paleidimas	5
7.1	Kodėl LXC	5
7.2	Template atsiuntimas	5
7.3	LXC kūrimas per GUI	5
7.4	Paleidimas ir prisijungimas	5
7.5	Bazinė konfigūracija LXC viduje	6
8.	Dažnos pradedančiųjų klaidos	6
9.	Rekomenduojami tolimesni žingsniai	6

Žemiau pateikiama **mokomoji medžiaga**, parengta **sistemų administravimo eksperto** lygiu. Tekstas skirtas pradedančiam–vidutinio lygio administratoriui, kuris nori **teisingai išsidiegti ir suprasti Proxmox VE (PVE)**, jo architektūrą ir kasdienį valdymą.

Proxmox VE: diegimas, konfigūravimas ir pirmieji žingsniai

1. Kas yra Proxmox VE

Proxmox Virtual Environment (PVE) – tai atviro kodo **virtualizacijos platforma**, jungianti:

- **KVM/QEMU** – pilnoms virtualioms mašinoms (VM)
- **LXC** – lengviems konteineriams
- **Web GUI + CLI**
- Integruotą **backup, storage, network, HA** (klasteryje)

PVE veikia ant **Debian Linux** bazės.

2. Proxmox VE diegimas

2.1 Reikalavimai

Minimalūs:

- 64-bit CPU su **VT-x / AMD-V**
- 8 GB RAM (rekomenduojama ≥ 16 GB)
- SSD (labai rekomenduojama)
- Bent 1 tinklo interfeisas

BIOS/UEFI nustatymai (kritiška):

- Enable: *Intel Virtualization Technology / SVM*
 - Disable: Secure Boot (nebūtina, bet supaprastina)
 - Storage: AHCI
-

2.2 Diegimo ISO paruošimas

1. Atsiųsti ISO:
<https://www.proxmox.com/en/downloads>
 2. Irašyti į USB (Rufus / BalenaEtcher)
 3. Paleisti serverį iš USB
-

2.3 Diegimo eiga

1. **Install Proxmox VE**
2. Sutikti su licencija
3. **Disk Configuration:**
 - Vienas diskas → ext4 (paprasčiausia)
 - Keli diskai → ZFS (jei suprantate pasekmes)
4. Region / Timezone
5. **Root slaptažodis + el. paštas**
6. **Network config:**
 - IP adresas (statinis!)
 - Hostname (pvz. pve01.lab.local)
 - Gateway
7. Install → Reboot

Po diegimo:

https://SERVER_IP:8006

3. Pradinė PVE konfigūracija po diegimo

3.1 Prisijungimas prie Web GUI

- Naršyklė → <https://IP:8006>
 - User: root
 - Realm: Linux PAM
-

3.2 Enterprise repo išjungimas (lab environment)

Pagal nutylėjimą PVE naudoja **Enterprise repo**, kuris reikalauja prenumeratos.

GUI būdu:

- Node → Updates → Repositories
- Disable: pve-enterprise
- Enable: pve-no-subscription

CLI:

```
sed -i 's/^deb/#deb/' /etc/apt/sources.list.d/pve-enterprise.list
echo "deb http://download.proxmox.com/debian/pve bookworm pve-no-subscription" \
> /etc/apt/sources.list.d/pve-no-subscription.list
```

```
apt update && apt full-upgrade -y
```

3.3 Host bazine konfigūracija

Rekomenduojama:

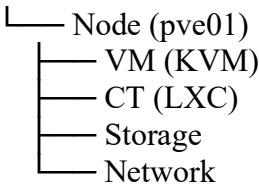
```
apt install -y vim htop curl net-tools
```

4. PVE GUI struktūros supratimas

4.1 Kairysis medis (Datacenter view)

Hierarchija:

Datacenter



4.2 Datacenter lygis

- **Permissions** – vartotojai, rolės
 - **Storage** – bendri storage
 - **Backup** – atsarginių kopijų planai
 - **HA** – klasteriams
-

4.3 Node lygis

Svarbiausi skyriai:

- **Summary** – CPU/RAM/Disk usage
- **Shell** – SSH per GUI
- **System** → **Network**
- **Disks**
- **Updates**

5. Diskų ir Storage valdymas

5.1 Diskai (Physical)

Node → Disks:

- **Disks** – visi fiziniai diskai
- **LVM**
- **ZFS**
- **Directory**

Patikra CLI:

lsblk

5.2 Storage tipai

Tipas	Paskirtis
-------	-----------

local	ISO, backup
-------	-------------

local-lvm	VM/LXC diskai
-----------	---------------

directory	Paprasti failai
-----------	-----------------

ZFS	Enterprise sprendimai
-----	-----------------------

NFS / CIFS	Network storage
------------	-----------------

5.3 Naujo storage pridėjimas (Directory)

GUI:

- Datacenter → Storage → Add → Directory
- Path: /mnt/storage1
- Content: Disk image, Container, ISO

CLI:

```
mkdir /mnt/storage1
```

6. Network konfigūravimas PVE

6.1 Linux bridge koncepcija

PVE naudoja **Linux Bridge**, pvz.:

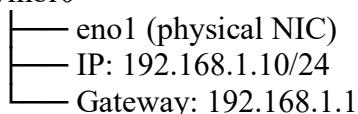
- vmbr0 – pagrindinis LAN
 - VM/LXC gauna IP per bridge
-

6.2 Network per GUI

Node → System → Network

Standartinė konfigūracija:

vmbr0



6.3 CLI konfigūracija

Failas:

```
/etc/network/interfaces
Pavyzdys:
auto vmbr0
iface vmbr0 inet static
    address 192.168.1.10/24
    gateway 192.168.1.1
    bridge-ports eno1
    bridge-stp off
    bridge-fd 0
```

Taikyti:
ifreload -a

7. Pirmojo LXC konteinerio paleidimas

7.1 Kodėl LXC

- Greitas
 - Minimalūs resursai
 - Idealu: DNS, Web, Monitoring, Docker host
-

7.2 Template atsiuntimas

GUI:

- Node → local → CT Templates → Templates → Download
- Pvz.: debian-12-standard

CLI:

```
pveam update
pveam available
pveam download local debian-12-standard_12.2-1_amd64.tar.zst
```

7.3 LXC kūrimas per GUI

Create CT

1. CT ID (pvz. 100)
 2. Hostname
 3. Template
 4. Disk size (pvz. 8 GB)
 5. CPU: 1–2
 6. RAM: 512–1024 MB
 7. Network:
 - Bridge: vmbr0
 - DHCP arba static
 8. Start after created → Finish
-

7.4 Paleidimas ir prisijungimas

GUI:

- CT → Console

CLI:

```
pct list
pct enter 100
```

7.5 Bazinė konfigūracija LXC viduje

```
apt update && apt upgrade -y  
apt install -y curl vim htop
```

8. Dažnos pradedančiųjų klaidos

1. DHCP vietoj statinio IP hostui
 2. VM naudojimas vietoje LXC paprastoms paslaugoms
 3. Backup nekonfigūravimas
 4. Netinkamas storage pasirinkimas
 5. Vienas NIC be VLAN planavimo
-

9. Rekomenduojami tolimesni žingsniai

- Backup server (PBS)
 - VLAN konfigūracija
 - Role-based access
 - Monitoring (Zabbix / Prometheus)
 - Cluster + HA
-