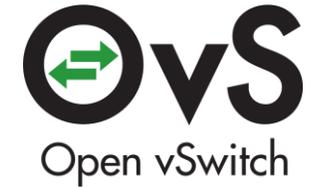




# VXLAN OVERLAY NETWORKS WITH OPEN VSWITCH



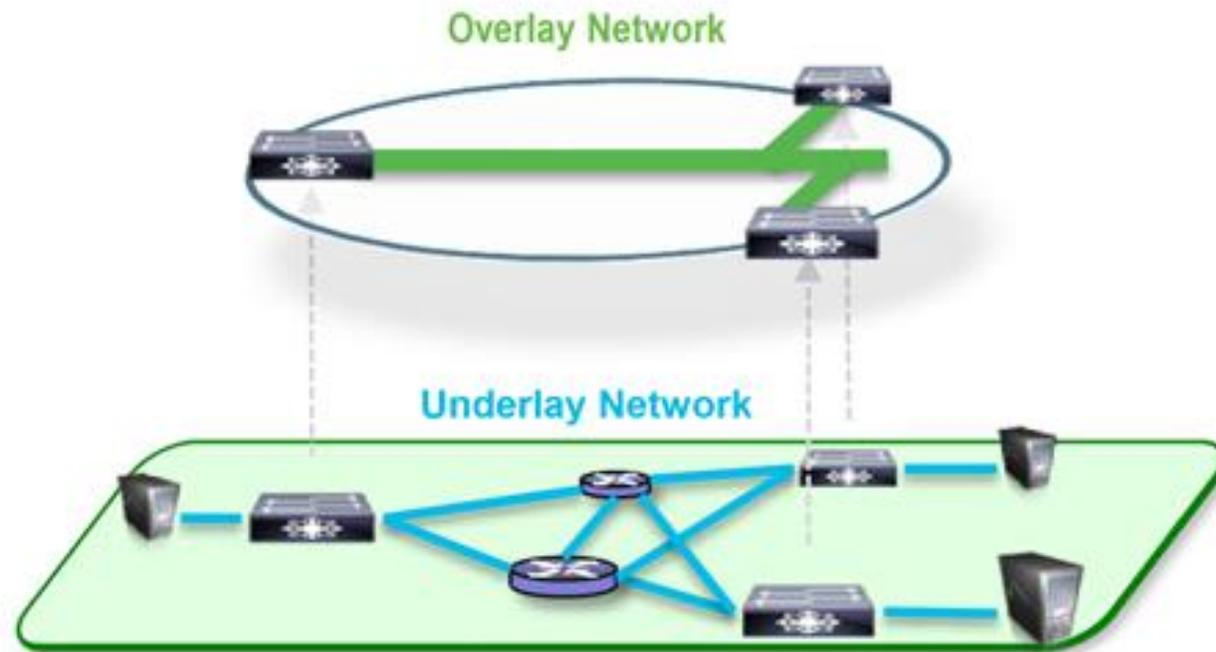
**I PUTU HARIYADI**

[putu.hariyadi@universitasbumigora.ac.id](mailto:putu.hariyadi@universitasbumigora.ac.id)

[www.iputuhariyadi.net](http://www.iputuhariyadi.net)

# UNDERLAY NETWORK VS OVERLAY NETWORK (1)

- **Underlay network** berhubungan dengan jaringan secara fisik (**physical network**).
- **Overlay network** merupakan jaringan logikal atau virtual (**virtual network**) yang berada diatas jaringan fisik.



Sumber gambar: [lookingpoint.com/blog/sd-access](http://lookingpoint.com/blog/sd-access)

## UNDERLAY NETWORK VS OVERLAY NETWORK (2)

BASIS OF COMPARISON	UNDERLAY NETWORK	OVERLAY NETWORK
<b>Description</b>	Underlay Network is physical infrastructure above which overlay network is built.	An Overlay Network is a virtual network that is built on top of an underlying Network infrastructure/Network layer (the underlay).
<b>Traffic Flow</b>	Transmits packets which traverse over network devices like switches and routers.	Transmits packets only along the virtual links between the overlay nodes.
<b>Deployment Time</b>	Less scalable and time consuming activity to setup new services and functions.	Ability to rapidly and incrementally deploy new functions through edge-centric innovations.
<b>Packet Control</b>	Hardware oriented.	Software oriented.

Sumber: [vivadifferences.com](http://vivadifferences.com)

## UNDERLAY NETWORK VS OVERLAY NETWORK (2)

BASIS OF COMPARISON	UNDERLAY NETWORK	OVERLAY NETWORK
<b>Packet Encapsulation And Overhead</b>	Packet delivery and reliability occurs at layer-3 and layer-4.	Needs to encapsulate packets across source and destination, hence incurs additional overhead.
<b>Multipath Forwarding</b>	Less scalable options of multipath forwarding. In fact using multiple paths can have associated overhead and complexity.	Support for multi-path forwarding within virtual networks.
<b>Managing Multitenancy</b>	NAT or VRF based segregation required which may face challenge in big environments.	Ability to manage overlapping IP addresses between multiple tenants.

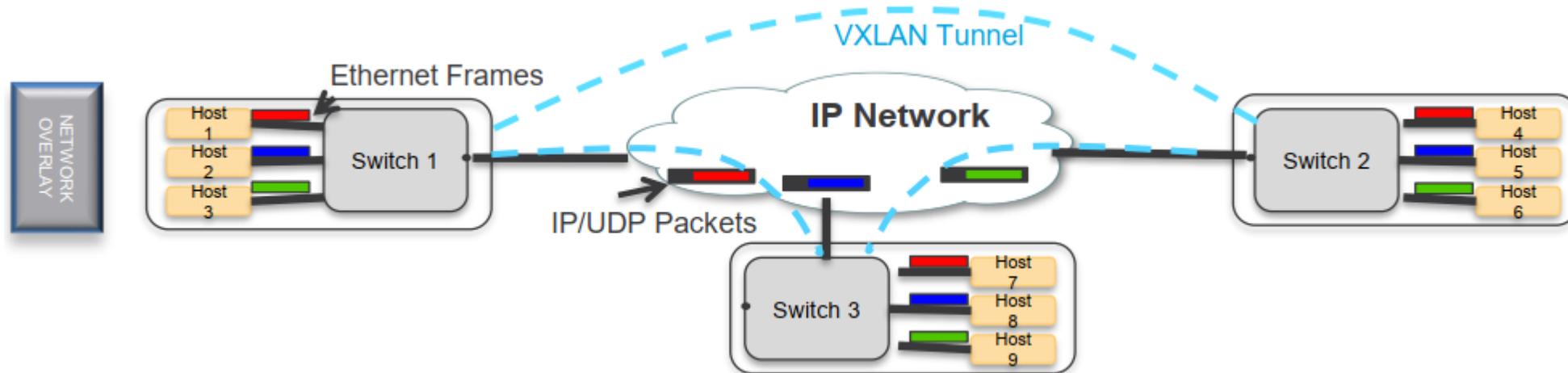
Sumber: [vivadifferences.com](http://vivadifferences.com)

## UNDERLAY NETWORK VS OVERLAY NETWORK (2)

BASIS OF COMPARISON	UNDERLAY NETWORK	OVERLAY NETWORK
<b>Scalability</b>	Less Scalable due to technology limitation.	Designed to provide more scalability than underlay network. E.g VLAN (Underlay Network) provides 4096 VLAN support while VXLAN (Overlay Network) provides up to 16 million identifiers.
<b>Packet Delivery</b>	Responsible for delivery of packets.	Offloaded from delivery of packets.
<b>Protocols</b>	Underlay protocols include Ethernet Switching, VLAN, Routing etc.	Overlay network protocols include Virtual Extensible LAN (VXLAN), Network Virtualization using Generic Encapsulation (NVGRE), Stateless Transport Tunneling (SST), Generic Routing Encapsulation (GRE), IP multicast and Network Virtualization overlays 3 (NVO3).

# PENGENALAN VIRTUAL EXTENSIBLE LOCAL AREA NETWORK (VXLAN)

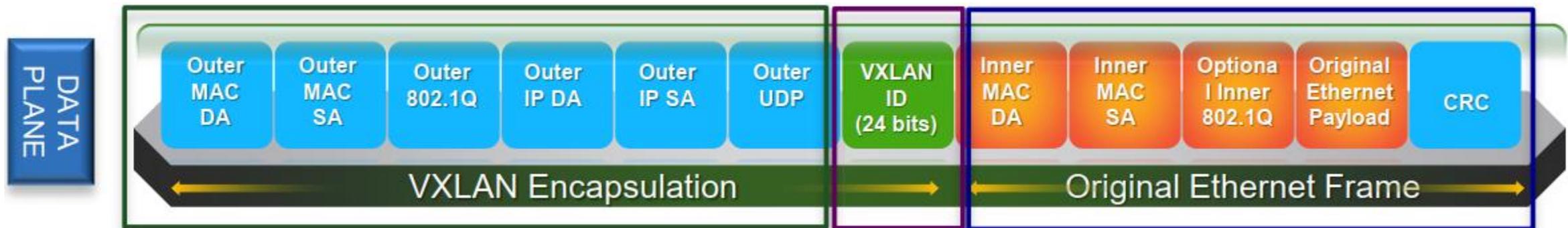
- Virtual Extensible Local Area Network (VXLAN) merupakan mekanisme **point-to-multipoint tunneling** untuk memperluas **layer 2 network** melewati **IP network**.



- Menurut RFC 7348, **VXLAN** menyediakan *layer 2 overlay network* di atas *layer 3 network*.
- Setiap *overlay* disebut dengan **VXLAN segment**.
- Virtual Machine (VM)** dapat berkomunikasi satu dengan yang lainnya hanya jika berada di **VXLAN segment** yang sama.

# PENGENALAN VIRTUAL EXTENSIBLE LOCAL AREA NETWORK (VXLAN)

- Setiap **VXLAN segment** diidentifikasi menggunakan **24 bit Segment ID** yang disebut dengan **VXLAN Network Identifier (VNI)**. Ini memungkinkan terdapat 16 juta VXLAN segment dalam satu domain administratif yang sama.
- **VNI** mengidentifikasi cakupan *frame* MAC bagian dalam (**inner MAC frame**) yang berasal dari VM individu.
- **VNI** berada di *header* luar (**outer header**) yang mengenkapsulasi MAC bagian dalam (**inner MAC frame**) yang berasal dari VM.
- **VXLAN** menggunakan **MAC** pada enkapsulasi **UDP** dengan nomor port tujuan (destination port) 4789.



# MENGAPA VXLAN OVERLAY?

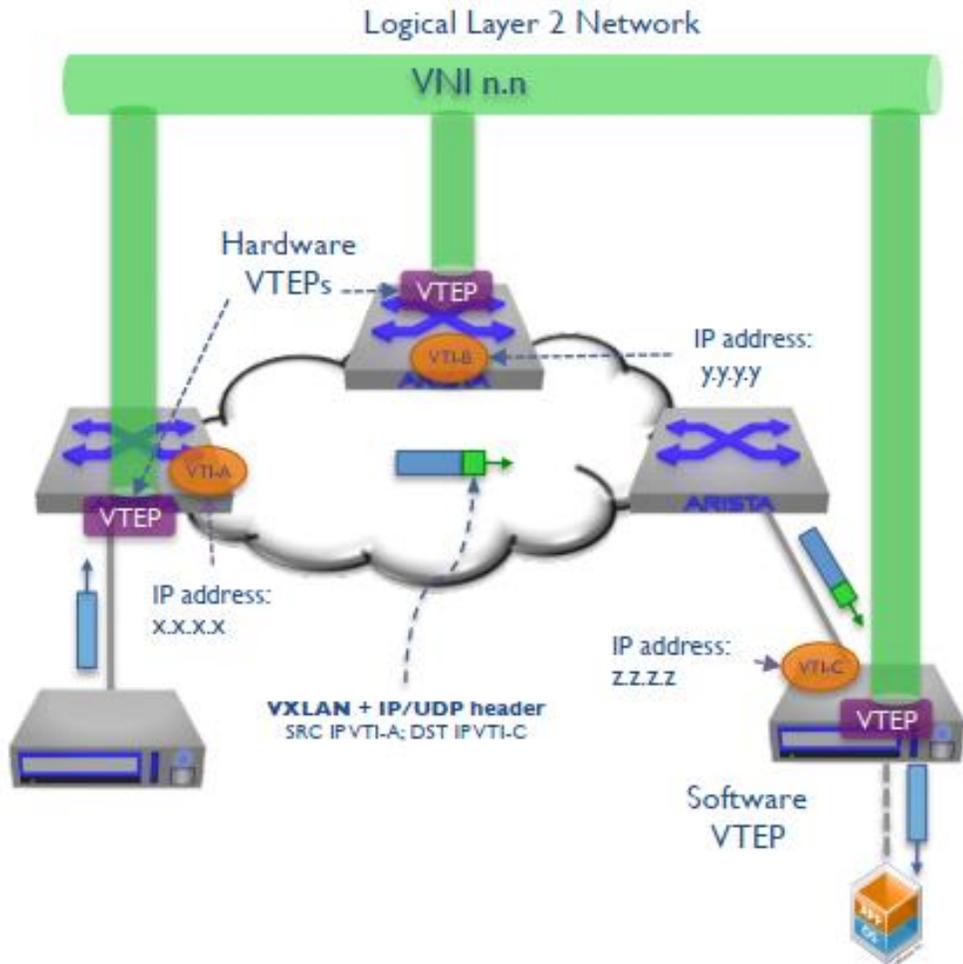
Customer Needs	VXLAN Delivered
Any workload anywhere – VLANs limited by L3 boundaries	Any Workload anywhere- across Layer 3 boundaries
VM Mobility	Seamless VM Mobility
Scale above 4k Segments (VLAN limitation)	Scale up to 16M segments
Efficient use of bandwidth	Leverages ECMP for optimal path usage over the transport network
Secure Multi-tenancy	Traffic & Address Isolation

# VLAN VS VXLAN

- VXLAN serupa dengan VLAN yang mengenkapsulasi *frame layer 2* dan mensegmentasi jaringan.
- Perbedaan utama VLAN dengan VXLAN adalah:

No.	VLAN	VXLAN
1.	Menggunakan Spanning Tree Protocol (STP) untuk memblokir redundant path sehingga hanya dapat memanfaatkan sebagian jalur yang tersedia.	Menggunakan protocol routing Layer 3 sehingga dapat memanfaatkan keseluruhan jalur yang tersedia melalui penerapan teknologi seperti Equal-Cost Multi-path Routing (ECMP).
2.	Skalabilitas kurang karena hanya mendukung 4094 VLAN sebagai dampak penggunaan 12 bit VLAN ID.	Skalabilitas tinggi karena mendukung 16.777.216 VXLAN segment sebagai dampak penggunaan 24 bit VNI.
3.	Kurang fleksibel untuk jaringan dengan multi-tenant.	Fleksibel untuk jaringan multi-tenant

# TERMINOLOGI PADA VXLAN



## Virtual Tunnel End Point (VTEP).

- Entitas yang melakukan enkapsulasi dan dekapsulasi VXLAN untuk *native frame*.
- Menambahkan *VXLAN header* yang sesuai.
- Dapat diimplementasikan pada *software virtual switch* atau *switch fisik*.

## Virtual Tunnel Identifier (VTI).

- IP dari *interface* yang digunakan sebagai alamat IP sumber untuk trafik VXLAN yang dienkapsulasi.
- Alamat IP tujuan untuk trafik VXLAN yang dienkapsulasi.

## Virtual Network Identifier (VNI).

- Field 24 bit yang ditambahkan di dalam *VXLAN Header* untuk mengidentifikasi segment *layer 2* dari *frame Ethernet* yang dienkapsulasi.

# DEPLOYING VXLAN

- **Host-based VXLAN.**

VXLAN yang diterapkan pada *host* seperti *hypervisor* yang mendukung virtual switch (*software*) seperti **Open vSwitch**.

Virtual Switch (vswitch) menjalankan fungsi VTEP dan mengenkapsulasi trafik yang datang dari VM sebelum diteruskan keluar melalui switch secara fisik.

Enkapsulasi VXLAN terjadi pada host sehingga infrastruktur jaringan lainnya hanya melihat trafik IP.

- **Gateway-based VXLAN.**

VXLAN yang diterapkan secara langsung pada perangkat keras (*hardware*) **switch** atau **router**.

Ketika switch dapat melakukan fungsi VTEP maka disebut sebagai VXLAN Gateway.

**Switch** dapat melakukan enkapsulasi dan dekapsulasi VXLAN serta mentranslasi VLAN ID ke VNI.

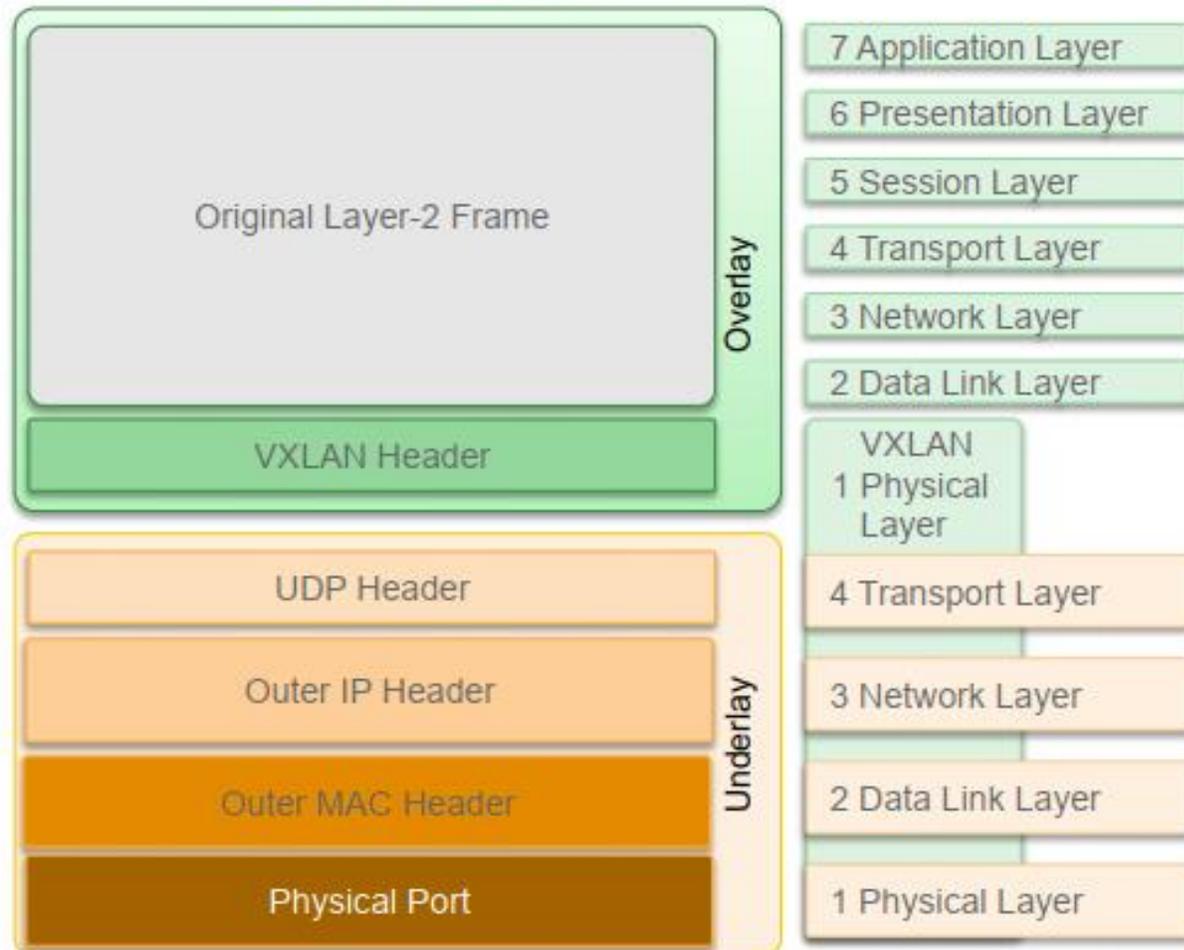
VXLAN Gateway membuat tunnel ke VTEP tujuan dapat berupa host atau gateway sehingga host atau infrastruktur IP tidak menyadari kehadiran VXLAN.

- **Hybrid Deployment.**

Campuran penerapan *Host-based* dan *Gateway-based VXLAN*.

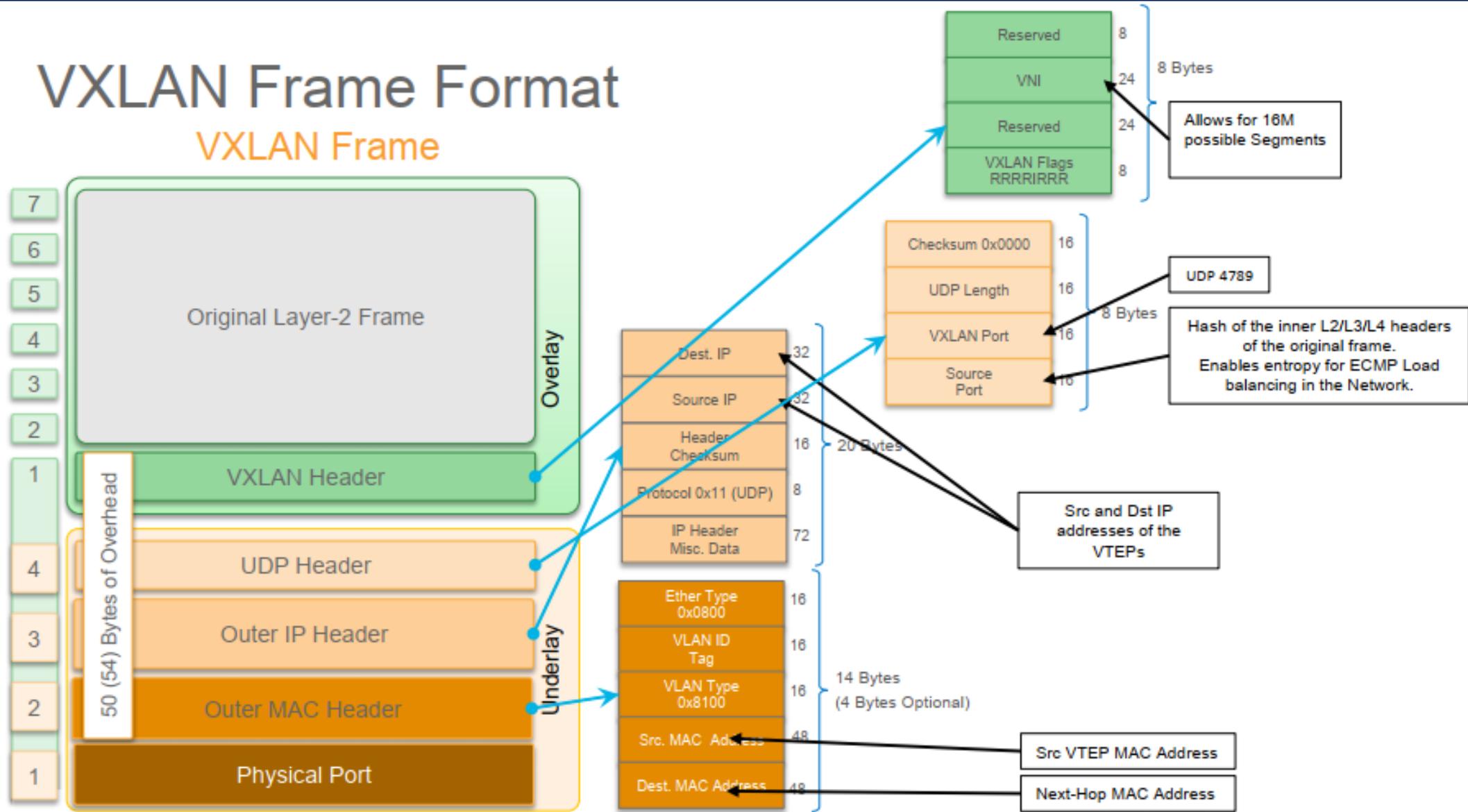
# VXLAN To OSI Model Correlation

## VXLAN Frame

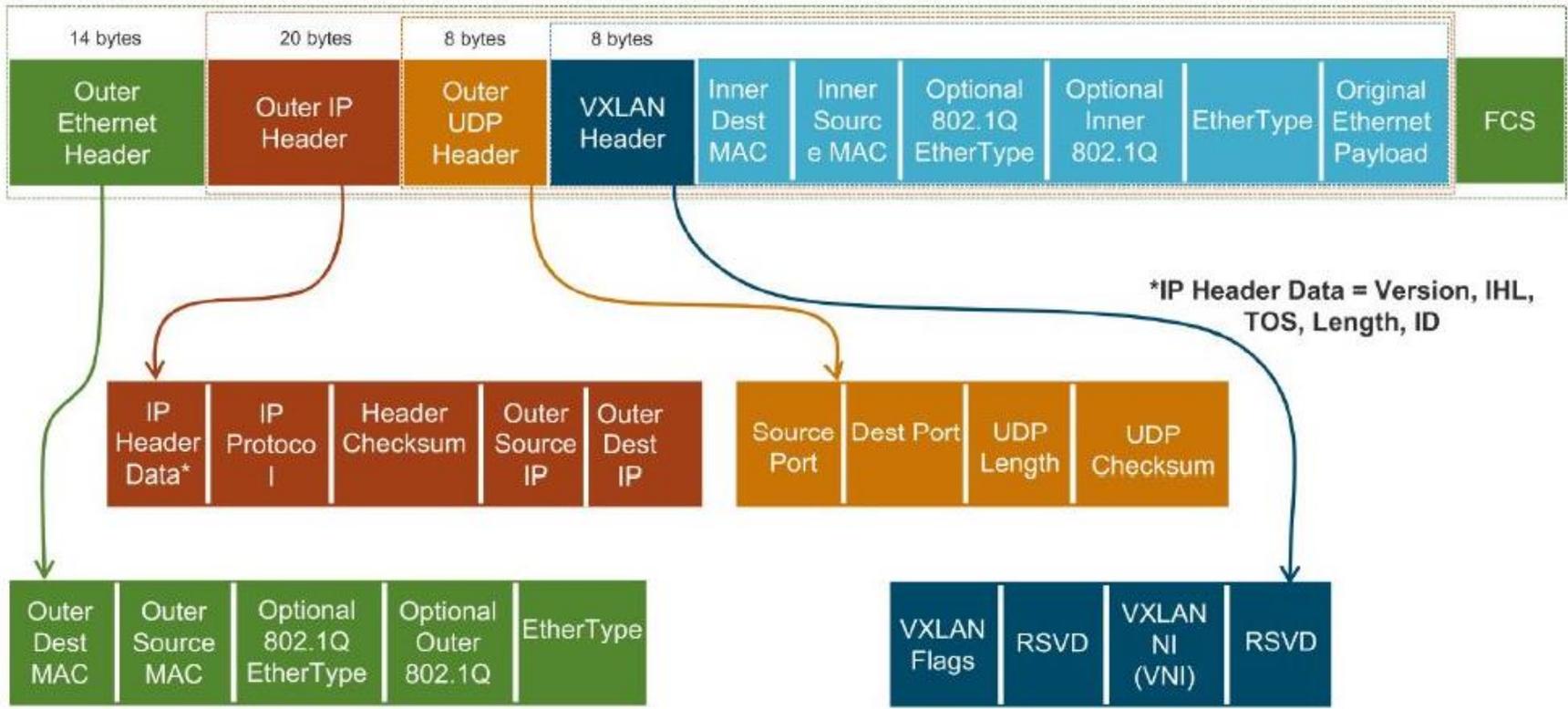
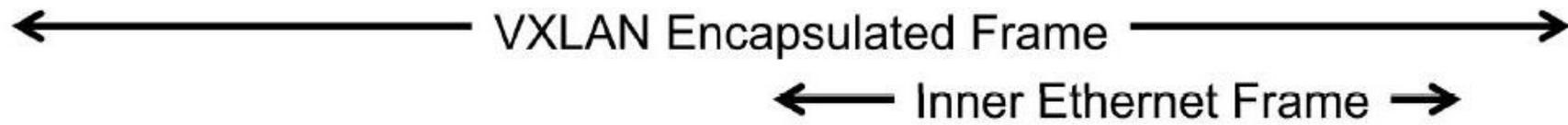


# VXLAN Frame Format

## VXLAN Frame



# VXLAN ENCAPSULATION



# CARA KERJA VXLAN (1)

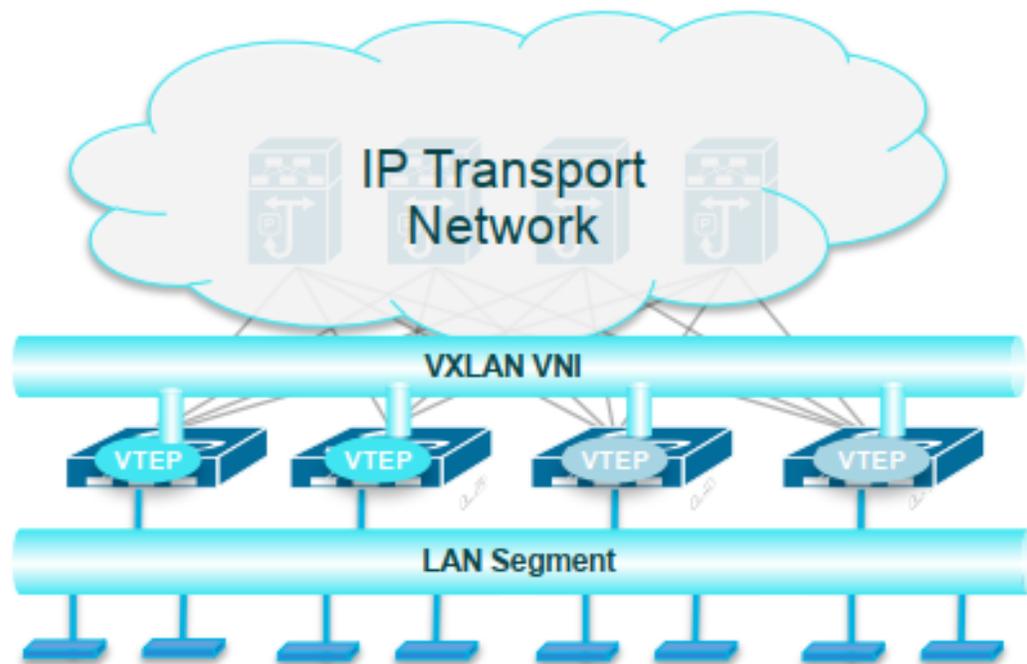
Gambar berikut memperlihatkan bagaimana proses komunikasi antar VM yang terdapat pada Hypervisor berbeda melalui **VXLAN Tunnel**. Hypervisor A dan B menggunakan **Open vSwitch** yang mendukung fitur **VXLAN** sehingga bertindak sebagai **host-based VXLAN**. Keseluruhan VM tidak mengetahui atau menyadari kehadiran **VXLAN**.



## CARA KERJA VXLAN (2)

1. VM1 yang terdapat pada Hypervisor A ingin berkomunikasi dengan VM4 yang terdapat pada Hypervisor B. Network adapter pada VM1 telah terkonfigurasi dengan VLAN ID tertentu. VM1 membuat frame layer 2 dan mengirimkannya ke *virtual switch port*.
2. Hypervisor A dengan peran sebagai VTEP memetakan VLAN ID sumber dengan VXLAN ID tujuan. VTEP menambahkan header VXLAN dan mengenkapsulasi frame layer 2 ke dalam paket layer 3 dan meneruskannya ke infrastruktur layer 3.
3. Infrastruktur layer 3 hanya melihat trafik IP dan tidak menyadari atau mengetahui terkait informasi VXLAN apapun. Trafik VXLAN di tunnel. Semua router di underlay network hanya melihat IP header sehingga akan meneruskannya.
4. Hypervisor B sebagai VTEP akan membuka paket dan menemukan informasi VXLAN didalamnya. Hypervisor B memahami VXLAN dan mengetahui pemetaan VXLAN ID ke VLAN ID dan menggunakan alamat MAC tujuan untuk meneruskan ke switchport yang sesuai.
5. VM4 menerima frame layer 2 yang dikirim dari VM1.

# NETWORK VIRTUALIZATION WITH VXLAN



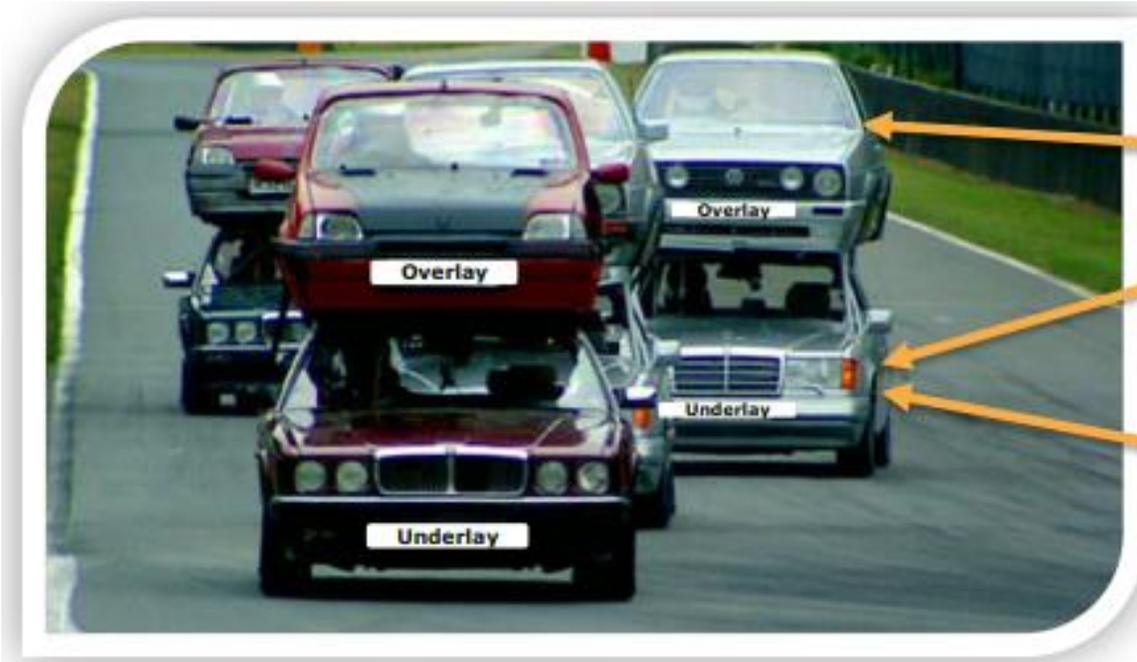
## Underlay Network:

- IP routing – proven, stable, scalable
- Support any routing protocols --- OSPF, EIGRP, IS-IS, BGP, etc.
- ECMP – utilize all available network paths

## Overlay Network:

- Standards-based overlay
- Layer-2 extensibility and mobility
- Expanded Layer-2 name space
- Scalable network domain
- Multi-Tenancy

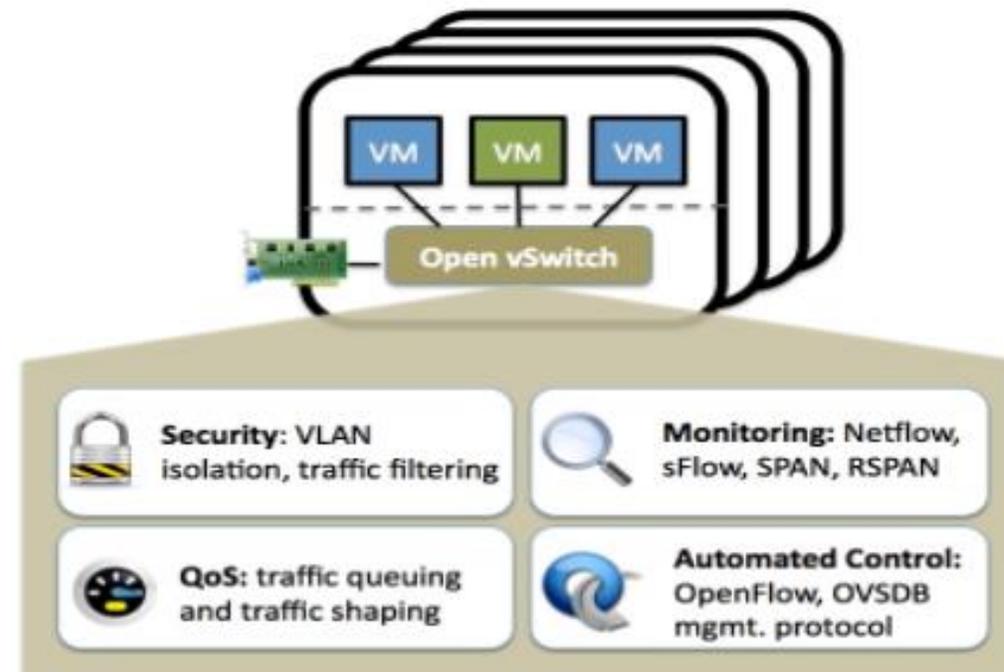
# UNDERLAYS AND OVERLAYS



- Host-to-Host connectivity
- VTEP-to-VTEP connectivity
- The underlay (Layer 1-4) is the “pseudo physical layer” (Layer 1) for the overlay.

# PENGENALAN OPEN VSWITCH (OVS)

- Open vSwitch (OVS) merupakan **multilayer virtual switch** yang didesain untuk digunakan pada lingkungan virtualisasi dan bersifat **open source**.
- OVS mendukung beragam fitur yang dapat dikategorikan menjadi 4 (empat) yaitu **security**, **monitoring**, **automated control** dan **QoS**.



Sumber gambar: [openvswitch.org](http://openvswitch.org)

# FITUR OPEN VSWITCH (1)

Menurut situs [openvswitch.org](http://openvswitch.org), fitur-fitur yang dimiliki oleh **Open vSwitch** meliputi:

- Visibility into inter-VM communication via NetFlow, sFlow(R), IPFIX, SPAN, RSPAN, and GRE-tunneled mirrors.
- LACP (IEEE 802.1AX-2008).
- **Standard 802.1Q VLAN model with trunking.**
- Multicast snooping.
- IETF Auto-Attach SPBM and rudimentary required LLDP support.
- BFD and 802.1ag link monitoring.
- STP (IEEE 802.1D-1998) and RSTP (IEEE 802.1D-2004).
- Fine-grained QoS control.
- Support for HFSC qdisc.

## FITUR OPEN VSWITCH (2)

- Per VM interface traffic policing.
- NIC bonding with source-MAC load balancing, active backup, and L4 hashing.
- OpenFlow protocol support (including many extensions for virtualization).
- IPv6 support.
- Multiple **tunneling protocols** (GRE, **VXLAN**, STT, and Geneve, with IPsec support).
- Remote configuration protocol with C and Python bindings.
- Kernel and user-space forwarding engine options.
- Multi-table forwarding pipeline with flow-caching engine.
- Forwarding layer abstraction to ease porting to new software and hardware platforms.



# **DEMO INSTALASI DAN KONFIGURASI VXLAN TUNNEL MENGUNAKAN OPEN VSWITCH PADA HYPERVISOR PROXMOX VIRTUAL ENVIRONMENT (PVE)**

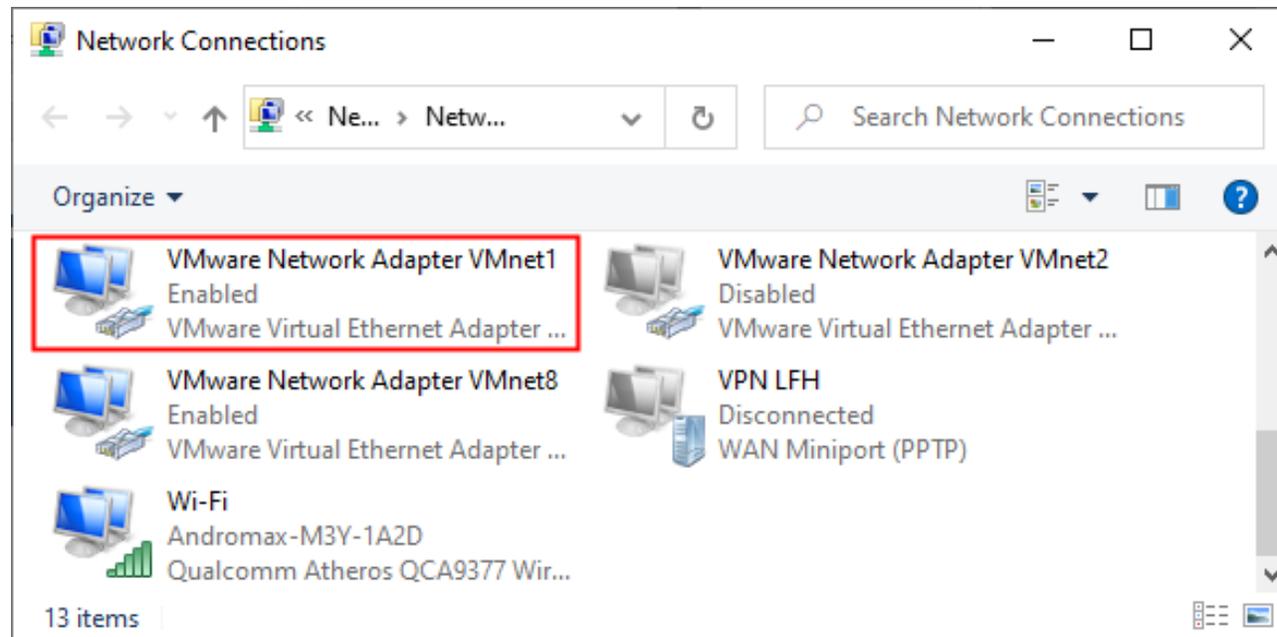




# KONFIGURASI PADA CLIENT WINDOWS 10

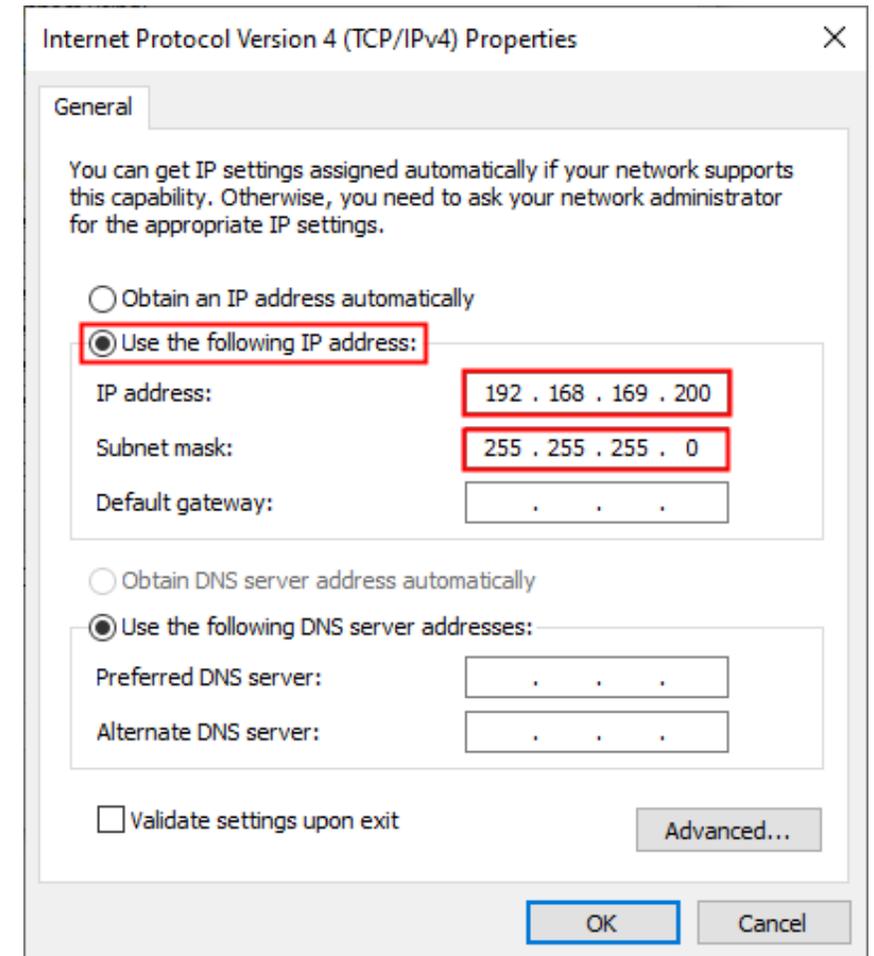
# MENGATUR PENGALAMATAN IP PADA VMWARE NETWORK ADAPTER VMNET1 DI CLIENT WINDOWS 10 (1)

- Lakukan pengaturan pengalamatan **IP** dan **Subnetmask** secara manual dari **VMware Network Adapter VMNet1** melalui **Control Panel > Network and Internet > Network and Sharing Center**.
- Pada kotak dialog **Network and Sharing** yang tampil, pilih **Change Adapter Settings**.
- Tampil kotak dialog **Network Connections**. Klik dua kali pada **VMware Network Adapter VMNet1**.



# MENGATUR PENGALAMATAN IP PADA VMWARE NETWORK ADAPTER VMNET1 DI CLIENT WINDOWS 10 (2)

- Pada kotak dialog **VMware Network Adapter VMnet1 Status** yang tampil, klik tombol **Properties**.
- Tampil kotak dialog **VMware Network Adapter VMnet1 Properties**. Pada bagian “**This connection uses the following items:**”, klik dua kali pada pilihan **Internet Protocol Version 4 (TCP/IPv4)**.
- Pada kotak dialog **Internet Protocol Version 4 (TCP/IPv4) Properties** yang tampil, pilih **Use the following IP Address**.
- Pada isian **IP address:**, masukkan **192.168.169.200**. Sedangkan pada isian **Subnet mask:**, masukkan **255.255.255.0**.
- Klik tombol **OK > OK > Close**.
- Tutup kotak dialog **Network and Sharing Center**.

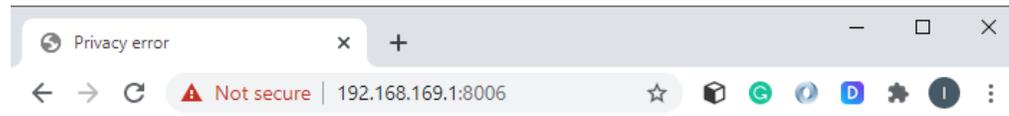




# KONFIGURASI PADA NODE PVE1

# MENGAKSES WEB GUI DARI NODE PVE1 (1)

- Buka **browser**, sebagai contoh menggunakan **Chrome**. Pada *address bar* dari *browser*, masukkan URL **https://192.168.169.1:8006**.



## Your connection is not private

Attackers might be trying to steal your information from **192.168.169.1** (for example, passwords, messages, or credit cards). [Learn more](#)

NET::ERR\_CERT\_AUTHORITY\_INVALID

Help improve security on the web for everyone by sending [URLs of some pages you visit, limited system information, and some page content](#) to Google. [Privacy policy](#)

Tampil pesan peringatan “**Your connection is not private**”.

Klik **Advanced** untuk melanjutkan pengaksesan.

Advanced

Back to safety

# MENGAKSES WEB GUI DARI NODE PVE1 (2)

- Klik link “**Proceed to 192.168.169.1 (unsafe)**”.



This server could not prove that it is **192.168.169.1**; its security certificate is not trusted by your computer's operating system. This may be caused by a misconfiguration or an attacker intercepting your connection.

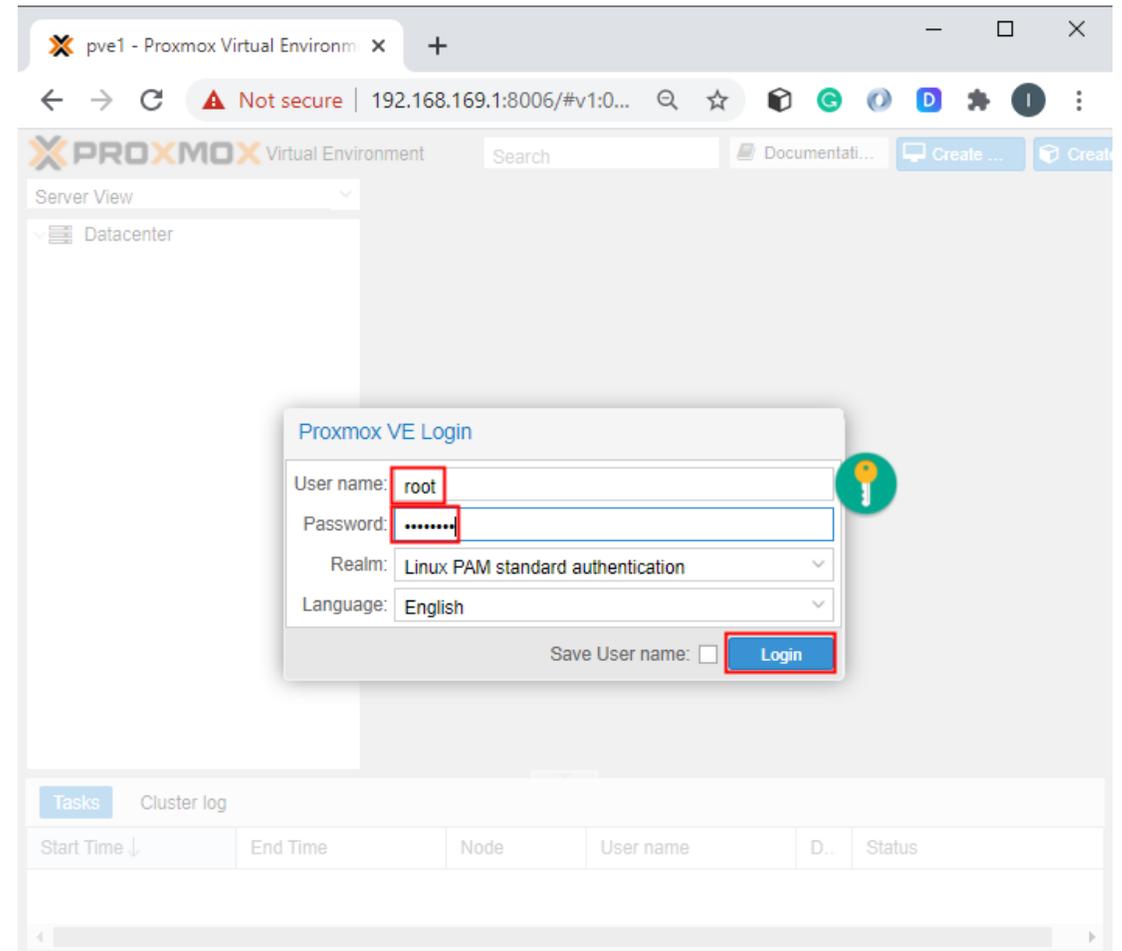
[Proceed to 192.168.169.1 \(unsafe\)](#)

Hide advanced

Back to safety

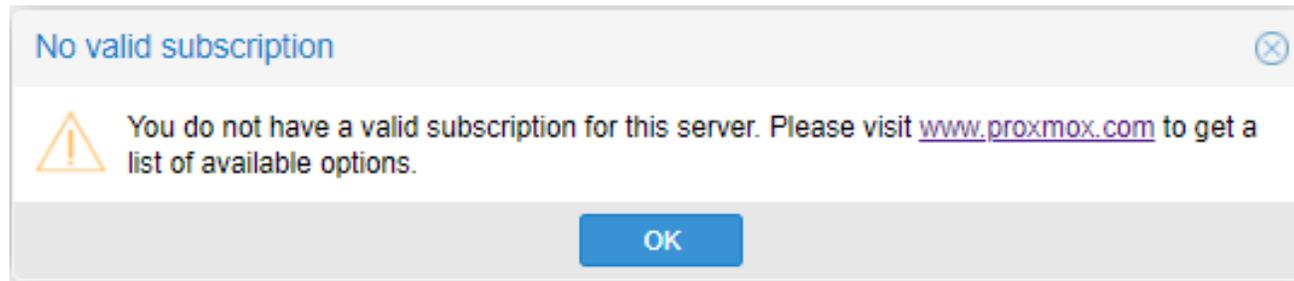
## MENGAKSES WEB GUI DARI NODE PVE1 (3)

- *Web interface* dari konfigurasi **Proxmox** berhasil diakses.
- Pada kotak dialog otentikasi **Proxmox VE Login**, lengkapi isian “**User name**” dan “**Password**”. Pada isian “**User name**”, masukkan “**root**”. Sedangkan pada isian “**Password**”, masukkan sandi login dari user “**root**” yaitu **12345678**. Klik tombol **Login**.



## MENGAKSES WEB GUI DARI NODE PVE1 (4)

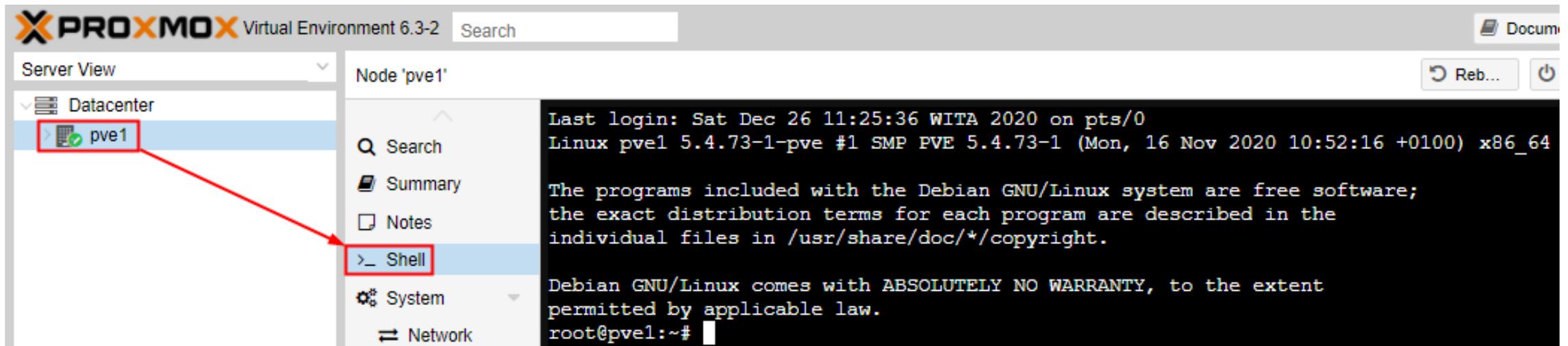
- Tampil kotak dialog “**No valid subscription**” yang menginformasikan bahwa Anda tidak memiliki *subscription* yang valid untuk *server* ini.



- Pilihan jenis *subscription* dapat diakses lebih lanjut pada situs **Proxmox** di alamat [www.proxmox.com](http://www.proxmox.com).
- Klik tombol **OK**.

# MENONAKTIFKAN PVE ENTERPRISE SUBSCRIPTION (1)

- Pada panel sebelah kiri dari **Web GUI**, pilih *node* "**PVE1**" yang terdapat di bawah menu **Datacenter**. Selanjutnya pilih menu **Shell** pada panel detail sebelah kanan.



The screenshot displays the Proxmox Web GUI interface. On the left sidebar, under the 'Datacenter' menu, the node 'pve1' is selected and highlighted with a red box. A red arrow points from this box to the 'Shell' option in the 'Node 'pve1'' menu, which is also highlighted with a red box. The main content area shows a terminal window with the following text:

```
Last login: Sat Dec 26 11:25:36 WITA 2020 on pts/0
Linux pve1 5.4.73-1-pve #1 SMP PVE 5.4.73-1 (Mon, 16 Nov 2020 10:52:16 +0100) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@pve1:~#
```

## MENONAKTIFKAN PVE ENTERPRISE SUBSCRIPTION (2)

- Menonaktifkan **PVE Enterprise subscription** dengan cara menambahkan tanda **#** di awal baris pertama dari konten pada file **/etc/apt/sources.list.d/pve-enterprise.list** menggunakan editor **nano**.

```
# nano /etc/apt/sources.list.d/pve-enterprise.list
```



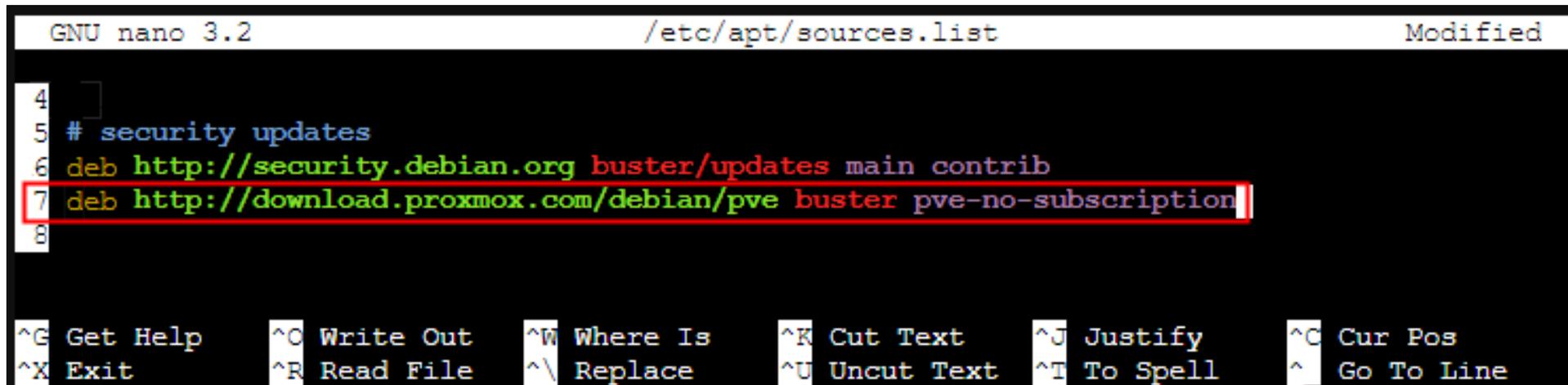
```
GNU nano 3.2 /etc/apt/sources.list.d/pve-enterprise.list
1 #deb https://enterprise.proxmox.com/debian/pve buster pve-enterprise
2
[ Wrote 1 line ]
^G Get Help      ^O Write Out     ^W Where Is     ^K Cut Text     ^J Justify      ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line
```

Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

# MENGAKTIFKAN PVE NO-SUBSCRIPTION

- Mengaktifkan **PVE No-subscription repository** dengan cara menambahkan parameter "**deb http://download.proxmox.com/debian/pve buster pve-no-subscription**" di baris terakhir pada file **/etc/apt/sources.list**.

```
# nano /etc/apt/sources.list
```



```
GNU nano 3.2 /etc/apt/sources.list Modified
4
5 # security updates
6 deb http://security.debian.org buster/updates main contrib
7 deb http://download.proxmox.com/debian/pve buster pve-no-subscription
8

^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit      ^R Read File  ^\ Replace   ^U Uncut Text ^T To Spell   ^ Go To Line
```

Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

# INSTALASI OPENVSWITCH DAN IFUPDOWN2 PADA NODE PVE1

- **Open vSwitch** tidak terinstall secara *default* pada PVE 6.3.
- Instalasi dapat dilakukan melalui **Shell** pada **Web GUI**.
- Memperbaharui **package index** dengan mengeksekusi perintah:  

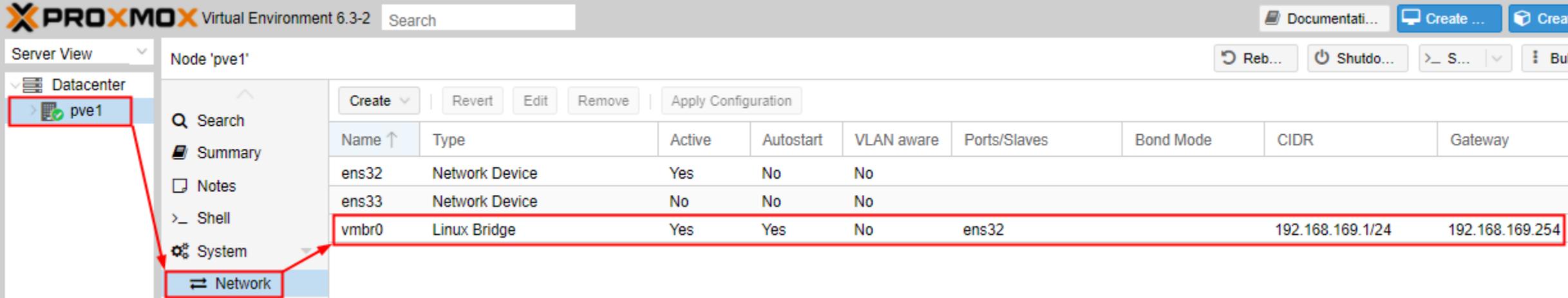
```
# apt update
```
- Menginstalasi paket aplikasi **Open vSwitch** dengan mengeksekusi perintah:  

```
# apt -y install openvswitch-switch
```
- Menginstalasi paket aplikasi **ifupdown2** agar dapat melakukan menerapkan perubahan konfigurasi jaringan melalui antarmuka berbasis web (**Web-GUI**) dari PVE **tanpa perlu reboot**.  

```
# apt -y install ifupdown2
```

# VERIFIKASI KONFIGURASI NETWORK PADA NODE PVE1

- Pada panel sebelah kiri dari **Web GUI**, pilih *node* "**PVE1**" yang terdapat di bawah menu **Datacenter**. Selanjutnya pilih menu **Network** yang terdapat dibawah menu **System** pada panel detail sebelah kanan.



Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	Network Device	Yes	No	No				
ens33	Network Device	No	No	No				
vibr0	Linux Bridge	Yes	Yes	No	ens32		192.168.169.1/24	192.168.169.254

- Terlihat terdapat satu *interface* bernama **vibr0** dengan jenis **Linux Bridge** yang memiliki *port* anggota yaitu **ens32** dan alamat **IP 192.168.169.1/24** serta **gateway 192.168.169.254**.

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE1 (1)

- Menghapus *interface Linux Bridge* `vibr0` dengan cara memilih baris yang bernilai `vibr0` dan tekan tombol **Remove**.

Node 'pve1'

Reb... Shutdo... S... Bu

Create | Revert Edit **Remove** | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	Network Device	Yes	No	No				
ens33	Network Device	No	No	No				
vibr0	Linux Bridge	Yes	Yes	No	ens32		192.168.169.1/24	192.168.169.254

Search Summary Notes Shell System Network

- *Interface* `vibr0` telah berhasil dihapus.

Node 'pve1'

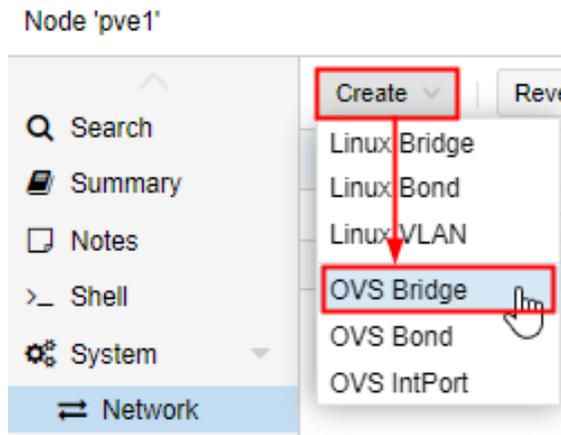
Create | Revert Edit Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves
ens32	Network Device	Yes	No	No	
ens33	Network Device	No	No	No	

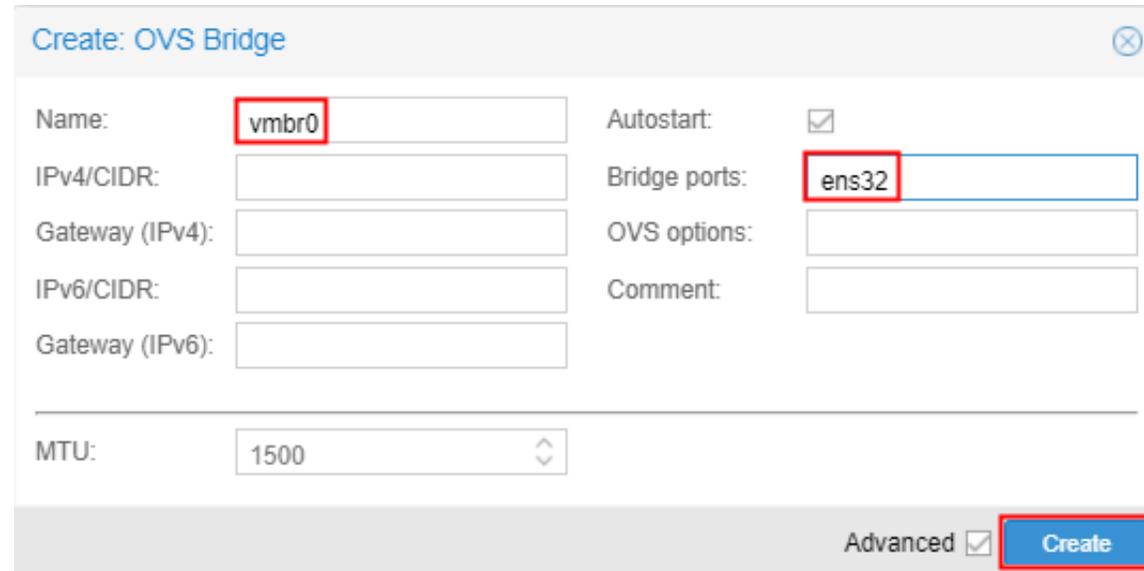
Search Summary Notes Shell System Network

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE1 (2)

- Membuat **OVS Bridge** dengan nama **vmbr0** dengan *interface* fisik yang menjadi anggotanya adalah **ens32** dengan memilih tombol **Create > OVS Bridge**.



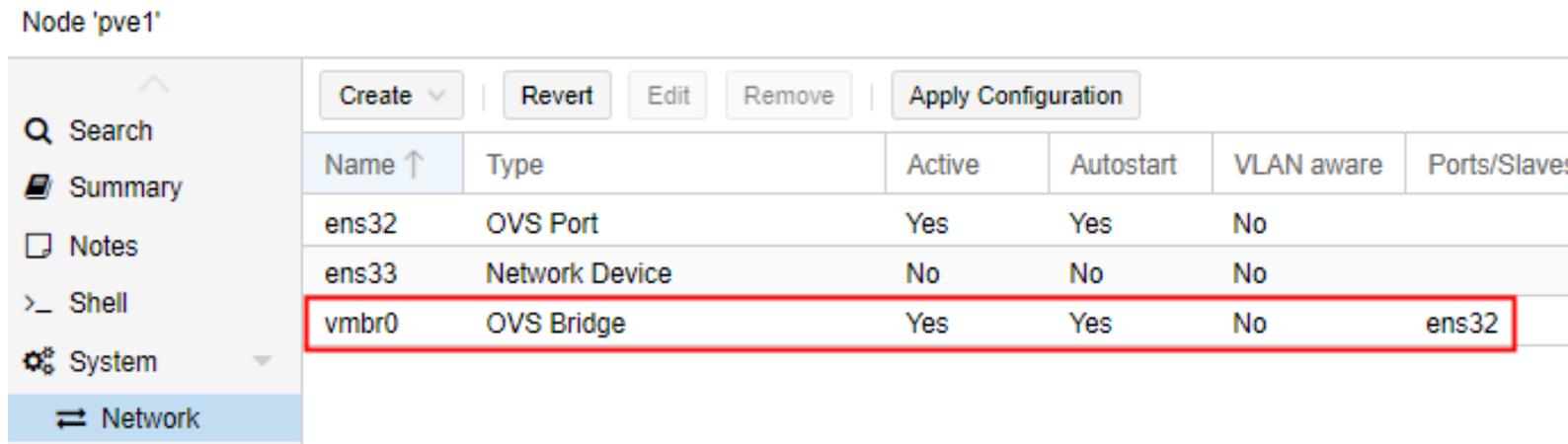
Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr0** dan **Bridge ports:** dengan **ens32**. Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The 'Name' field contains 'vmbr0' (highlighted with a red box). The 'Autostart' checkbox is checked. The 'Bridge ports' field contains 'ens32' (highlighted with a red box). The 'MTU' field is set to '1500'. At the bottom right, there is an 'Advanced' checkbox and a 'Create' button (highlighted with a red box).

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE1 (2)

- Hasil dari pembuatan **OVS Bridge** bernama **vmbr0**.

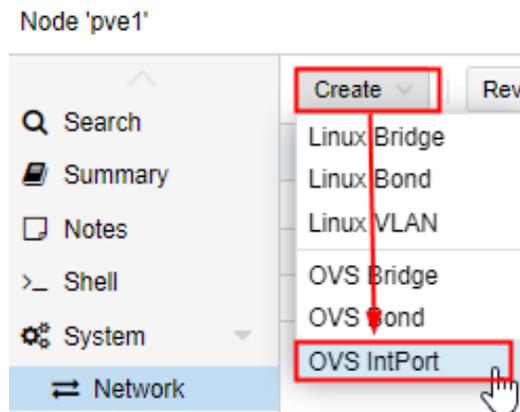
Node 'pve1'



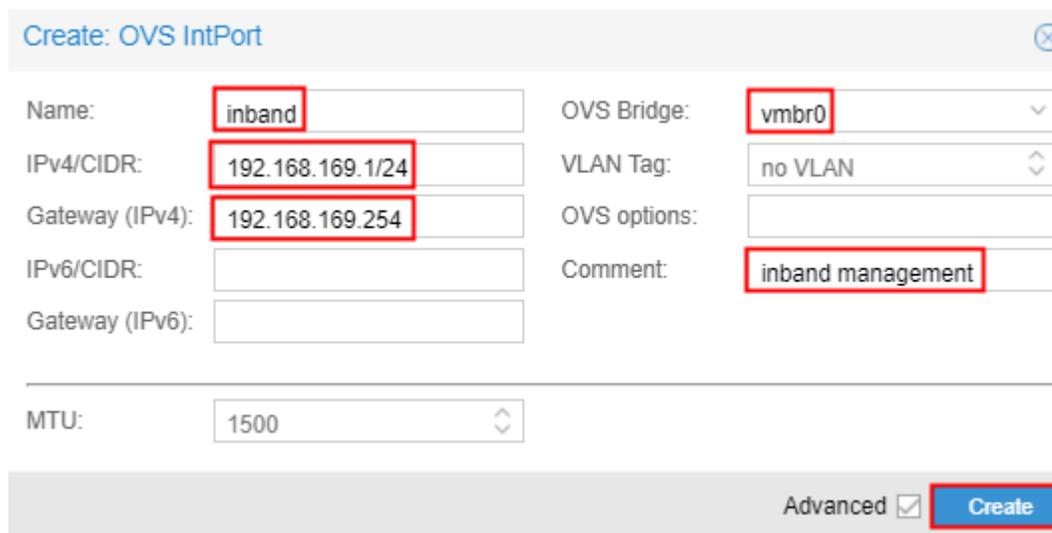
Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves
ens32	OVS Port	Yes	Yes	No	
ens33	Network Device	No	No	No	
vmbr0	OVS Bridge	Yes	Yes	No	ens32

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE1 (2)

- Membuat **OVS Internal Port (IntPort)** untuk manajemen *node PVE1* dengan nama "inband". Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter-  
**Name:** dengan **inband**, **OVS Bridge:** **vmbr0**, **IPv4/CIDR:** **192.168.169.1/24**,  
dan **Gateway (IPv4):** **192.168.169.254** serta **Comment:** **inband management**.  
Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled with the following values: Name: inband, OVS Bridge: vmbr0, IPv4/CIDR: 192.168.169.1/24, VLAN Tag: no VLAN, Gateway (IPv4): 192.168.169.254, OVS options: (empty), IPv6/CIDR: (empty), Comment: inband management, and MTU: 1500. The 'Create' button is highlighted with a red box. The 'Advanced' checkbox is checked.

Name:	inband	OVS Bridge:	vmbr0
IPv4/CIDR:	192.168.169.1/24	VLAN Tag:	no VLAN
Gateway (IPv4):	192.168.169.254	OVS options:	
IPv6/CIDR:		Comment:	inband management
Gateway (IPv6):			
MTU:	1500		

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE1 (2)

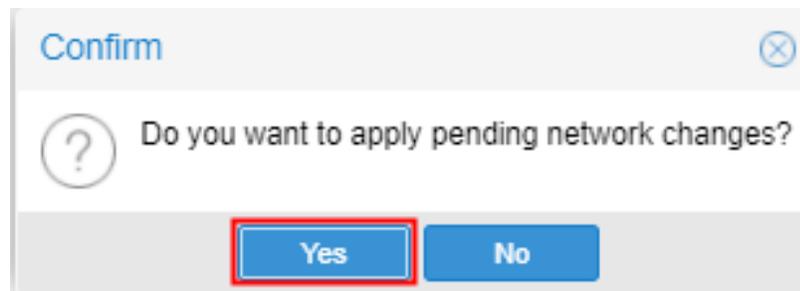
- Hasil dari pembuatan **OVS IntPort inband**.

Node 'pve1' Reb... Shutdo... >\_ S... Bulk Actio...

Create | Revert | Edit | Remove | **Apply Configuration**

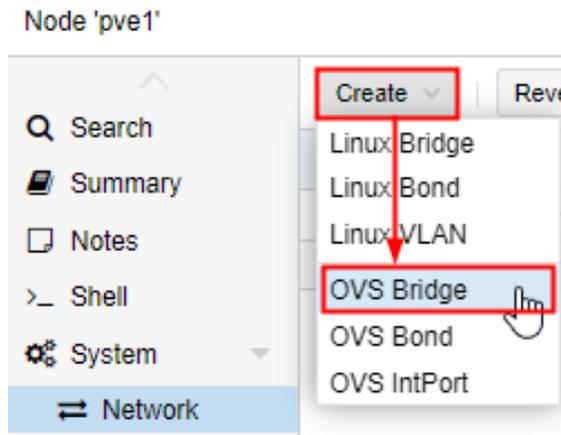
Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	Network Device	No	No	No					
<b>inband</b>	<b>OVS IntPort</b>	No	Yes	No			192.168.169.1/24	192.168.169.254	<b>inband management</b>
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				

- Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.



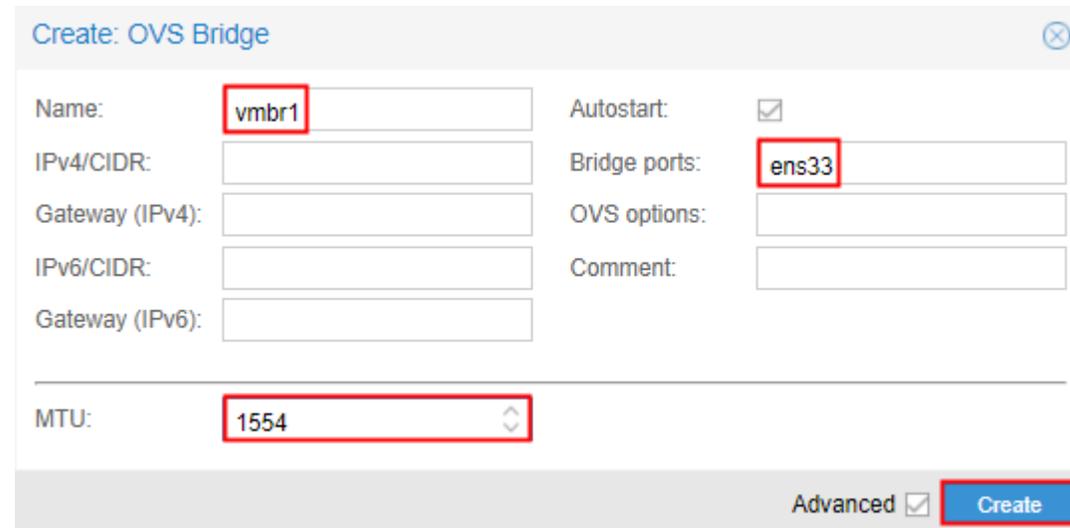
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE1 KE PVE2 (1)

- Membuat **OVS Bridge** dengan nama "**vmbr1**" dengan *interface* fisik yang menjadi anggotanya adalah **ens33**. *Interface ens33* merupakan *interface* fisik yang menghubungkan *node PVE1* ke **PVE2**.



Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr1** dan **Bridge ports:** dengan **ens33** serta **MTU:** 1554.

Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The fields are filled as follows: Name: vmbr1, Autostart: checked, Bridge ports: ens33, MTU: 1554. The 'Create' button at the bottom right is highlighted with a red box. The 'Advanced' checkbox is also checked.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE1 KE PVE2 (2)

- Hasil dari pembuatan **OVS Bridge vubr1**.

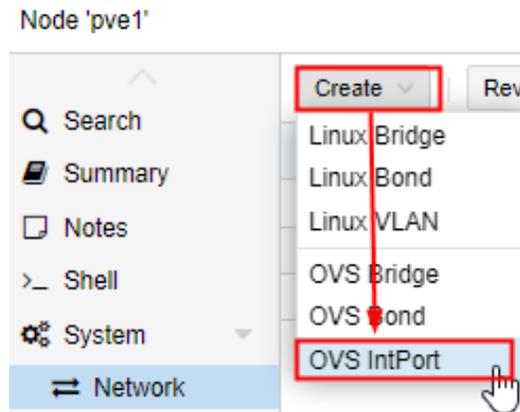
Node 'pve1' Reb... Shutdo... S... Bulk Actio...

Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	No	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.1/24	192.168.169.254	inband management
vubr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vubr1	OVS Bridge	No	Yes	No	ens33				

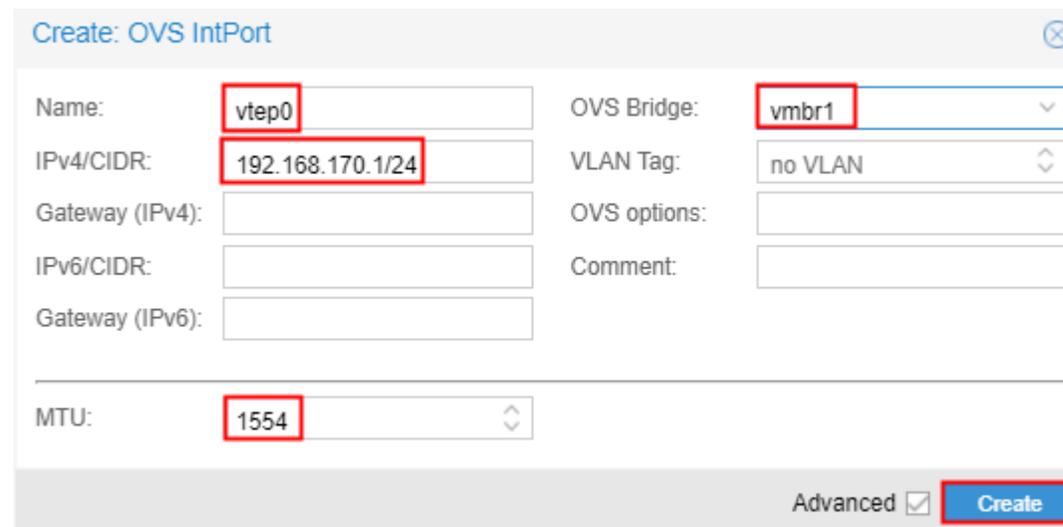
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE1 KE PVE2 (3)

- Membuat **OVS Internal Port (IntPort)** dengan nama **vtep0**. *Interface vtep0* merupakan **VXLAN Tunnel End Point (VTEP)** sebagai titik enkapsulasi dari VXLAN dan terhubung ke sumber dari trafik yaitu *hypervisor PVE1*. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter-**Name:** dengan **vtep0**, **OVS Bridge:** **vmbr1** dan **IPv4/CIDR:** **192.168.170.1/24** serta **MTU:** **1554**.

Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled as follows: Name: vtep0, OVS Bridge: vmbr1, IPv4/CIDR: 192.168.170.1/24, VLAN Tag: no VLAN, MTU: 1554. The 'Create' button at the bottom right is highlighted with a red box. The 'Advanced' checkbox is checked.

Name:	<input type="text" value="vtep0"/>	OVS Bridge:	<input type="text" value="vmbr1"/>
IPv4/CIDR:	<input type="text" value="192.168.170.1/24"/>	VLAN Tag:	<input type="text" value="no VLAN"/>
Gateway (IPv4):	<input type="text"/>	OVS options:	<input type="text"/>
IPv6/CIDR:	<input type="text"/>	Comment:	<input type="text"/>
Gateway (IPv6):	<input type="text"/>		
MTU:	<input type="text" value="1554"/>		

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE1 KE PVE2 (4)

- Hasil dari pembuatan **OVS IntPort vtep0**.

Node 'pve1' Reb... Shutdo... > S... Bulk Actio...

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	No	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.1/24	192.168.169.254	inband management
vibr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vibr1	OVS Bridge	No	Yes	No	ens33 vtep0				
vtep0	OVS IntPort	No	Yes	No			192.168.170.1/24		

# MENGUBAH MTU PADA INTERFACE FISIK ENS33

- Mengubah **Maximum Transfer Unit (MTU)** dari *interface* **ens33** dengan cara memilih baris yang bernilai **ens33** dan tekan tombol **Edit**.

Node 'pve1'

Name ↑	Type	Active	Autostart	VLAN a
ens32	OVS Port	Yes	Yes	No
ens33	OVS Port	No	Yes	No
inband	OVS IntPort	Yes	Yes	No
vmbr0	OVS Bridge	Yes	Yes	No
vmbr1	OVS Bridge	No	Yes	No
vtep0	OVS IntPort	No	Yes	No

Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.

Tampil kotak dialog **Edit: OVS Port** dan lakukan penyesuaian pada nilai dari parameter **MTU**: menjadi **1554**. Klik tombol **OK** untuk menyimpan perubahan.

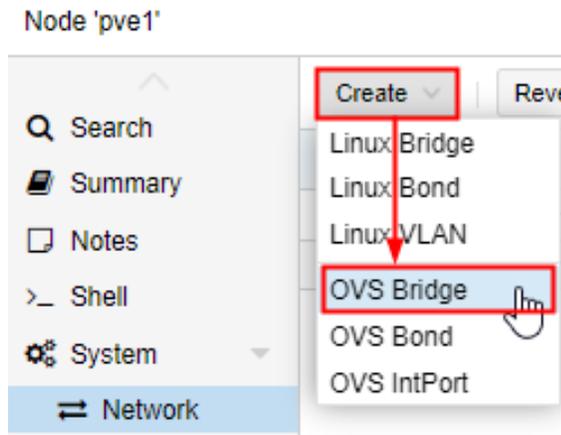
Edit: OVS Port

Name:	ens33	OVS Bridge:	vmbr1
IPv4/CIDR:	<input type="text"/>	VLAN Tag:	no VLAN
Gateway (IPv4):	<input type="text"/>	OVS options:	<input type="text"/>
IPv6/CIDR:	<input type="text"/>	Comment:	<input type="text"/>
Gateway (IPv6):	<input type="text"/>		
MTU:	<input type="text" value="1554"/>		

Advanced  **OK** **Reset**

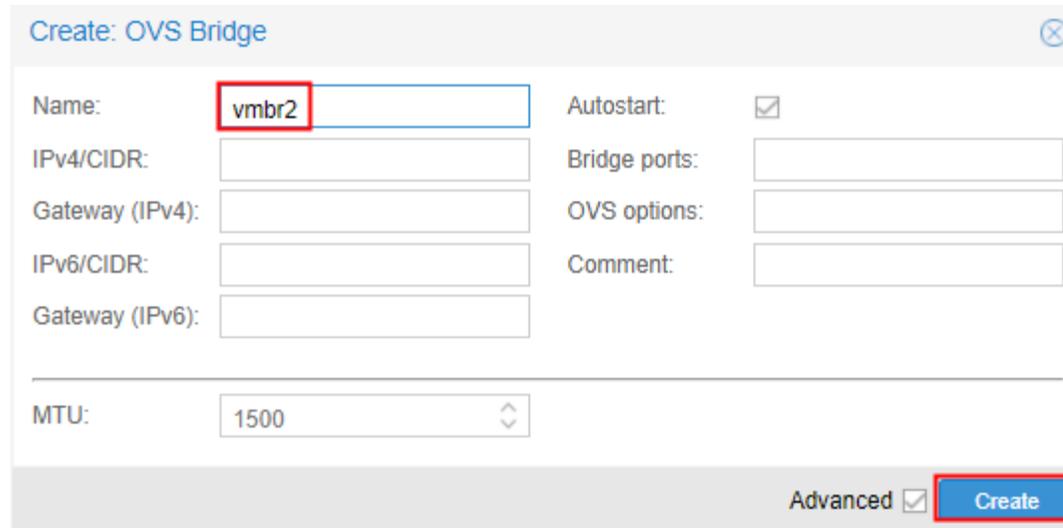
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (1)

- Membuat **OVS Bridge** dengan nama "**vmbr2**" yang digunakan sebagai **switch** untuk koneksi dari **Container (CT)** yang dibuat pada **node PVE1** ke jaringan.



Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr2**.

Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The 'Name' field is filled with 'vmbr2' and is highlighted with a red box. The 'Autostart' checkbox is checked. The 'MTU' field is set to 1500. At the bottom right, the 'Advanced' checkbox is checked and the 'Create' button is highlighted with a red box.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (2)

- Hasil dari pembuatan **OVS Bridge vmbr2**.

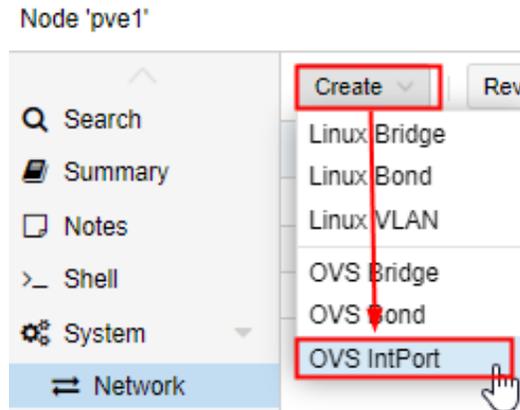
Node 'pve1'

Create | Revert | Edit | Remove | Apply Configuration

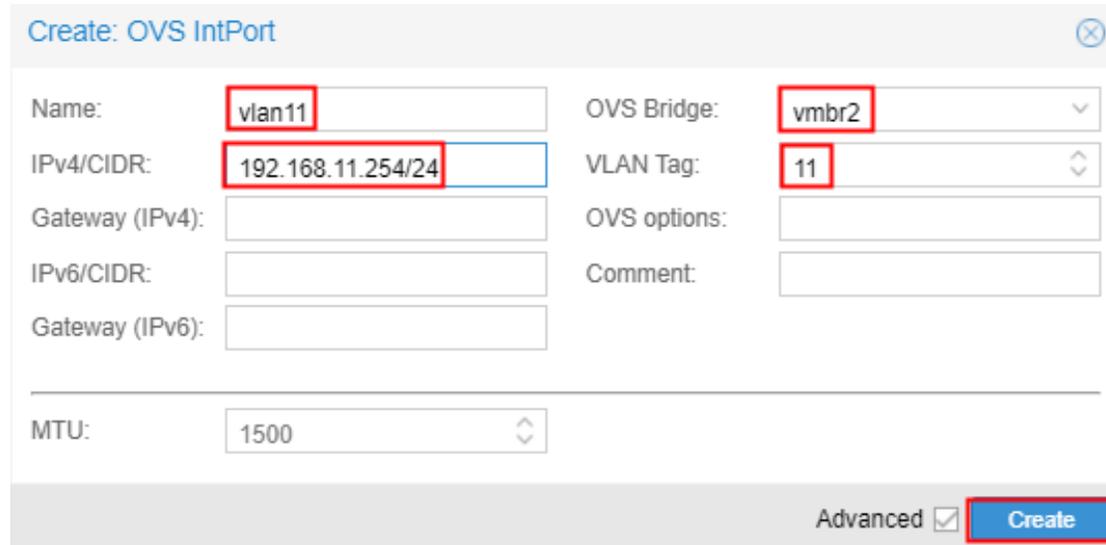
Name	Type ↑	Active	Autostart	VLAN aware
vmbr2	OVS Bridge	No	Yes	No
vmbr1	OVS Bridge	Yes	Yes	No
vmbr0	OVS Bridge	Yes	Yes	No
inband	OVS IntPort	Yes	Yes	No
vtep0	OVS IntPort	Yes	Yes	No
ens32	OVS Port	Yes	Yes	No
ens33	OVS Port	Yes	Yes	No

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (3)

- Membuat **OVS Internal Port (IntPort)** untuk setiap VLAN ID yaitu dengan nama **vlan11** dan **vlan12**. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter **Name:** dengan **vlan11**, **OVS Bridge:** **vmbr2** dan **IPv4/CIDR:** **192.168.11.254/24** serta **VLAN Tag:** **11** untuk VLAN ID 11. Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled as follows: Name: vlan11, OVS Bridge: vmbr2, IPv4/CIDR: 192.168.11.254/24, VLAN Tag: 11, MTU: 1500. The 'Create' button at the bottom right is highlighted with a red box. The 'Advanced' checkbox is checked.

Name:	<input type="text" value="vlan11"/>	OVS Bridge:	<input type="text" value="vmbr2"/>
IPv4/CIDR:	<input type="text" value="192.168.11.254/24"/>	VLAN Tag:	<input type="text" value="11"/>
Gateway (IPv4):	<input type="text"/>	OVS options:	<input type="text"/>
IPv6/CIDR:	<input type="text"/>	Comment:	<input type="text"/>
Gateway (IPv6):	<input type="text"/>		
MTU:	<input type="text" value="1500"/>		

Advanced

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (4)

- Hasil dari pembuatan **OVS IntPort** **vlan11**.

Node 'pve1' Reb... Shutdo... > S... Bulk Actio...

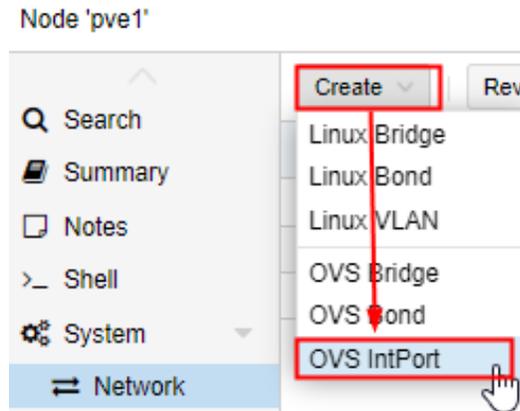
Create Revert Edit Remove Apply Configuration

Name	Type ↑	Active	Autostart	VLAN aware	Ports/Slave...1	Bond Mode	CIDR	Gateway	Comment
vmbr2	OVS Bridge	No	Yes	No	vlan11				
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0				
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.1/24		
vlan11	OVS IntPort	No	Yes	No			192.168.11.254/24		
inband	OVS IntPort	Yes	Yes	No			192.168.169.1/24	192.168.169.254	inband management
ens33	OVS Port	Yes	Yes	No					
ens32	OVS Port	Yes	Yes	No					

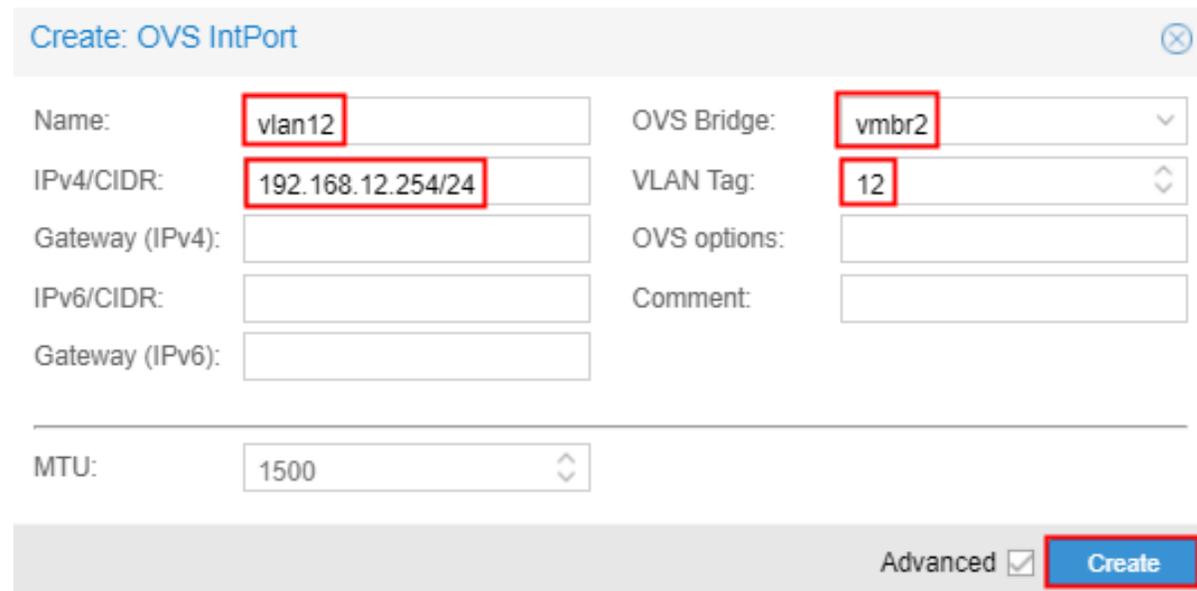
- Dengan cara yang sama lakukan pembuatan **OVS IntPort** untuk **VLAN ID 12**.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (5)

- Membuat **OVS Internal Port (IntPort)** untuk **vlan12**. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter **Name:** dengan **vlan12**, **OVS Bridge:** **vibr2** dan **IPv4/CIDR:** **192.168.12.254/24** serta **VLAN Tag:** **12** untuk VLAN ID 12. Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled as follows: Name: vlan12, OVS Bridge: vibr2, IPv4/CIDR: 192.168.12.254/24, VLAN Tag: 12, MTU: 1500. The 'Advanced' checkbox is checked, and the 'Create' button is highlighted with a red box.

Name:	<input type="text" value="vlan12"/>	OVS Bridge:	<input type="text" value="vibr2"/>
IPv4/CIDR:	<input type="text" value="192.168.12.254/24"/>	VLAN Tag:	<input type="text" value="12"/>
Gateway (IPv4):	<input type="text"/>	OVS options:	<input type="text"/>
IPv6/CIDR:	<input type="text"/>	Comment:	<input type="text"/>
Gateway (IPv6):	<input type="text"/>		
MTU:	<input type="text" value="1500"/>		

Advanced

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE1 (6)

- Hasil dari pembuatan **OVS IntPort vlan12**.

Node 'pve1'

Reb... Shutdo... >\_ S... Bulk Actio... ?

Create Revert Edit Remove Apply Configuration

Name	Type ↑	Active	Autostart	VLAN aware	Ports/Slave...1	Bond Mode	CIDR	Gateway	Comment
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0				
vmbr2	OVS Bridge	No	Yes	No	vlan11 vlan12				
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vlan12	OVS IntPort	No	Yes	No			192.168.12.254/24		
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.1/24		
vlan11	OVS IntPort	No	Yes	No			192.168.11.254/24		
inband	OVS IntPort	Yes	Yes	No			192.168.169.1/24	192.168.169.254	inband management
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	Yes	Yes	No					

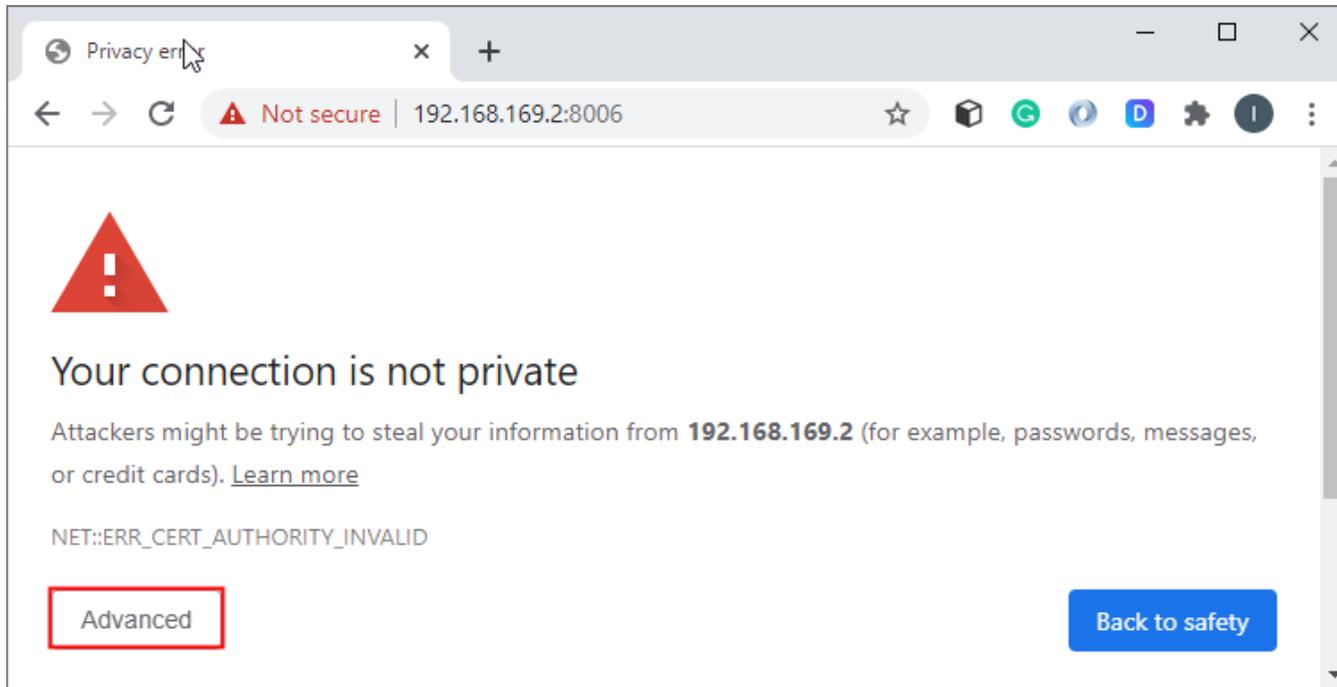
- Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.



# KONFIGURASI PADA NODE PVE2

# MENGAKSES WEB GUI DARI NODE PVE2 (1)

- Buka **browser**, sebagai contoh menggunakan **Chrome**. Pada *address bar* dari *browser*, masukkan URL **https://192.168.169.2:8006**.

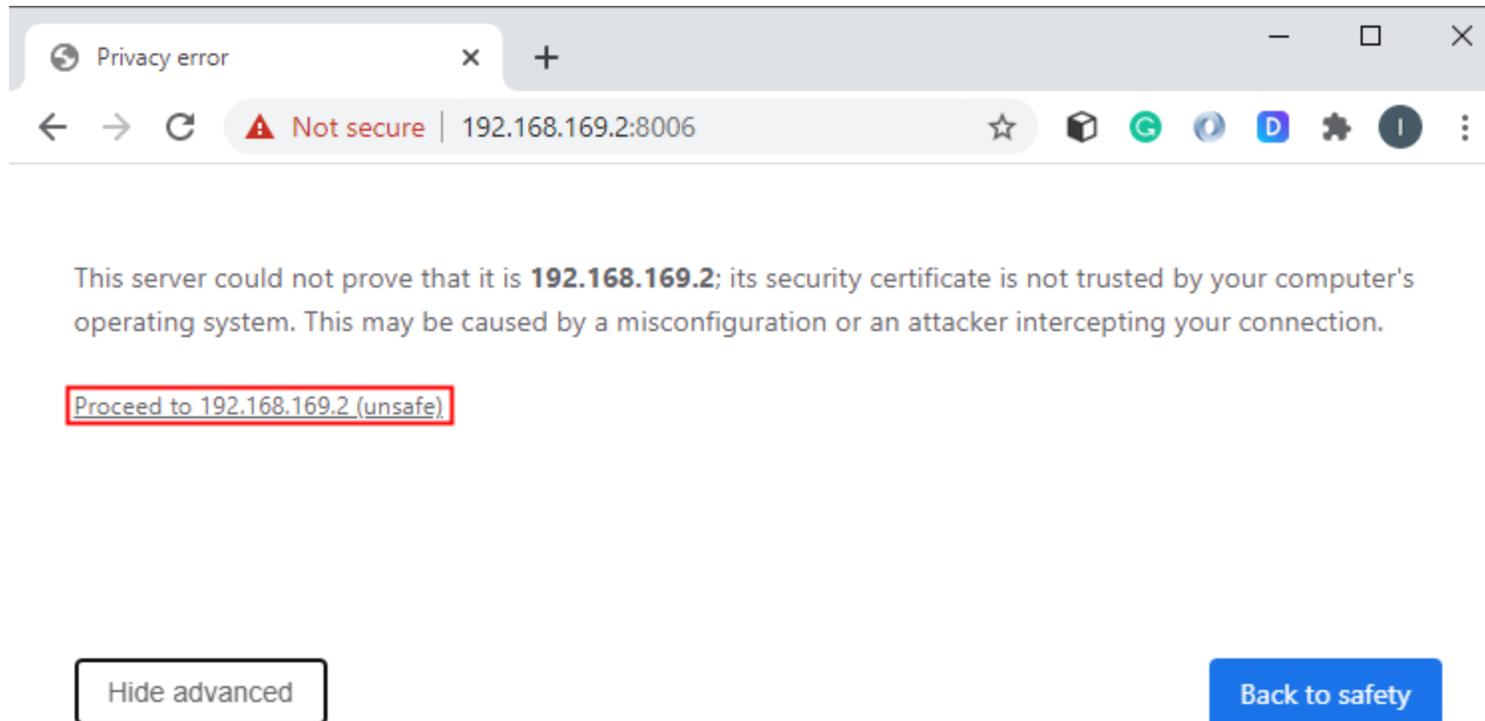


Tampil pesan peringatan “**Your connection is not private**”.

Klik **Advanced** untuk melanjutkan pengaksesan.

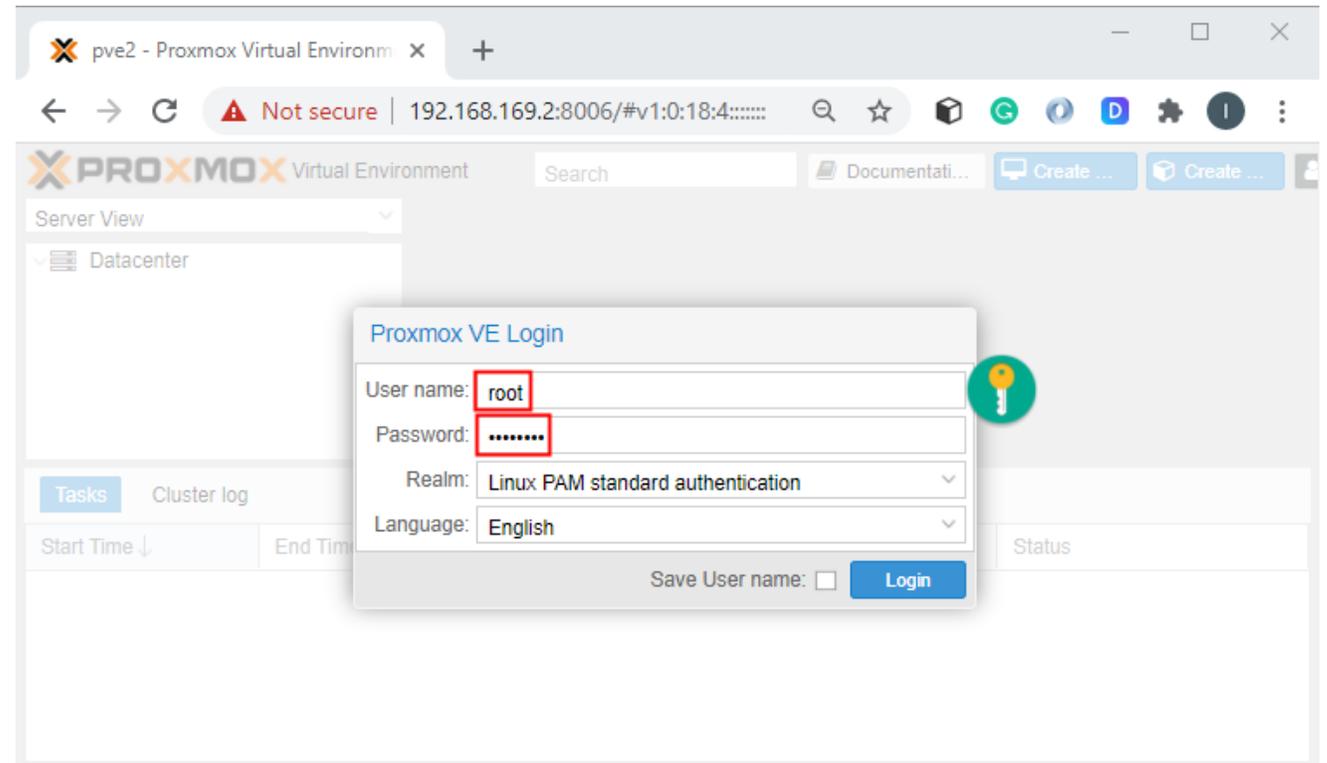
# MENGAKSES WEB GUI DARI NODE PVE2 (2)

- Klik link “**Proceed to 192.168.169.2 (unsafe)**”.



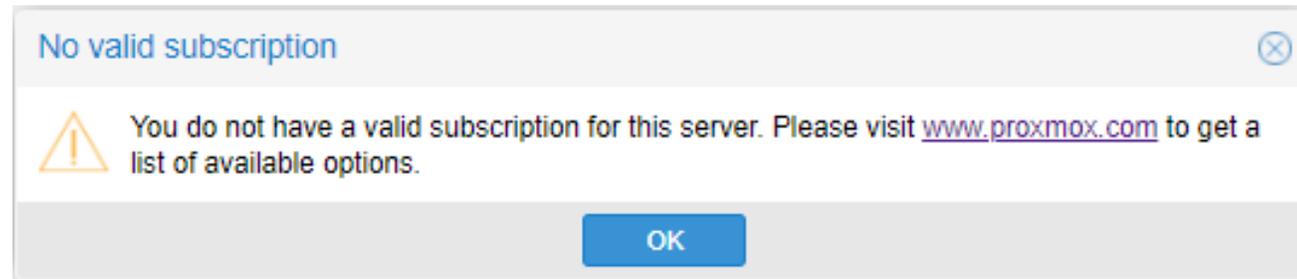
## MENGAKSES WEB GUI DARI NODE PVE2 (3)

- *Web interface* dari konfigurasi **Proxmox** berhasil diakses.
- Pada kotak dialog otentikasi **Proxmox VE Login**, lengkapi isian “**User name**” dan “**Password**”. Pada isian “**User name**”, masukkan “**root**”. Sedangkan pada isian “**Password**”, masukkan sandi login dari user “**root**” yaitu **12345678**. Klik tombol **Login**.



## MENGAKSES WEB GUI DARI NODE PVE2 (4)

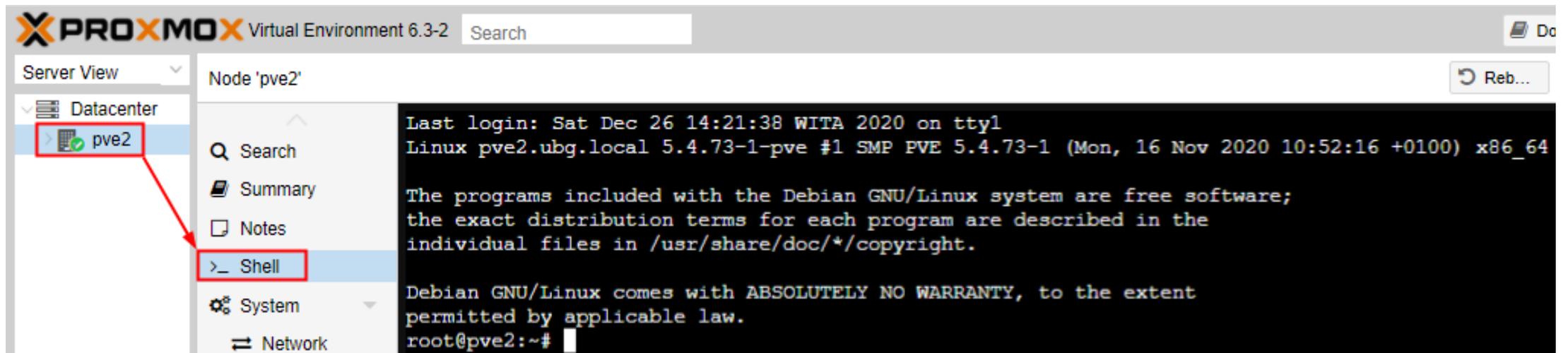
- Tampil kotak dialog “**No valid subscription**” yang menginformasikan bahwa Anda tidak memiliki *subscription* yang valid untuk *server* ini.



- Pilihan jenis *subscription* dapat diakses lebih lanjut pada situs **Proxmox** di alamat [www.proxmox.com](http://www.proxmox.com).
- Klik tombol **OK**.

# MENONAKTIFKAN PVE ENTERPRISE SUBSCRIPTION (1)

- Pada panel sebelah kiri dari **Web GUI**, pilih *node* "**PVE2**" yang terdapat di bawah menu **Datacenter**. Selanjutnya pilih menu **Shell** pada panel detail sebelah kanan.



## MENONAKTIFKAN PVE ENTERPRISE SUBSCRIPTION (2)

- Menonaktifkan **PVE Enterprise subscription** dengan cara menambahkan tanda **#** di awal baris pertama dari konten pada file **/etc/apt/sources.list.d/pve-enterprise.list** menggunakan editor **nano**.

```
# nano /etc/apt/sources.list.d/pve-enterprise.list
```



```
GNU nano 3.2 /etc/apt/sources.list.d/pve-enterprise.list
1 #deb https://enterprise.proxmox.com/debian/pve buster pve-enterprise
2
[ Wrote 1 line ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit          ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

# MENGAKTIFKAN PVE NO-SUBSCRIPTION

- Mengaktifkan **PVE No-subscription repository** dengan cara menambahkan parameter "**deb http://download.proxmox.com/debian/pve buster pve-no-subscription**" di baris terakhir pada file **/etc/apt/sources.list**.

```
# nano /etc/apt/sources.list
```



```
GNU nano 3.2 /etc/apt/sources.list Modified
4
5 # security updates
6 deb http://security.debian.org buster/updates main contrib
7 deb http://download.proxmox.com/debian/pve buster pve-no-subscription
8

^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos
^X Exit          ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

# INSTALASI OPENVSWITCH DAN IFUPDOWN2 PADA NODE PVE2

- **Open vSwitch** tidak terinstall secara *default* pada PVE 6.3.
- Instalasi dapat dilakukan melalui **Shell** pada **Web GUI**.
- Memperbaharui **package index** dengan mengeksekusi perintah:  

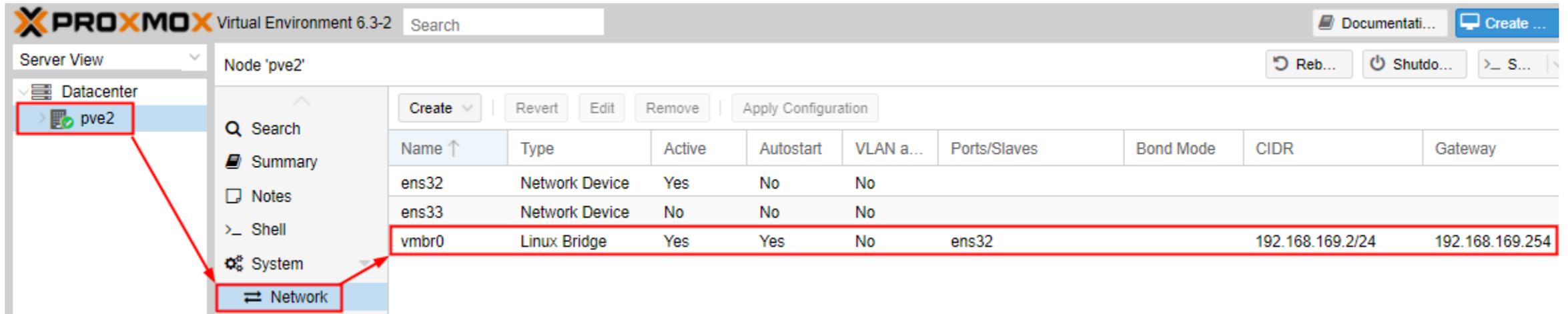
```
# apt update
```
- Menginstalasi paket aplikasi **Open vSwitch** dengan mengeksekusi perintah:  

```
# apt -y install openvswitch-switch
```
- Menginstalasi paket aplikasi **ifupdown2** agar dapat melakukan menerapkan perubahan konfigurasi jaringan melalui antarmuka berbasis web (**Web-GUI**) dari PVE **tanpa perlu reboot**.  

```
# apt -y install ifupdown2
```

# VERIFIKASI KONFIGURASI NETWORK PADA NODE PVE2

- Pada panel sebelah kiri dari **Web GUI**, pilih *node* "**PVE2**" yang terdapat di bawah menu **Datacenter**. Selanjutnya pilih menu **Network** yang terdapat dibawah menu **System** pada panel detail sebelah kanan.



The screenshot shows the Proxmox VE Web GUI interface. On the left sidebar, the 'Datacenter' menu is expanded, and 'pve2' is selected. Below it, the 'System' menu is expanded, and 'Network' is selected. The main panel displays the network configuration for node 'pve2'. A table lists the network devices:

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	Network Device	Yes	No	No				
ens33	Network Device	No	No	No				
vmbr0	Linux Bridge	Yes	Yes	No	ens32		192.168.169.2/24	192.168.169.254

- Terlihat terdapat satu *interface* bernama **vmbr0** dengan jenis **Linux Bridge** yang memiliki *port* anggota yaitu **ens32** dan alamat **IP 192.168.169.2/24** serta **gateway 192.168.169.254**.

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE2 (1)

- Menghapus *interface Linux Bridge* **vmbr0** dengan cara memilih baris yang bernilai **vmbr0** dan tekan tombol **Remove**.

Node 'pve2'

Reb... Shutdo... >\_ S...

Create | Revert | Edit | **Remove** | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	Network Device	Yes	No	No				
ens33	Network Device	No	No	No				
<b>vmbr0</b>	<b>Linux Bridge</b>	<b>Yes</b>	<b>Yes</b>	<b>No</b>	<b>ens32</b>		<b>192.168.169.2/24</b>	<b>192.168.169.254</b>

Search  
Summary  
Notes  
Shell  
System  
Network

- *Interface* **vmbr0** telah berhasil dihapus.

Node 'pve1'

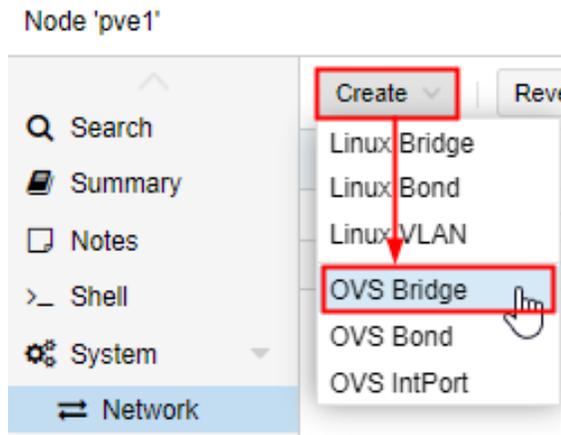
Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves
ens32	Network Device	Yes	No	No	
ens33	Network Device	No	No	No	

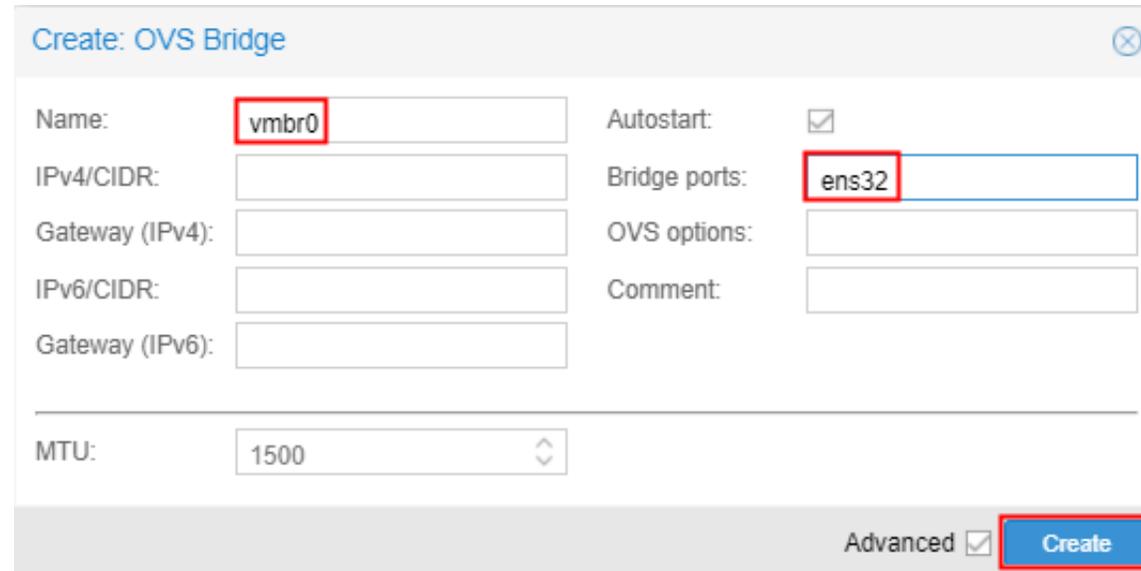
Search  
Summary  
Notes  
Shell  
System  
Network

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE2 (2)

- Membuat **OVS Bridge** dengan nama **vmbr0** dengan *interface* fisik yang menjadi anggotanya adalah **ens32** dengan memilih tombol **Create > OVS Bridge**.



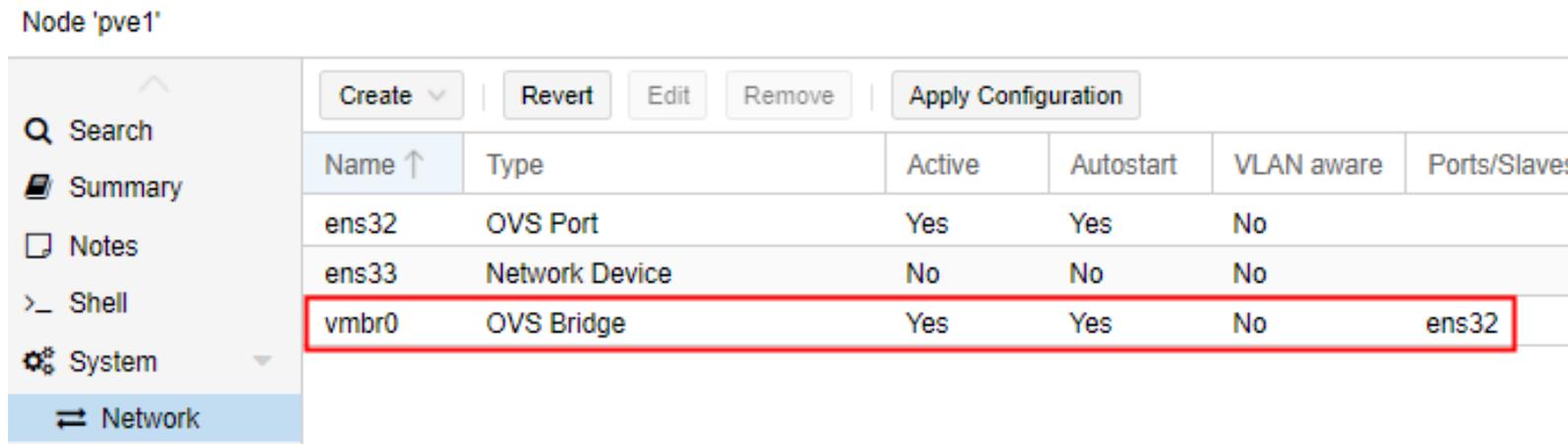
Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr0** dan **Bridge ports:** dengan **ens32**. Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The 'Name' field contains 'vmbr0' (highlighted with a red box). The 'Autostart' checkbox is checked. The 'Bridge ports' field contains 'ens32' (highlighted with a red box). The 'MTU' field is set to '1500'. At the bottom right, there is an 'Advanced' checkbox and a 'Create' button (highlighted with a red box).

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE2 (2)

- Hasil dari pembuatan **OVS Bridge** bernama **vmbr0**.

Node 'pve1'

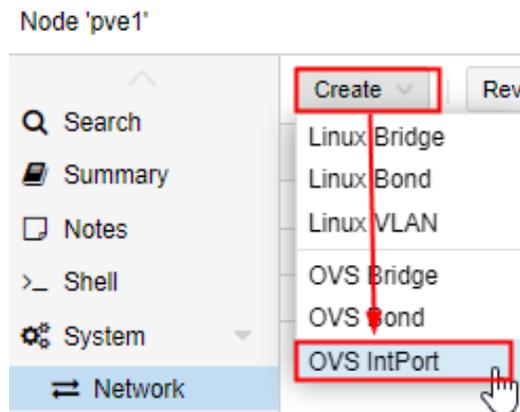


The screenshot shows the network configuration interface for Node 'pve1'. On the left is a sidebar with navigation options: Search, Summary, Notes, Shell, System, and Network (selected). The main area contains a table of network devices. Above the table are buttons for 'Create', 'Revert', 'Edit', 'Remove', and 'Apply Configuration'. The table has columns for Name, Type, Active, Autostart, VLAN aware, and Ports/Slaves. The row for 'vmbr0' is highlighted with a red border, showing it is an OVS Bridge, active, autostarted, not VLAN aware, and connected to the 'ens32' port.

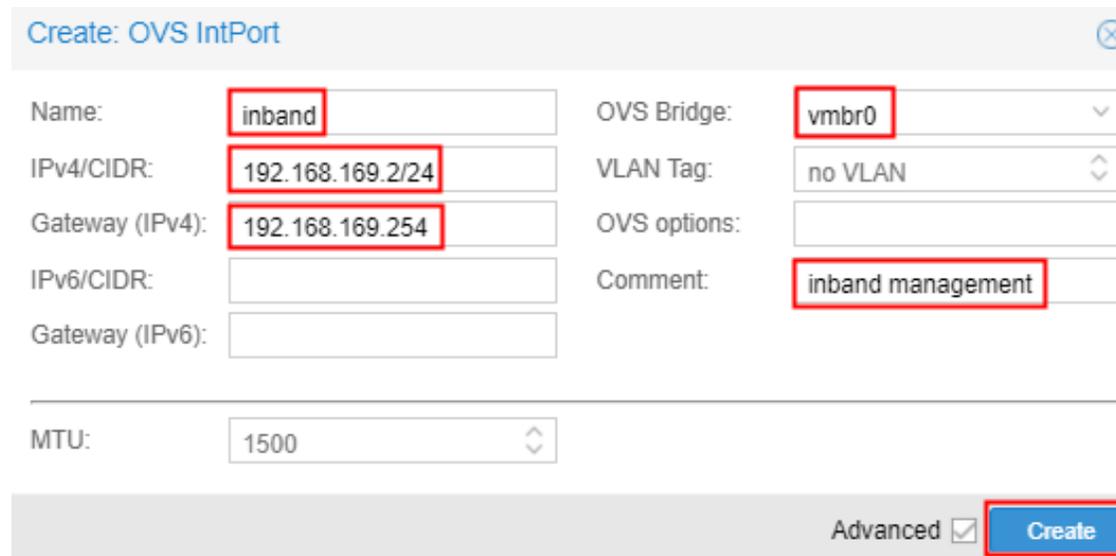
Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves
ens32	OVS Port	Yes	Yes	No	
ens33	Network Device	No	No	No	
vmbr0	OVS Bridge	Yes	Yes	No	ens32

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE2 (2)

- Membuat **OVS Internal Port (IntPort)** untuk manajemen *node PVE1* dengan nama "inband". Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter-**Name:** dengan **inband**, **OVS Bridge:** **vmbr0**, **IPv4/CIDR:** **192.168.169.2/24**, dan **Gateway (IPv4):** **192.168.169.254** serta **Comment:** **inband management**. Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled as follows: Name: inband, OVS Bridge: vmbr0, IPv4/CIDR: 192.168.169.2/24, VLAN Tag: no VLAN, Gateway (IPv4): 192.168.169.254, OVS options: (empty), IPv6/CIDR: (empty), Comment: inband management, Gateway (IPv6): (empty), and MTU: 1500. The 'Advanced' checkbox is checked, and the 'Create' button is highlighted with a red box.

# MENGUBAH KONFIGURASI NETWORK DARI LINUX BRIDGE KE OPEN VSWITCH PADA NODE PVE2 (2)

- Hasil dari pembuatan **OVS IntPort inband**.

Node 'pve2'

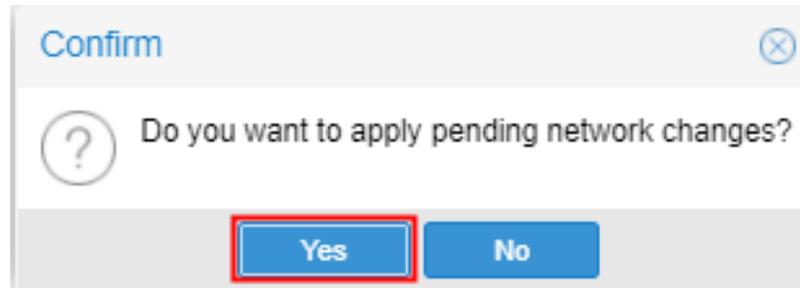
Reb... Shutdo... >\_ S... Bulk Actio... ?

Create | Revert Edit Remove | **Apply Configuration**

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	Network Device	No	No	No					
<b>inband</b>	<b>OVS IntPort</b>	No	Yes	No			192.168.169.2/24	192.168.169.254	<b>inband management</b>
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				

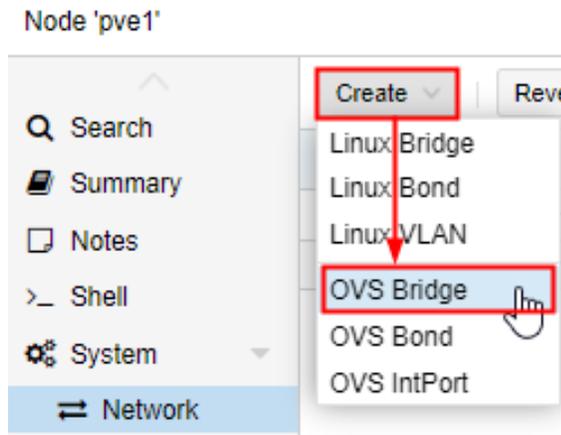
Network

- Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.



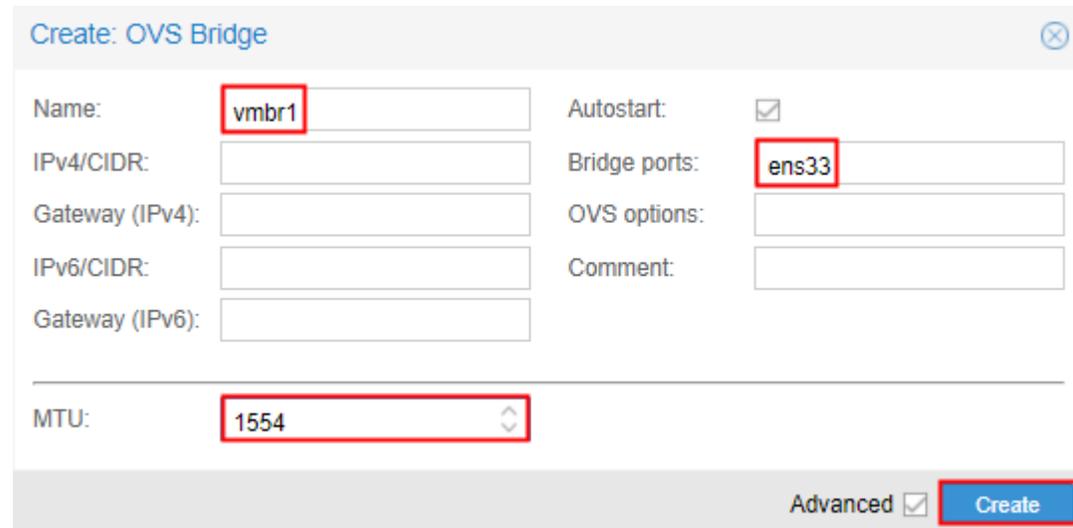
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE2 KE PVE1 (1)

- Membuat **OVS Bridge** dengan nama "**vmbr1**" dengan *interface* fisik yang menjadi anggotanya adalah **ens33**. *Interface ens33* merupakan *interface* fisik yang menghubungkan *node PVE2* ke **PVE1**.



Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr1** dan **Bridge ports:** dengan **ens33** serta **MTU:** 1554.

Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The dialog has a title bar with a close button. It contains several input fields: 'Name:' with 'vmbr1' entered, 'Autostart:' with a checked checkbox, 'IPv4/CIDR:', 'Gateway (IPv4):', 'IPv6/CIDR:', 'Gateway (IPv6):', 'Bridge ports:' with 'ens33' entered, 'OVS options:', and 'Comment:'. At the bottom, there is an 'MTU:' field with a dropdown menu set to '1554'. A red box highlights the 'Create' button at the bottom right. The 'Advanced' checkbox is also checked.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE2 KE PVE1 (2)

- Hasil dari pembuatan **OVS Bridge vubr1**.

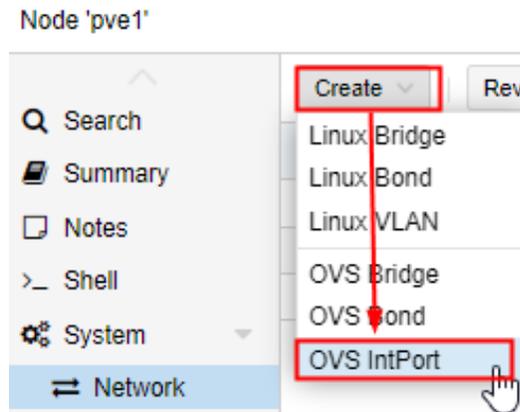
Node 'pve2' Reb... Shutdo... > S... Bulk Actio... ?

Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	No	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.2/24	192.168.169.254	inband management
vubr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vubr1	OVS Bridge	No	Yes	No	ens33				

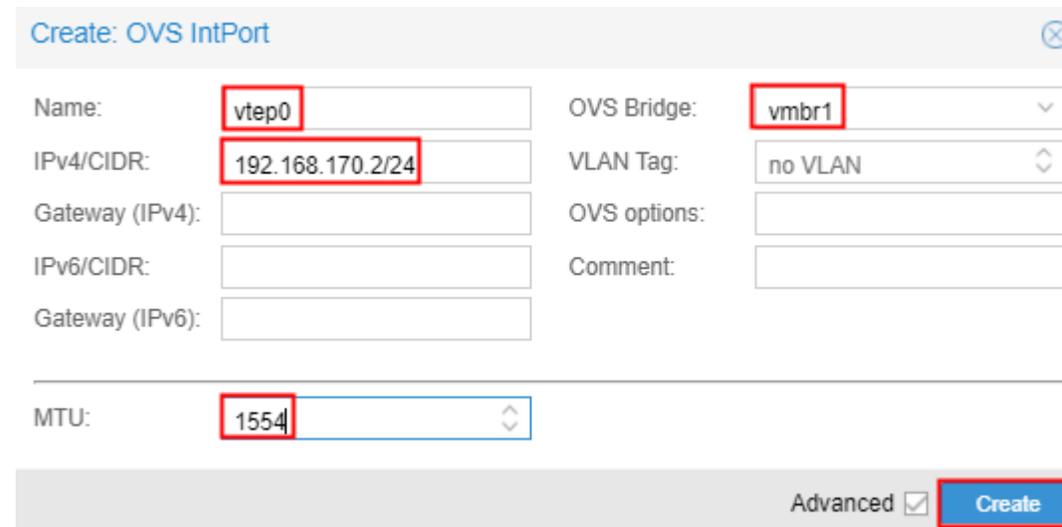
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE2 KE PVE1 (3)

- Membuat **OVS Internal Port (IntPort)** dengan nama **vtep0**. *Interface vtep0* merupakan **VXLAN Tunnel End Point (VTEP)** sebagai titik enkapsulasi dari VXLAN dan terhubung ke sumber dari trafik yaitu *hypervisor PVE1*. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter-**Name:** dengan **vtep0**, **OVS Bridge:** **vmbr1** dan **IPv4/CIDR:** **192.168.170.2/24** serta **MTU:** **1554**.

Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The dialog has a title bar with a close button. It contains several input fields: 'Name' with 'vtep0', 'OVS Bridge' with 'vmbr1', 'IPv4/CIDR' with '192.168.170.2/24', 'VLAN Tag' with 'no VLAN', 'MTU' with '1554', and 'Advanced' checked. There are also empty fields for 'Gateway (IPv4)', 'Gateway (IPv6)', and 'Comment'. A 'Create' button is at the bottom right.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK UNDERLAY NETWORK DARI NODE PVE2 KE PVE1 (4)

- Hasil dari pembuatan **OVS IntPort vtep0**.

Node 'pve2'

Reb... Shutdo... S... Bulk Actio... ?

Create Revert Edit Remove Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	No	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.2/24	192.168.169.254	inband management
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vmbr1	OVS Bridge	No	Yes	No	ens33 vtep0				
vtep0	OVS IntPort	No	Yes	No			192.168.170.2/24		

# MENGUBAH MTU PADA INTERFACE FISIK ENS33

- Mengubah **Maximum Transfer Unit (MTU)** dari *interface* **ens33** dengan cara memilih baris yang bernilai **ens33** dan tekan tombol **Edit**.

Node 'pve1'

Create | Revert | **Edit** | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a
ens32	OVS Port	Yes	Yes	No
<b>ens33</b>	OVS Port	No	Yes	No
inband	OVS IntPort	Yes	Yes	No
vmbr0	OVS Bridge	Yes	Yes	No
vmbr1	OVS Bridge	No	Yes	No
vtep0	OVS IntPort	No	Yes	No

Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.

Tampil kotak dialog **Edit: OVS Port** dan lakukan penyesuaian pada nilai dari parameter **MTU**: menjadi **1554**. Klik tombol **OK** untuk menyimpan perubahan.

Edit: OVS Port

Name: ens33 OVS Bridge: vmbr1

IPv4/CIDR:  VLAN Tag: no VLAN

Gateway (IPv4):  OVS options:

IPv6/CIDR:  Comment:

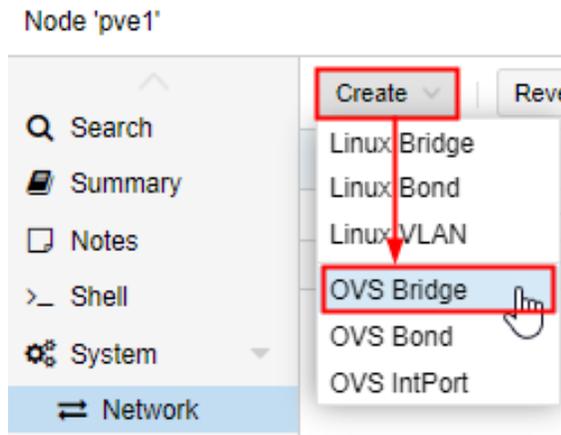
Gateway (IPv6):

MTU:

Advanced  **OK** Reset

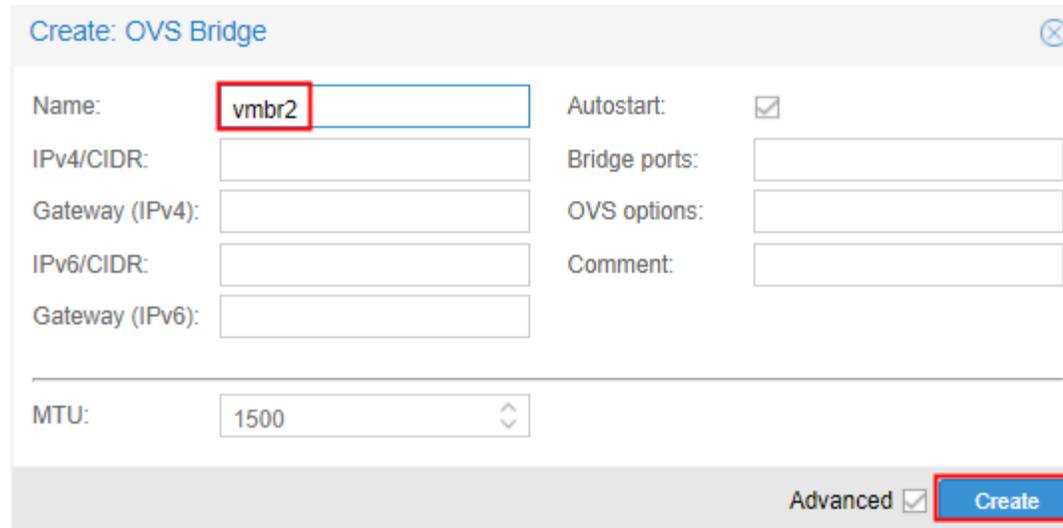
# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (1)

- Membuat **OVS Bridge** dengan nama "**vmbr2**" yang digunakan sebagai **switch** untuk koneksi dari **Container (CT)** yang dibuat pada *node PVE2* ke jaringan.



Pada kotak dialog **Create: OVS Bridge** yang tampil, lengkapi parameter **Name:** dengan **vmbr2**.

Klik tombol **Create** untuk memproses pembuatan **OVS Bridge** tersebut.

A screenshot of the 'Create: OVS Bridge' dialog box. The 'Name' field is filled with 'vmbr2' and is highlighted with a red box. The 'Autostart' checkbox is checked. The 'MTU' field is set to 1500. At the bottom right, the 'Create' button is highlighted with a red box. The 'Advanced' checkbox is also checked.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (2)

- Hasil dari pembuatan **OVS Bridge vmbr2**.

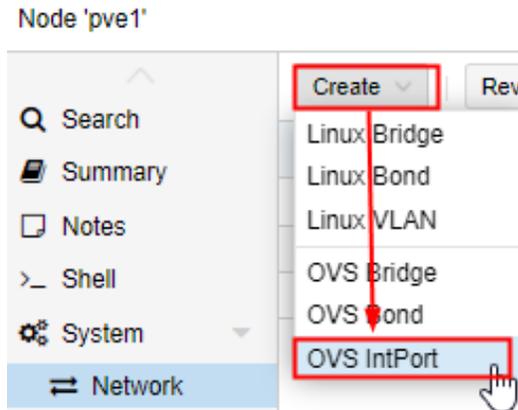
Node 'pve1'

Create | Revert | Edit | Remove | Apply Configuration

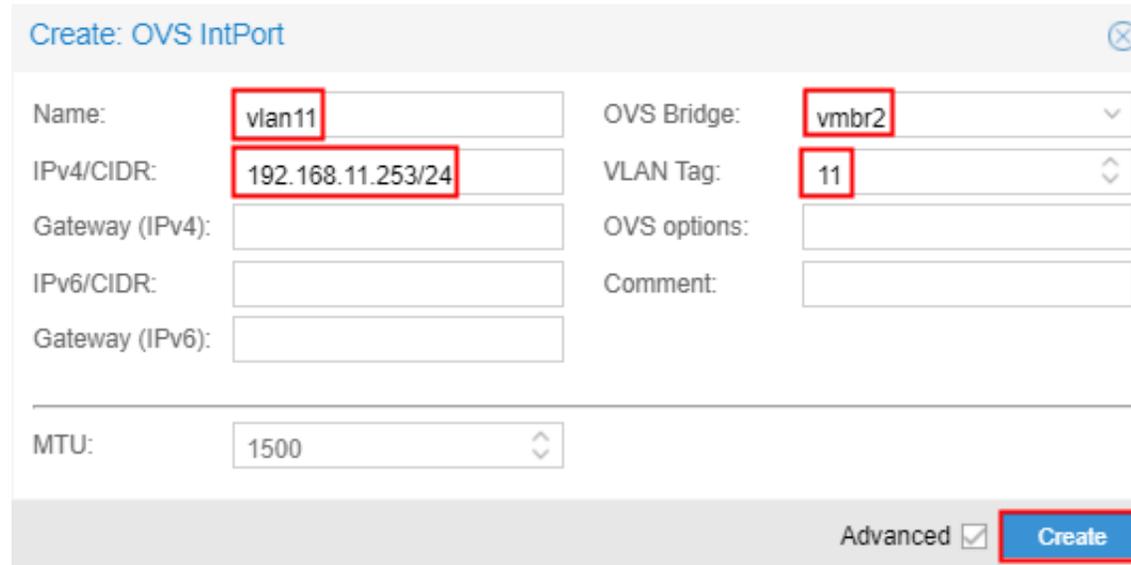
Name	Type ↑	Active	Autostart	VLAN aware
vmbr2	OVS Bridge	No	Yes	No
vmbr1	OVS Bridge	Yes	Yes	No
vmbr0	OVS Bridge	Yes	Yes	No
inband	OVS IntPort	Yes	Yes	No
vtep0	OVS IntPort	Yes	Yes	No
ens32	OVS Port	Yes	Yes	No
ens33	OVS Port	Yes	Yes	No

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (3)

- Membuat **OVS Internal Port (IntPort)** untuk setiap VLAN ID yaitu dengan nama **vlan11** dan **vlan12**. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter **Name:** dengan **vlan11**, **OVS Bridge:** **vmbr2** dan **IPv4/CIDR:** **192.168.11.253/24** serta **VLAN Tag:** **11** untuk VLAN ID 11. Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are: Name: vlan11, OVS Bridge: vmbr2, IPv4/CIDR: 192.168.11.253/24, VLAN Tag: 11, Gateway (IPv4):, OVS options:, IPv6/CIDR:, Comment:, Gateway (IPv6):, and MTU: 1500. A red box highlights the 'Create' button at the bottom right. The 'Advanced' checkbox is checked.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (4)

- Hasil dari pembuatan **OVS IntPort** **vlan11**.

Node 'pve2' Reb... Shutdo... > S... Bulk Actio...

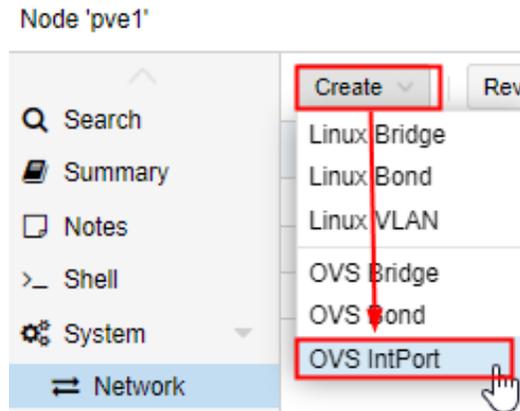
Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	Yes	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.2/24	192.168.169.254	inband management
vlan11	OVS IntPort	No	Yes	No			192.168.11.253/24		
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0				
vmbr2	OVS Bridge	No	Yes	No	vlan11				
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.2/24		

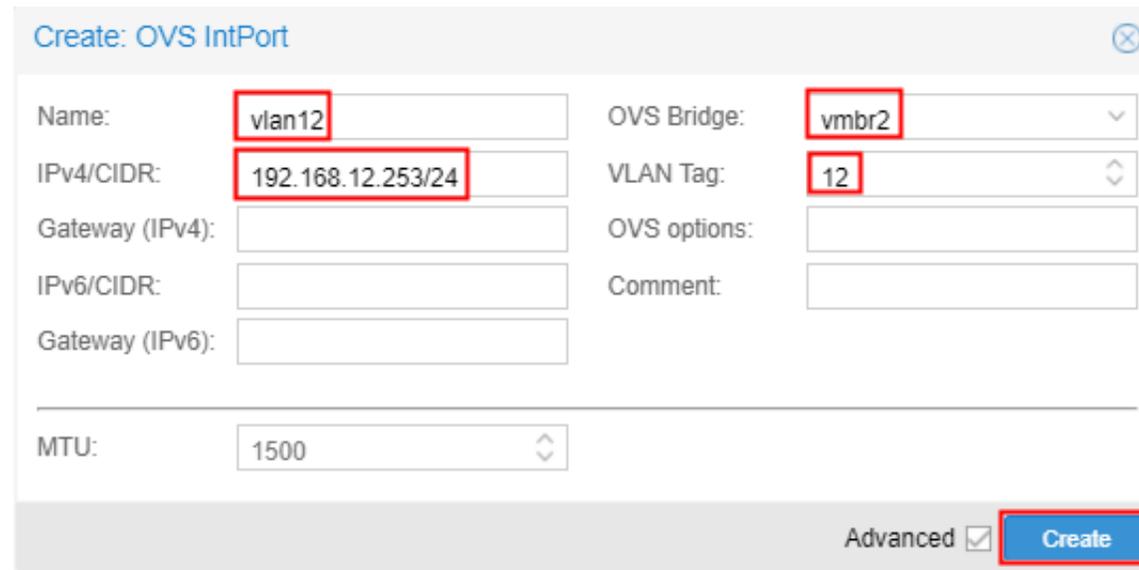
- Dengan cara yang sama lakukan pembuatan **OVS IntPort** untuk **VLAN ID 12**.

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (5)

- Membuat **OVS Internal Port (IntPort)** untuk **vlan12**. Klik tombol **Create > OVS IntPort**.



Pada kotak dialog **Create: OVS IntPort** yang tampil, lengkapi parameter **Name:** dengan **vlan12**, **OVS Bridge:** **vmbr2** dan **IPv4/CIDR:** **192.168.12.253/24** serta **VLAN Tag:** **12** untuk VLAN ID 12. Klik tombol **Create** untuk memproses pembuatan **OVS IntPort** tersebut.

A screenshot of the 'Create: OVS IntPort' dialog box. The fields are filled as follows: Name: vlan12, IPv4/CIDR: 192.168.12.253/24, OVS Bridge: vmbr2, and VLAN Tag: 12. The MTU is set to 1500. The 'Advanced' checkbox is checked, and the 'Create' button is highlighted with a red box.

Name:	<input type="text" value="vlan12"/>	OVS Bridge:	<input type="text" value="vmbr2"/>
IPv4/CIDR:	<input type="text" value="192.168.12.253/24"/>	VLAN Tag:	<input type="text" value="12"/>
Gateway (IPv4):	<input type="text"/>	OVS options:	<input type="text"/>
IPv6/CIDR:	<input type="text"/>	Comment:	<input type="text"/>
Gateway (IPv6):	<input type="text"/>		
MTU:	<input type="text" value="1500"/>		

Advanced

# MEMBUAT OVS BRIDGE DAN INTPORT UNTUK SETIAP VLAN PADA NODE PVE2 (6)

- Hasil dari pembuatan **OVS IntPort vlan12**.

Node 'pve2'

Reb... Shutdo... S... Bulk Actio... ?

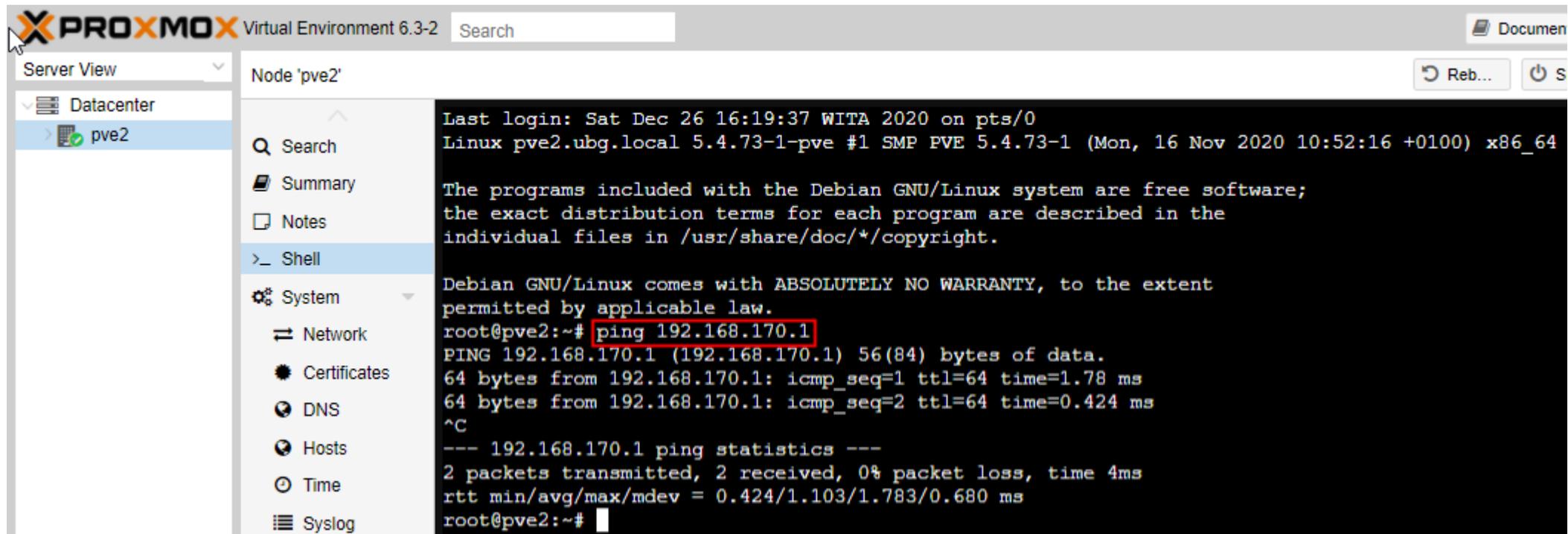
Create | Revert Edit Remove **Apply Configuration**

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway	Comment
ens32	OVS Port	Yes	Yes	No					
ens33	OVS Port	Yes	Yes	No					
inband	OVS IntPort	Yes	Yes	No			192.168.169.2/24	192.168.169.254	inband management
vlan11	OVS IntPort	No	Yes	No			192.168.11.253/24		
<b>vlan12</b>	<b>OVS IntPort</b>	<b>No</b>	<b>Yes</b>	<b>No</b>			<b>192.168.12.253/24</b>		
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband				
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0				
vmbr2	OVS Bridge	No	Yes	No	vlan11 vlan12				
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.2/24		

- Klik tombol **Apply Configuration** untuk menerapkan perubahan konfigurasi jaringan yang telah dilakukan. Tampil kotak dialog **Confirm** dan klik tombol **Yes** untuk menerapkan perubahan jaringan yang tertunda.

# VERIFIKASI KONEKSI UNDERLAY NETWORK DARI NODE PVE2 KE PVE1

- Lakukan verifikasi koneksi *underlay network* dari node **PVE2** ke **PVE1** melalui **Shell** menggunakan perintah `ping 192.168.170.1`.



The screenshot shows the Proxmox VE interface for node 'pve2'. The 'Shell' tab is active, displaying the following output:

```
Last login: Sat Dec 26 16:19:37 WITA 2020 on pts/0
Linux pve2.ubg.local 5.4.73-1-pve #1 SMP PVE 5.4.73-1 (Mon, 16 Nov 2020 10:52:16 +0100) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@pve2:~# ping 192.168.170.1
PING 192.168.170.1 (192.168.170.1) 56(84) bytes of data.
64 bytes from 192.168.170.1: icmp_seq=1 ttl=64 time=1.78 ms
64 bytes from 192.168.170.1: icmp_seq=2 ttl=64 time=0.424 ms
^C
--- 192.168.170.1 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 0.424/1.103/1.783/0.680 ms
root@pve2:~#
```

- Terlihat koneksi dari node **PVE2** ke **PVE1** melalui *underlay network* berhasil dilakukan.

# MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE2 (1)

- Membuat **OVS VXLAN Tunnel** pada *file* `/etc/network/interfaces` yang memuat konfigurasi jaringan dari *node* **PVE2**.

```
# nano /etc/network/interfaces
```

Navigasi ke bagian akhir dari isi *file* tersebut.

```
GNU nano 3.2 /etc/network/interfaces
58     ovs_ports ens32 inband
59
60 auto vmbr1
61 iface vmbr1 inet manual
62     ovs_type OVSBridge
63     ovs_ports ens33 vtep0
64     ovs_mtu 1554
65
66 auto vmbr2
67 iface vmbr2 inet manual
68     ovs_type OVSBridge
69     ovs_ports vlan11 vlan12
70 |
^G Get Help      ^O Write Out    ^W Where Is    ^R Cut Text     ^J Justify     ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace     ^U Uncut Text  ^T To Spell    ^_ Go To Line
```

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE2 (2)

- Lakukan penambahan 6 (enam) baris kode konfigurasi pembuatan **VXLAN Tunnel** berikut pada baris 66:

```
auto vxlan0
iface vxlan0 inet manual
    ovs_type OVSTunnel
    ovs_bridge vmbr2
    ovs_tunnel_type vxlan
    ovs_tunnel_options options:remote_ip=192.168.170.1 options:key=100 options:dst_port=4789
```

- Selain itu tambahkan *interface vxlan0* sebagai *port* anggota dari *bridge vmbr2* pada bagian akhir dari baris 76, seperti berikut:

```
auto vmbr2
iface vmbr2 inet manual
    ovs_type OVSBridge
    ovs_ports vlan11 vlan12 vxlan0
```

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE2 (3)

- Hasil akhir dari keseluruhan pengaturan, seperti terlihat pada baris 66 sampai dengan 71 dan baris 76.

```
66 auto vxlan0
67 iface vxlan0 inet manual
68     ovs_type OVSTunnel
69     ovs_bridge vobr2
70     ovs_tunnel_type vxlan
71     ovs_tunnel_options options:remote_ip=192.168.170.1 options:key=100 options:dst_port=4789
72
73 auto vobr2
74 iface vobr2 inet manual
75     ovs_type OVSBridge
76     ovs_ports vlan11 vlan12 vxlan0
77
```

- Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE2 (4)

Penjelasan baris kode konfigurasi pembuatan **VXLAN Tunnel**:

- Baris 66: digunakan untuk mengaktifkan *interface* **vxlan0** ketika proses *booting* sistem.
- Baris 67: digunakan untuk membuat *interface* jaringan bernama **vxlan0** tanpa pengaturan alamat IP.
- Baris 68: digunakan untuk menentukan jenis dari *interface* OVS yaitu **OVSTunnel**.
- Baris 69: digunakan untuk menentukan agar *interface* **vxlan0** menjadi anggota dari bridge **vmbr2**.
- Baris 70: digunakan untuk menentukan jenis **OVS Tunnel** yaitu **vxlan**.
- Baris 71: digunakan untuk mengatur **options** dari *interface* **OVSTunnel** meliputi **remote\_ip**, **key**, dan **dst\_port**. **remote\_ip** digunakan untuk menentukan alamat IP dari **VTEP** lawan (*node* **PVE1**) yaitu **192.168.170.1**. **key** digunakan untuk menentukan **VXLAN Network Identifier (VNI)** yaitu **100**. **dst\_port** digunakan untuk menentukan nomor port tujuan UDP dari trafik VXLAN yaitu **4789**.
- Baris 76: digunakan untuk mengatur *port-port* yang menjadi anggota dari *bridge* **vmbr2** yaitu **vlan11**, **vlan12** dan **vxlan0**.

# MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE2 (4)

- Me-restart service *networking* untuk menerapkan perubahan yang telah dilakukan.

```
# systemctl restart networking
```

- Verifikasi hasil pembuatan OVS VXLAN Tunnel pada Web GUI.

Node 'pve2'

Reb... Shutdo... S... Bulk Actio... ?

Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN a...	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	OVS Port	Yes	Yes	No				
ens33	OVS Port	Yes	Yes	No				
inband	OVS IntPort	Yes	Yes	No			192.168.169.2/24	192.168.169.254
vlan11	OVS IntPort	Yes	Yes	No			192.168.11.253/24	
vlan12	OVS IntPort	Yes	Yes	No			192.168.12.253/24	
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband			
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0			
vmbr2	OVS Bridge	Yes	Yes	No	vlan11 vlan12 vxlan0			
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.2/24	
vxlan0	Unknown	No	Yes	No				

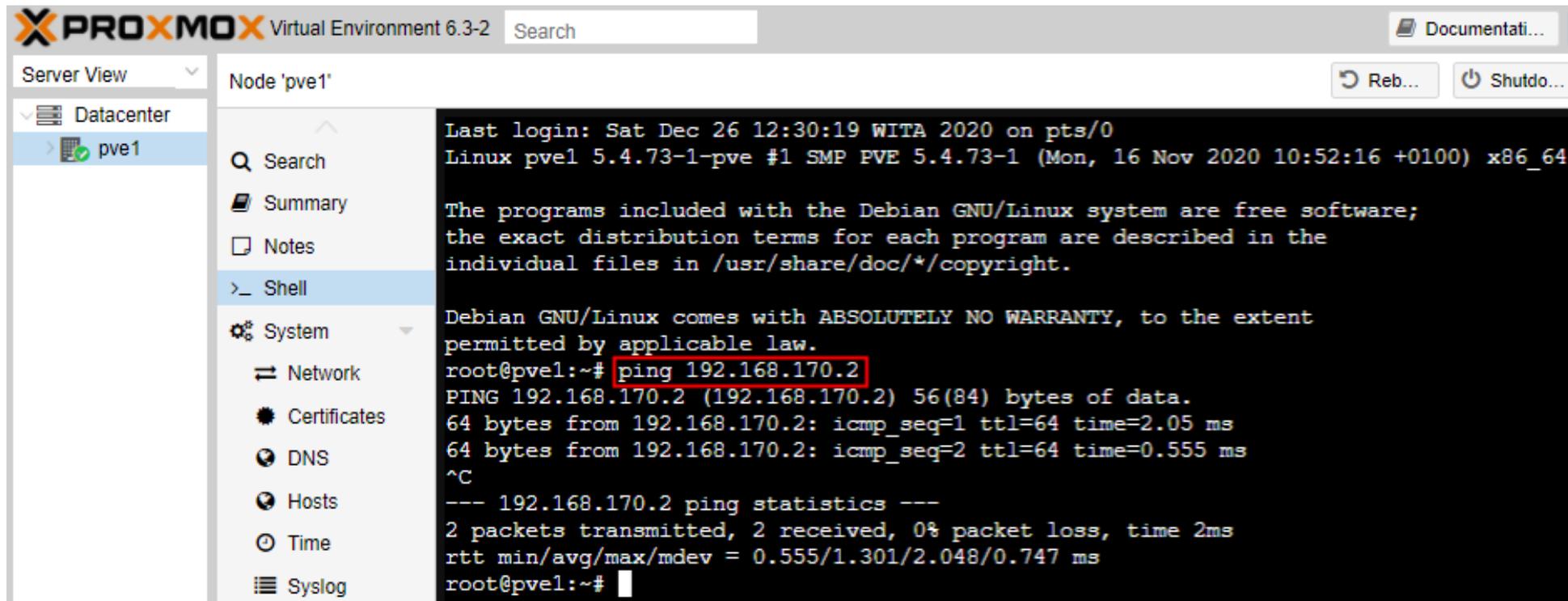
Terlihat telah terdapat **interface tunnel vxlan0** dan menjadi anggota dari **bridge vmbr2**.



# KONFIGURASI OVS VXLAN TUNNEL PADA NODE PVE1

# VERIFIKASI KONEKSI UNDERLAY NETWORK DARI NODE PVE1 KE PVE2

- Lakukan verifikasi koneksi *underlay network* dari node **PVE1** ke **PVE2** melalui **Shell** menggunakan perintah `ping 192.168.170.2`.



The screenshot shows the Proxmox VE interface for Node 'pve1'. The 'Shell' tab is active, displaying a terminal window. The terminal output shows the system's login information, followed by the execution of the command `ping 192.168.170.2`. The output of the ping command indicates a successful connection with 2 packets transmitted and received, 0% packet loss, and a total time of 2ms. The command `ping 192.168.170.2` is highlighted with a red box in the original image.

```
PROXMOX Virtual Environment 6.3-2 Search Documentati...
Server View Node 'pve1'
Datacenter
  pve1
  Search
  Summary
  Notes
  Shell
  System
  Network
  Certificates
  DNS
  Hosts
  Time
  Syslog

Last login: Sat Dec 26 12:30:19 WITA 2020 on pts/0
Linux pve1 5.4.73-1-pve #1 SMP PVE 5.4.73-1 (Mon, 16 Nov 2020 10:52:16 +0100) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@pve1:~# ping 192.168.170.2
PING 192.168.170.2 (192.168.170.2) 56(84) bytes of data:
64 bytes from 192.168.170.2: icmp_seq=1 ttl=64 time=2.05 ms
64 bytes from 192.168.170.2: icmp_seq=2 ttl=64 time=0.555 ms
^C
--- 192.168.170.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2ms
rtt min/avg/max/mdev = 0.555/1.301/2.048/0.747 ms
root@pve1:~#
```

- Terlihat koneksi dari node **PVE1** ke **PVE2** melalui *underlay network* berhasil dilakukan.

# MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE1 (1)

- Membuat **OVS VXLAN Tunnel** pada *file* `/etc/network/interfaces` yang memuat konfigurasi jaringan dari *node* **PVE1**.

```
# nano /etc/network/interfaces
```

Navigasi ke bagian akhir dari isi *file* tersebut.

```
GNU nano 3.2 /etc/network/interfaces
58     ovs_ports ens32 inband
59
60 auto vubr1
61 iface vubr1 inet manual
62     ovs_type OVSBridge
63     ovs_ports ens33 vtep0
64     ovs_mtu 1554
65
66 auto vubr2
67 iface vubr2 inet manual
68     ovs_type OVSBridge
69     ovs_ports vlan11 vlan12
70 |
^G Get Help      ^O Write Out    ^W Where Is     ^R Cut Text     ^J Justify      ^C Cur Pos
^X Exit          ^R Read File    ^\ Replace      ^U Uncut Text   ^T To Spell     ^_ Go To Line
```

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE1 (2)

- Lakukan penambahan 6 (enam) baris kode konfigurasi pembuatan **VXLAN Tunnel** berikut pada baris 66:

```
auto vxlan0
iface vxlan0 inet manual
    ovs_type OVSTunnel
    ovs_bridge vmbr2
    ovs_tunnel_type vxlan
    ovs_tunnel_options options:remote_ip=192.168.170.2 options:key=100 options:dst_port=4789
```

- Selain itu tambahkan *interface* **vxlan0** sebagai *port* anggota dari **bridge vmbr2** pada bagian akhir dari baris 76, seperti berikut:

```
auto vmbr2
iface vmbr2 inet manual
    ovs_type OVSBridge
    ovs_ports vlan11 vlan12 vxlan0
```

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE1 (3)

- Hasil akhir dari keseluruhan pengaturan, seperti terlihat pada baris 66 sampai dengan 71 dan baris 76.

```
GNU nano 3.2 /etc/network/interfaces
65
66 auto vxlan0
67 iface vxlan0 inet manual
68     ovs_type OVSTunnel
69     ovs_bridge vbr2
70     ovs_tunnel_type vxlan
71     ovs_tunnel_options options:remote_ip=192.168.170.2 options:key=100 options:dst_port=4789
72
73 auto vbr2
74 iface vbr2 inet manual
75     ovs_type OVSBridge
76     ovs_ports vlan11 vlan12 vxlan0
77

[ Wrote 76 lines ]
^G Get Help      ^O Write Out    ^W Where Is    ^K Cut Text    ^J Justify     ^C Cur Pos     M-U Undo
^X Exit          ^R Read File   ^\ Replace     ^U Uncut Text  ^T To Spell   ^_ Go To Line   M-E Redo
```

- Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari editor **nano**.

## MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE1 (4)

Penjelasan baris kode konfigurasi pembuatan **VXLAN Tunnel**:

- Baris 66: digunakan untuk mengaktifkan *interface* `vxlan0` ketika proses *booting* sistem.
- Baris 67: digunakan untuk membuat *interface* jaringan bernama `vxlan0` tanpa pengaturan alamat IP.
- Baris 68: digunakan untuk menentukan jenis dari *interface* OVS yaitu **OVSTunnel**.
- Baris 69: digunakan untuk menentukan agar *interface* `vxlan0` menjadi anggota dari bridge `vmbr2`.
- Baris 70: digunakan untuk menentukan jenis **OVS Tunnel** yaitu `vxlan`.
- Baris 71: digunakan untuk mengatur **options** dari *interface* **OVSTunnel** meliputi `remote_ip`, `key`, dan `dst_port`. `remote_ip` digunakan untuk menentukan alamat IP dari **VTEP** lawan (*node* `PVE1`) yaitu `192.168.170.2`. `key` digunakan untuk menentukan **VXLAN Network Identifier (VNI)** yaitu `100`. `dst_port` digunakan untuk menentukan nomor port tujuan UDP dari trafik VXLAN yaitu `4789`.
- Baris 76: digunakan untuk mengatur *port-port* yang menjadi anggota dari *bridge* `vmbr2` yaitu `vlan11`, `vlan12` dan `vxlan0`.

# MEMBUAT OVS VXLAN TUNNEL PADA NODE PVE1 (4)

- *Me-restart service networking* untuk menerapkan perubahan yang telah dilakukan.

```
# systemctl restart networking
```

- Verifikasi hasil pembuatan OVS VXLAN Tunnel pada Web GUI.

Node 'pve1'

Reb... Shutdo... S... Bulk Actio... H...

Create | Revert | Edit | Remove | Apply Configuration

Name ↑	Type	Active	Autostart	VLAN aware	Ports/Slaves	Bond Mode	CIDR	Gateway
ens32	OVS Port	Yes	Yes	No				
ens33	OVS Port	Yes	Yes	No				
inband	OVS IntPort	Yes	Yes	No			192.168.169.1/24	192.168.169.254
vlan11	OVS IntPort	Yes	Yes	No			192.168.11.254/24	
vlan12	OVS IntPort	Yes	Yes	No			192.168.12.254/24	
vmbr0	OVS Bridge	Yes	Yes	No	ens32 inband			
vmbr1	OVS Bridge	Yes	Yes	No	ens33 vtep0			
vmbr2	OVS Bridge	Yes	Yes	No	vlan11 vlan12 vxlan0			
vtep0	OVS IntPort	Yes	Yes	No			192.168.170.1/24	
vxlan0	Unknown	No	Yes	No				

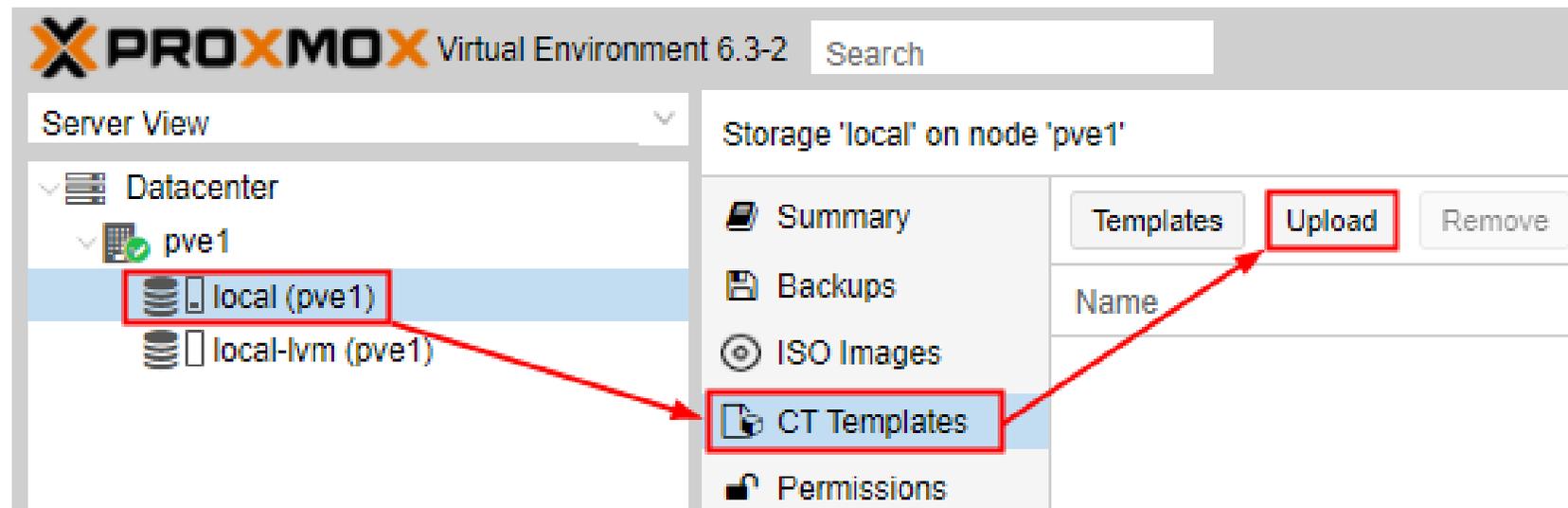
Terlihat telah terdapat **interface tunnel vxlan0** dan menjadi anggota dari **bridge vmbr2**.



# MANAJEMEN CONTAINER DI NODE PVE1

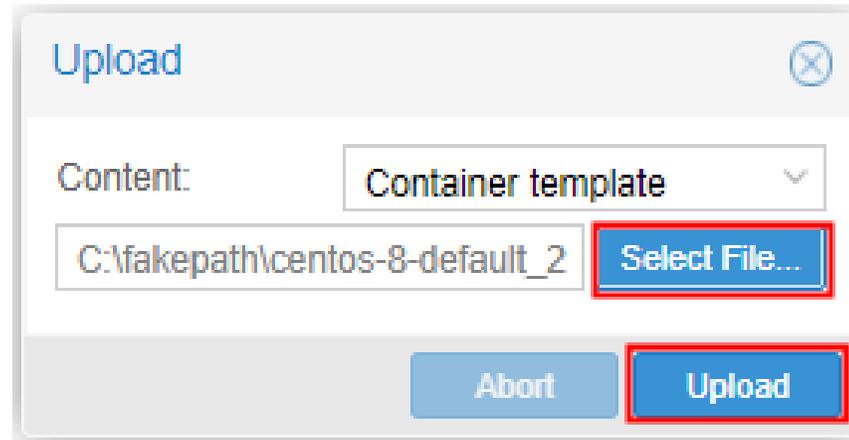
# MENGUNGGAH CONTAINER TEMPLATE CENTOS 8 PADA PVE1 (1)

- Akses *node PVE1* di bawah menu *Datacenter* yang terdapat pada panel sebelah kiri dan pilih *storage local (pve1)*. Pada panel detail sebelah kanan, pilih menu *CT Templates* dan klik tombol *Upload* untuk mengunggah *file template container CentOS 8*.



## MENGUNGGAH CONTAINER TEMPLATE CENTOS 8 PADA PVE1 (2)

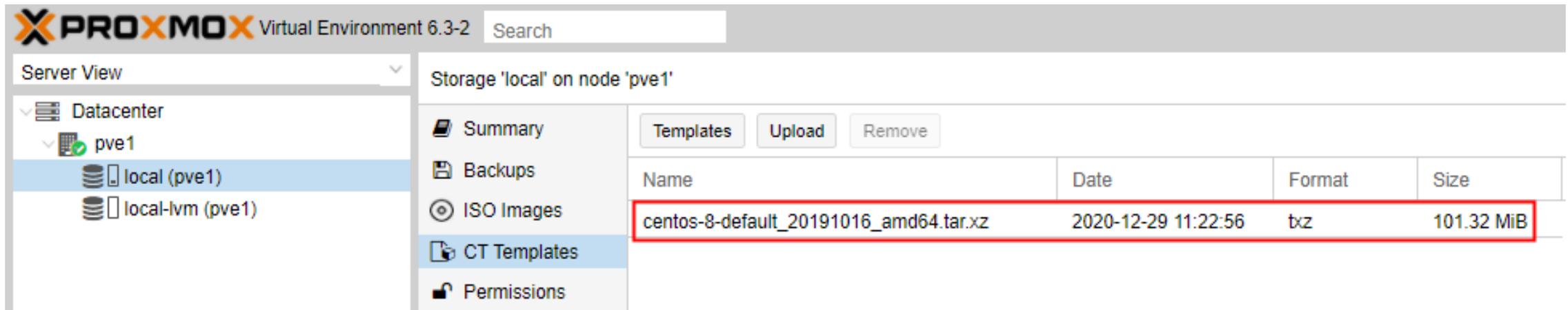
- Tampil kotak dialog **Upload** dan tekan tombol **Select File...** untuk mengarahkan ke lokasi direktori penyimpanan **file template CentOS 8**, sebagai contoh terdapat di **D:\Master\centos-8-default\_20191016\_amd64.tar.xz**



- **File template container CentOS 8** tersebut dapat diperoleh melalui alamat <http://download.proxmox.com/images/system/>
- Tekan tombol **Upload** dan tunggu hingga proses pengunggahan file selesai dilakukan.

# HASIL UNGGAH CONTAINER TEMPLATE CENTOS 8 PADA NODE PVE1

- Terlihat **CT Template centos-8-default\_20191016\_amd64.tar.xz** telah berhasil diunggah ke *storage local*.

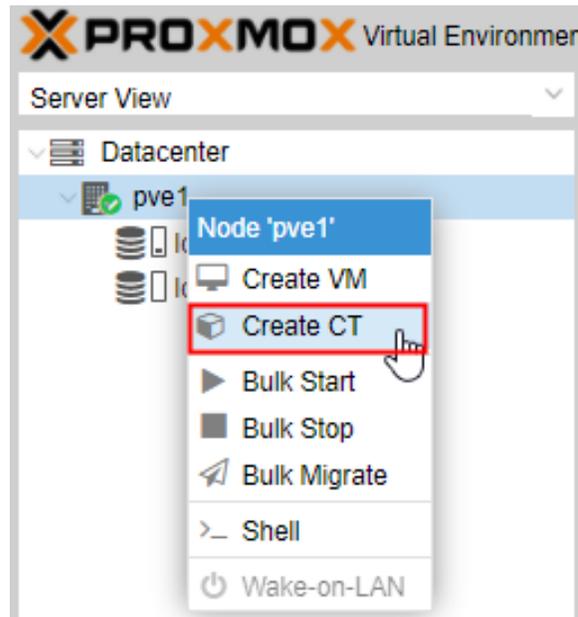


The screenshot shows the Proxmox Virtual Environment 6.3-2 interface. On the left, the 'Server View' sidebar shows the 'Datacenter' expanded to 'pve1', with 'local (pve1)' selected. The main panel displays 'Storage 'local' on node 'pve1'' with a 'CT Templates' menu item highlighted. A table lists the uploaded template, with the row for 'centos-8-default\_20191016\_amd64.tar.xz' highlighted in red. The table has columns for Name, Date, Format, and Size.

Name	Date	Format	Size
centos-8-default_20191016_amd64.tar.xz	2020-12-29 11:22:56	txz	101.32 MiB

# MEMBUAT CONTAINER PADA NODE PVE1 (1)

- Klik kanan pada *node* “pve1” dibawah menu **Datacenter** di panel sebelah kiri dan memilih **Create CT**.



## MEMBUAT CONTAINER PADA NODE PVE1 (2)

- Tampil kotak dialog **Create: LXC Container**. Terdapat beberapa parameter yang diatur di bagian **General** dari LXC Container.
- Pada parameter **CT ID:**, masukkan ID dari container yang akan dibuat yaitu **111**.
- Pada parameter **Hostname:**, masukkan nama komputer dan nama domain dari Container CentOS 8, sebagai contoh “**server111.ubg.local**”.
- Parameter **Password:** dan **Confirm password:**, masukkan sandi login dari user “**root**” untuk *container CentOS 8*, sebagai contoh “**12345678**”.
- Klik tombol **Next** untuk melanjutkan.

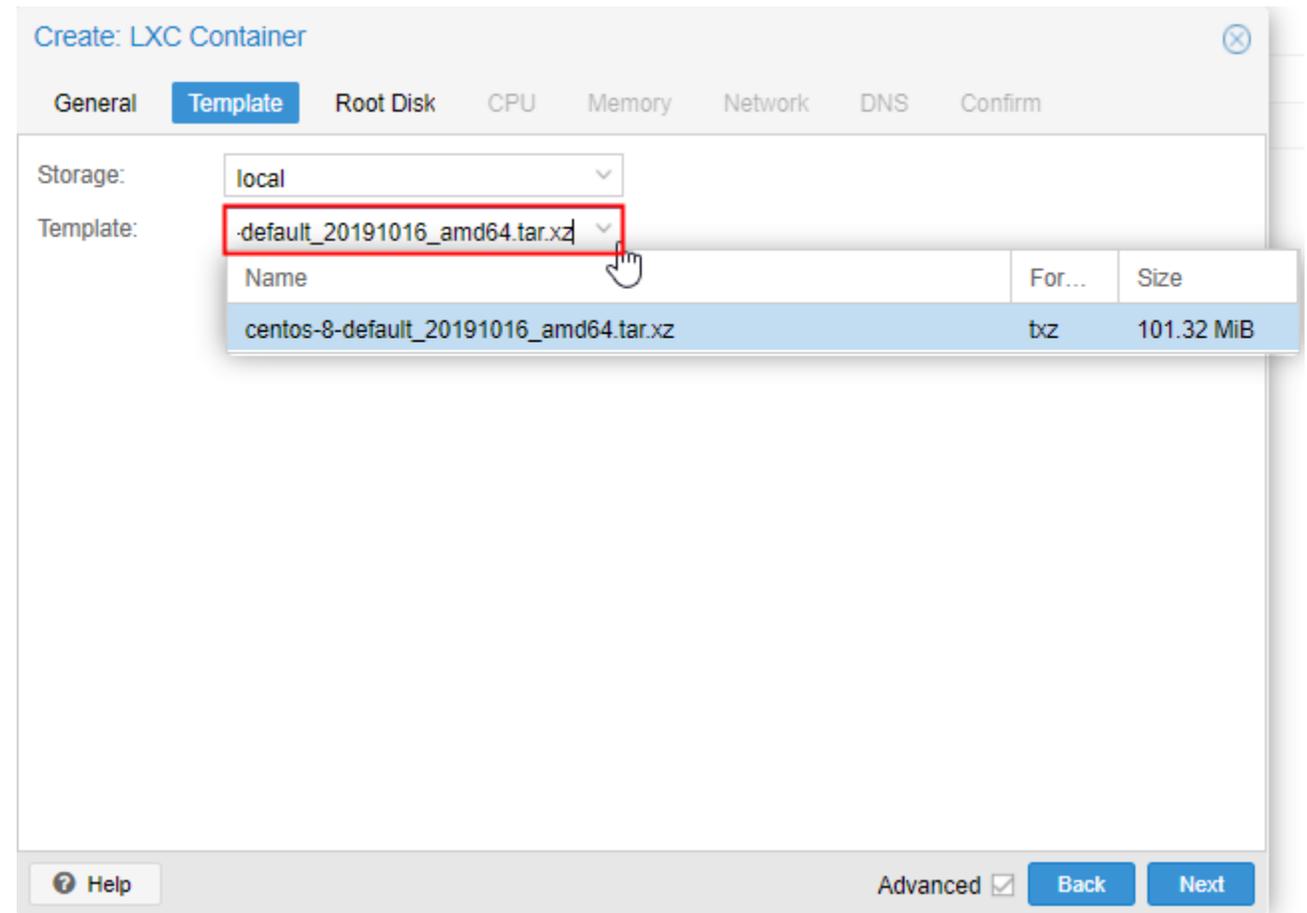
The screenshot shows the 'Create: LXC Container' dialog box with the following fields and values:

- Node:** pve1
- CT ID:** 111
- Hostname:** server111.ubg.local
- Unprivileged container:**
- Resource Pool:** (empty)
- Password:** (masked with dots)
- Confirm password:** (masked with dots)
- SSH public key:** (empty)

Buttons: **Load SSH Key File**, **Help**, **Advanced** (checked), **Back**, **Next**.

# MEMBUAT CONTAINER PADA NODE PVE1 (3)

- Tampil kotak dialog pengaturan bagian **Template** dari **LNX Container**. Pilih **centos-8-default\_20191016\_amd64.tar.xz** pada parameter **Template**:
- Klik tombol **Next** untuk melanjutkan.



# MEMBUAT CONTAINER PADA NODE PVE1 (4)

- Tampil kotak dialog pengaturan bagian **Root Disk** dari **LNX Container**. Lakukan penyesuaian ukuran hardisk yang digunakan pada parameter **Disk size (GB)**:, sebagai contoh menggunakan **3 GB**
- Klik tombol **Next** untuk melanjutkan.

Create: LXC Container

General Template **Root Disk** CPU Memory Network DNS Confirm

Storage: local-lvm

Disk size (GiB): 3

Enable quota:

ACLs: Default

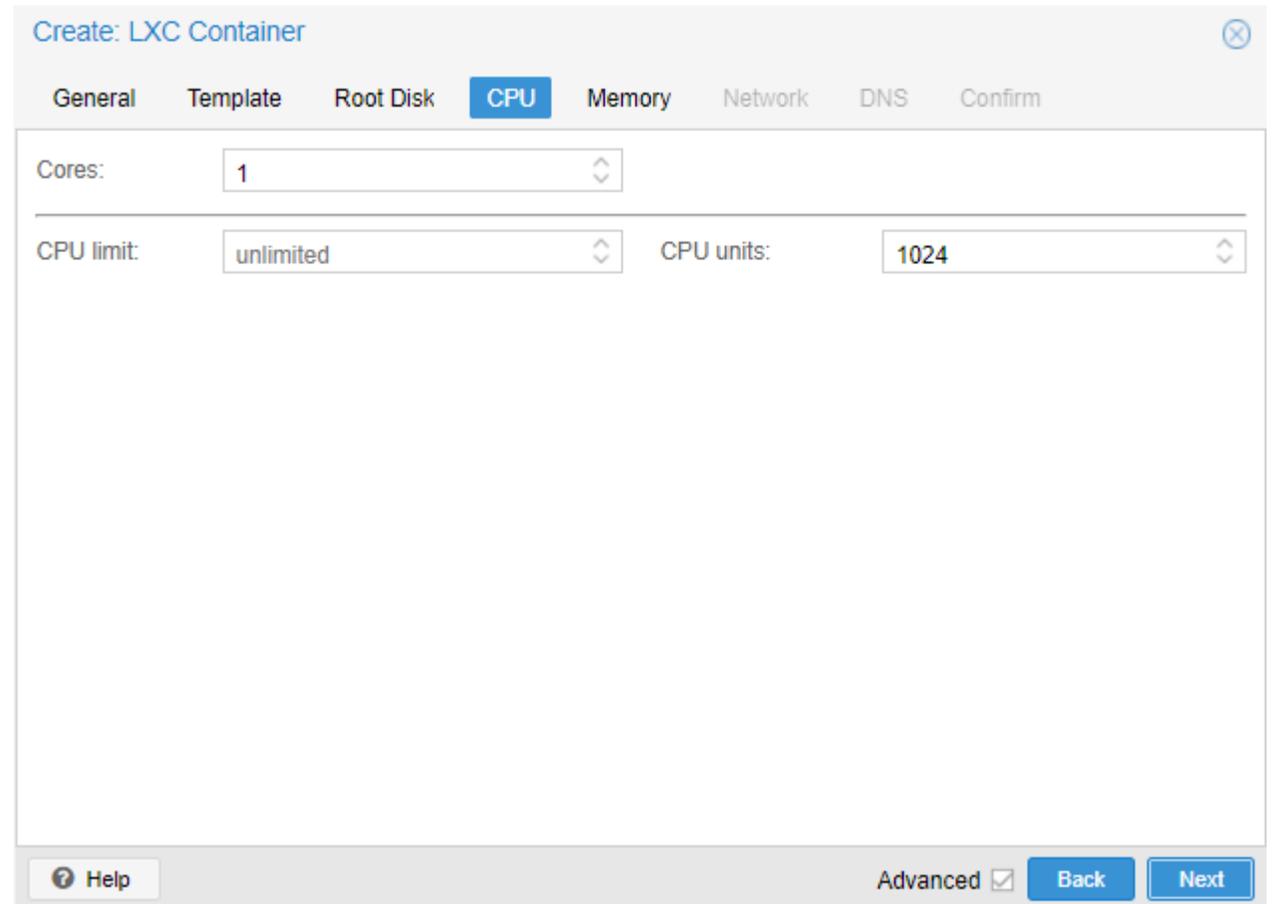
Mount options:

Skip replication:

Help Advanced  Back Next

## MEMBUAT CONTAINER PADA NODE PVE1 (5)

- Tampil kotak dialog pengaturan **CPU** dari **LNX Container**. Pada parameter **Cores**:, lakukan penyesuaian jumlah Core CPU yang digunakan apabila diperlukan. Secara default bernilai **1** (satu).
- Klik tombol **Next** untuk melanjutkan.

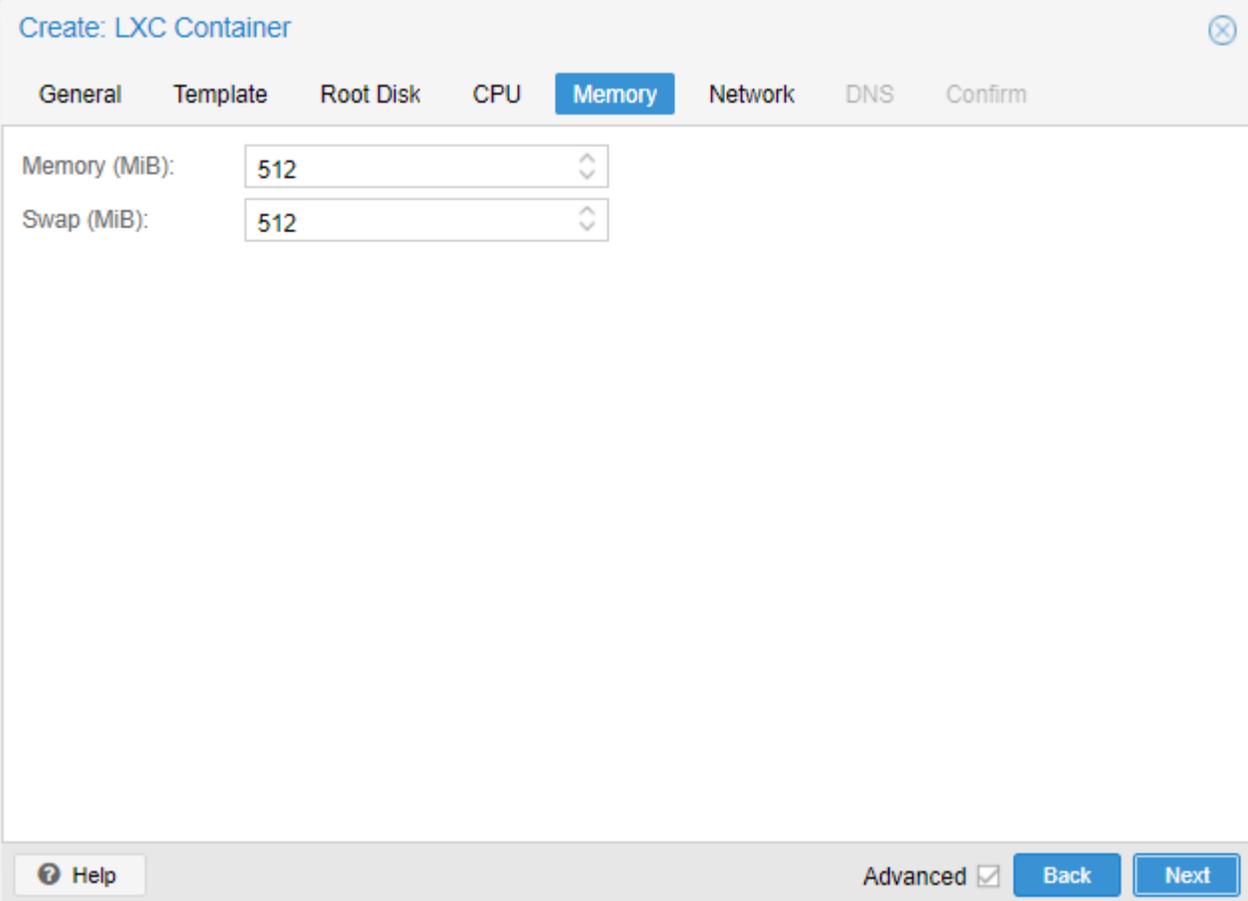


The screenshot shows the 'Create: LXC Container' dialog box with the 'CPU' tab selected. The 'Cores' field is set to 1, 'CPU limit' is unlimited, and 'CPU units' is 1024. The 'Next' button is highlighted.

Field	Value
Cores:	1
CPU limit:	unlimited
CPU units:	1024

## MEMBUAT CONTAINER PADA NODE PVE1 (6)

- Tampil kotak dialog pengaturan **Memory** dari **LNX Container**. Terdapat 2 (dua) parameter yang dapat diatur yaitu **Memory (MB)** dan **Swap (MB)**. Secara default masing-masing parameter tersebut bernilai **512 MB**.
- Klik tombol **Next** untuk melanjutkan.



The screenshot shows a web-based interface for creating an LXC container. The title bar reads "Create: LXC Container" with a close button. Below the title bar are several tabs: "General", "Template", "Root Disk", "CPU", "Memory" (which is selected and highlighted in blue), "Network", "DNS", and "Confirm". The main content area contains two input fields: "Memory (MiB):" and "Swap (MiB):", both of which have the value "512" entered. Each input field has a small up/down arrow icon to its right. At the bottom of the dialog, there is a "Help" button with a question mark icon, an "Advanced" checkbox which is checked, and two buttons labeled "Back" and "Next".

# MEMBUAT CONTAINER PADA NODE PVE1 (7)

- Tampil kotak dialog pengaturan **Network** dari **LNX Container**.
- Pada parameter **Bridge:**, pilih **vbr2**.
- Pada parameter **VLAN Tag:**, masukkan VLAN ID **11**.
- Pada parameter **IPv4/CIDR:** masukkan alamat IP dan subnetmask yang digunakan oleh *container CentOS 8* yaitu **192.168.11.111/24**. Sedangkan pada bagian **Gateway (IPv4):**, masukkan alamat IP **192.168.11.254**.
- Klik tombol **Next** untuk melanjutkan.

The screenshot shows the 'Create: LXC Container' dialog box with the 'Network' tab selected. The 'Bridge' field is set to 'vbr2', 'VLAN Tag' is '11', 'IPv4/CIDR' is '192.168.11.111/24', and 'Gateway (IPv4)' is '192.168.11.254'. The 'Next' button is highlighted.

Field	Value
Name	eth0
MAC address	auto
Bridge	vbr2
VLAN Tag	11
Rate limit (MB/s)	unlimited
Firewall	<input checked="" type="checkbox"/>
IPv4	Static
IPv4/CIDR	192.168.11.111/24
Gateway (IPv4)	192.168.11.254
IPv6	Static
IPv6/CIDR	None
Gateway (IPv6)	

## MEMBUAT CONTAINER PADA NODE PVE1 (8)

- Tampil kotak dialog pengaturan **DNS** dari **LNX Container**.
- Pada parameter **DNS domain**: masukkan nama domain yang digunakan oleh *container CentOS 8*, sebagai contoh menggunakan “**ubg.local**”. Sedangkan pada bagian **DNS server 1**:, masukkan alamat IP dari *Primary Name Server*, sebagai contoh menggunakan alamat IP **192.168.169.254**.
- Klik tombol **Next** untuk melanjutkan.

Create: LXC Container

General Template Root Disk CPU Memory Network **DNS** Confirm

DNS domain:

DNS servers:

Advanced  Back Next

# MEMBUAT CONTAINER PADA NODE PVE1 (9)

- Tampil kotak dialog **Confirm** dari **LNX Container** yang menampilkan ringkasan pengaturan yang telah dilakukan terkait pembuatan *container CentOS 8*.
- Klik tombol **Finish**.

Create: LXC Container

General Template Root Disk CPU Memory Network DNS Confirm

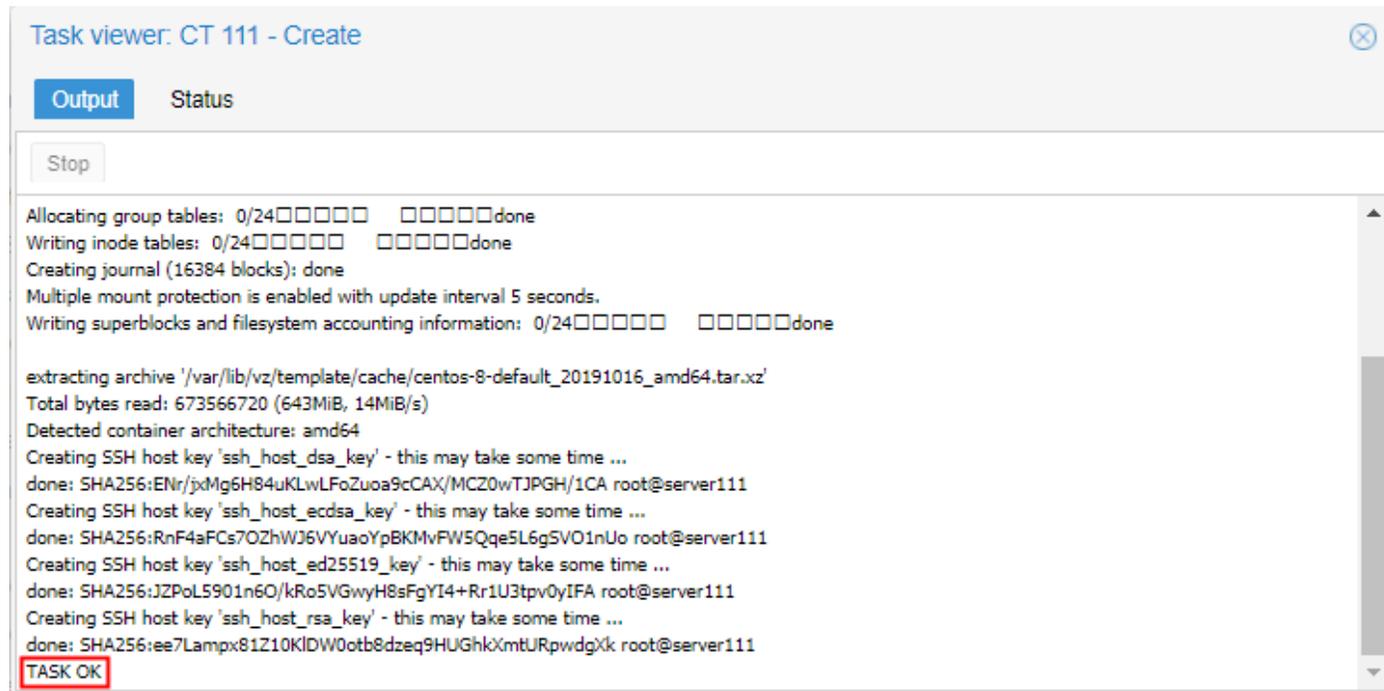
Key ↑	Value
cores	1
hostname	server111.ubg.local
memory	512
nameserver	192.168.169.254
net0	bridge=vibr2,name=eth0,ip=192.168.11.111/24,gw=192.168.11.254,firewall=1,ta...
nodename	pve1
ostemplate	local:vztmpl/centos-8-default_20191016_amd64.tar.xz
pool	
rootfs	local-lvm:3
searchdomain	ubg.local
swap	512
unprivileged	1
vmid	111

Start after created

Advanced  Back Finish

# MEMBUAT CONTAINER PADA NODE PVE1 (10)

- Tampil kotak dialog **Task viewer: CT 111 – Create**. Tunggu hingga proses pembuatan container *CentOS 8* selesai dibuat dimana ditandai dengan pesan **“TASK OK”** pada bagian **Output** dari kotak dialog **Task viewer: CT 111 – Create**.



```
Task viewer: CT 111 - Create
Output Status
Stop
Allocating group tables: 0/24□□□□□□ □□□□□done
Writing inode tables: 0/24□□□□□□ □□□□□done
Creating journal (16384 blocks): done
Multiple mount protection is enabled with update interval 5 seconds.
Writing superblocks and filesystem accounting information: 0/24□□□□□□ □□□□□done

extracting archive '/var/lib/vz/template/cache/centos-8-default_20191016_amd64.tar.xz'
Total bytes read: 673566720 (643MiB, 14MiB/s)
Detected container architecture: amd64
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:ENr/jxMg6H84uKLwLFoZuoa9cCAX/MCZ0wTJPGH/1CA root@server111
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:RnF4aFCs7OZhWJ6VYuoYpBKMvFW5Qqe5L6gSVO1nUo root@server111
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:JZPoL5901n6O/kRo5VGwyH8sFgYI4+Rr1U3tpv0yIFA root@server111
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:ee7Lampx81Z10KIDW0otb8dzeq9HUGhkXmtURpwdgXk root@server111
TASK OK
```

- Tutup kotak dialog **Task viewer: CT 111 – Create**.

# VERIFIKASI HASIL DARI PEMBUATAN CONTAINER PADA NODE PVE1

- Pada panel sebelah kiri dibawah *node pve1* terlihat hasil dari pembuatan *container CentOS 8* dengan ID **111**.

The screenshot shows the Proxmox VE interface. On the left, the 'Server View' sidebar shows the hierarchy: Datacenter > pve1 > 111 (server111.ubg.local). The container '111 (server111.ubg.local)' is highlighted with a red box. Below it are 'local (pve1)' and 'local-lvm (pve1)'. The main panel shows 'Node 'pve1'' with a table of resources:

Type ↑	Description	Disk usage...	Memory us...	CPU usage	Uptime
lxc	111 (server111.ubg.local)				-
storage	local (pve1)	22.1 %			-
storage	local-lvm (pve1)	4.1 %			-

## PEMBUATAN 3 CONTAINER (CT) BARU PADA NODE PVE1

Dengan cara yang sama, lakukan pembuatan 3 (tiga) CT baru dengan ketentuan sebagai berikut:

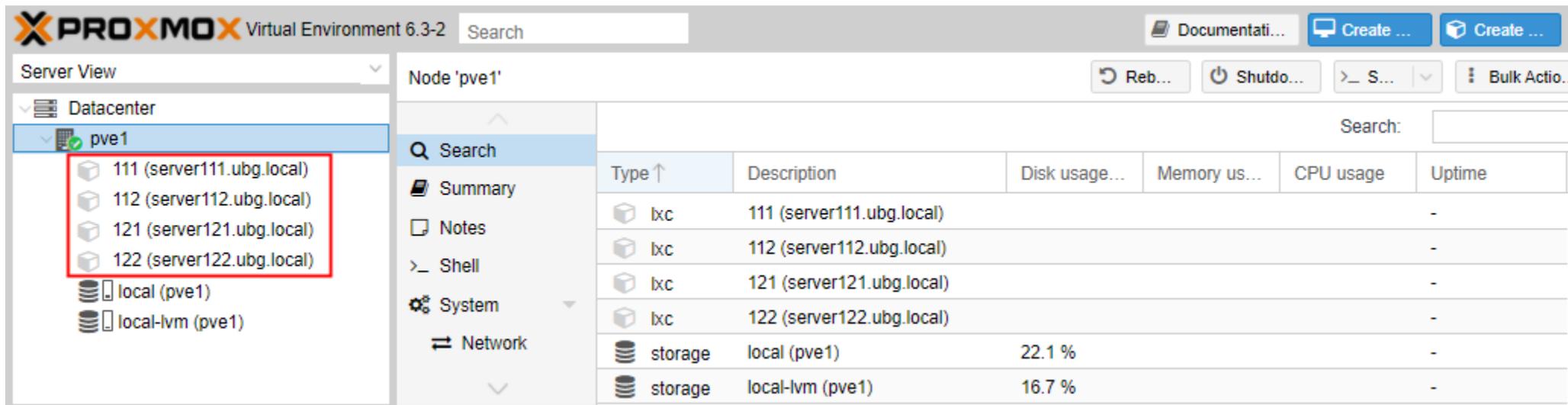
CT ID	Hostname	VLAN ID	IP Address	Gateway
112	server112.ubg.local	11	192.168.11.112/24	192.168.11.254
113	server113.ubg.local	12	192.168.12.121/24	192.168.12.254
114	server114.ubg.local	12	192.168.12.122/24	192.168.12.254

Keseluruhan CT menggunakan nilai yang sama untuk parameter-parameter berikut:

- **Password root** menggunakan nilai “12345678”.
- **Container Template** menggunakan `centos-8-default_20191016_amd64.tar.xz` yang tersimpan di storage local.
- Lokasi penyimpanan *container* adalah di storage `local-lvm` dengan ukuran hardisk sebesar **3 GB**.
- **CPU** menggunakan **1 Core**.
- **Memory** dan swap masing-masing menggunakan **512 MB**.
- **DNS Domain** menggunakan `ubg.local`.
- **DNS Servers** menggunakan `192.168.169.254`.

# HASIL PEMBUATAN KESELURUHAN CONTAINER (CT) DI NODE PVE1

- Terlihat pada *node PVE1* telah terbuat 4 (empat) CT yaitu masing-masing dengan ID 111, 112, 121 dan 122.



The screenshot displays the Proxmox VE 6.3-2 interface. On the left, the 'Server View' sidebar shows the 'Datacenter' tree with 'pve1' selected. Under 'pve1', four containers are listed and highlighted with a red box: 111 (server111.ubg.local), 112 (server112.ubg.local), 121 (server121.ubg.local), and 122 (server122.ubg.local). Below these are 'local (pve1)' and 'local-lvm (pve1)'. The main panel shows the 'Node 'pve1'' view with a table of containers and their resources.

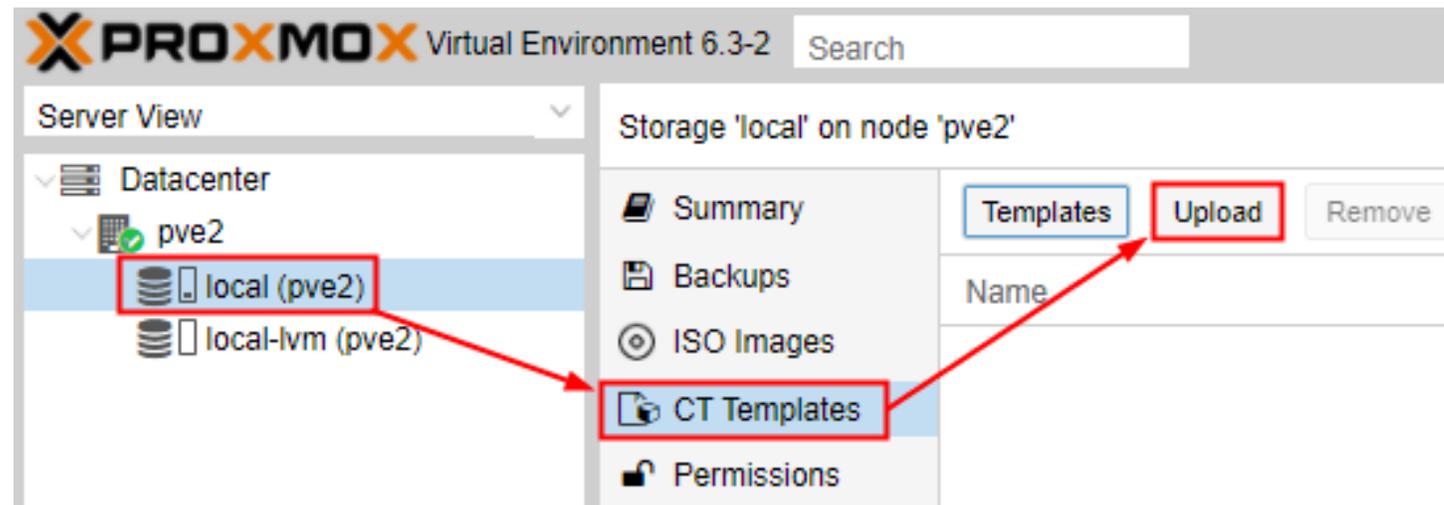
Type ↑	Description	Disk usage...	Memory us...	CPU usage	Uptime
lxc	111 (server111.ubg.local)				-
lxc	112 (server112.ubg.local)				-
lxc	121 (server121.ubg.local)				-
lxc	122 (server122.ubg.local)				-
storage	local (pve1)	22.1 %			-
storage	local-lvm (pve1)	16.7 %			-



# MANAJEMEN CONTAINER DI NODE PVE2

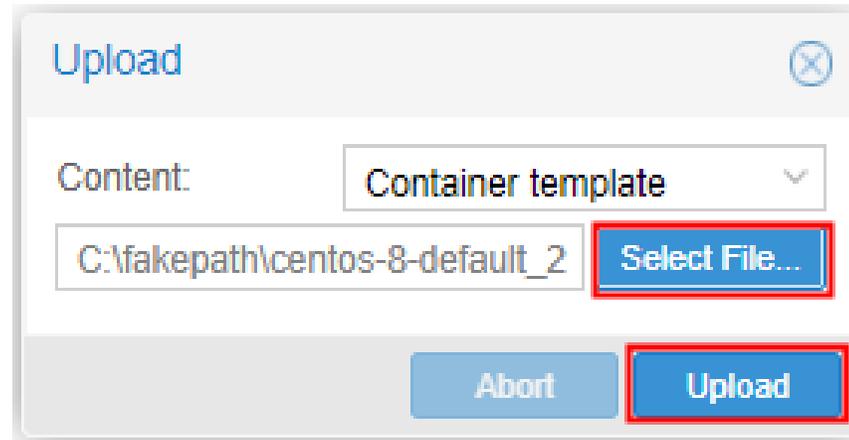
# MENGGUNGAH CONTAINER TEMPLATE CENTOS 8 PADA PVE2 (1)

- Akses *node PVE2* di bawah menu **Datacenter** yang terdapat pada panel sebelah kiri dan pilih *storage local (pve2)*. Pada panel detail sebelah kanan, pilih menu **CT Templates** dan klik tombol **Upload** untuk mengunggah *file template container CentOS 8*.



## MENGGUNGAH CONTAINER TEMPLATE CENTOS 8 PADA PVE2 (2)

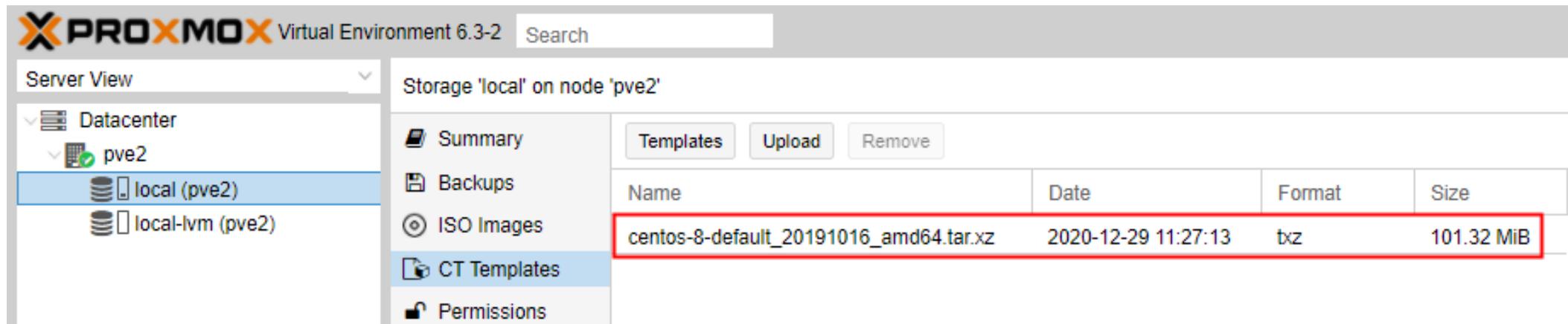
- Tampil kotak dialog **Upload** dan tekan tombol **Select File...** untuk mengarahkan ke lokasi direktori penyimpanan **file template CentOS 8**, sebagai contoh terdapat di **D:\Master\centos-8-default\_20191016\_amd64.tar.xz**



- **File template container CentOS 8** tersebut dapat diperoleh melalui alamat <http://download.proxmox.com/images/system/>
- Tekan tombol **Upload** dan tunggu hingga proses pengunggahan file selesai dilakukan.

# HASIL UNGGAH CONTAINER TEMPLATE CENTOS 8 PADA NODE PVE2

- Terlihat **CT Template centos-8-default\_20191016\_amd64.tar.xz** telah berhasil diunggah ke *storage local*.

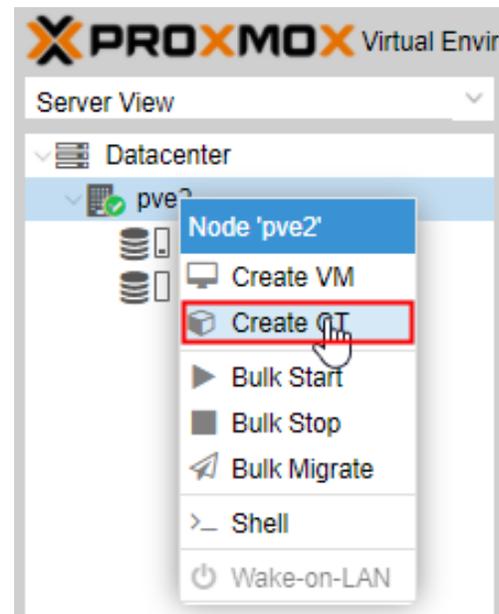


The screenshot shows the Proxmox VE interface for node 'pve2'. The left sidebar shows the navigation tree with 'Datacenter' expanded to 'pve2', and 'local (pve2)' selected under storage. The main panel shows the 'Storage 'local' on node 'pve2'' view. A table lists the uploaded templates, with the entry 'centos-8-default\_20191016\_amd64.tar.xz' highlighted in red. The table has columns for Name, Date, Format, and Size.

Name	Date	Format	Size
centos-8-default_20191016_amd64.tar.xz	2020-12-29 11:27:13	txz	101.32 MiB

# MEMBUAT CONTAINER PADA NODE PVE2 (1)

- Klik kanan pada *node* “pve2” dibawah menu **Datacenter** di panel sebelah kiri dan memilih **Create CT**.



## MEMBUAT CONTAINER PADA NODE PVE2 (2)

- Tampil kotak dialog **Create: LXC Container**. Terdapat beberapa parameter yang diatur di bagian **General** dari LXC Container.
- Pada parameter **CT ID:**, masukkan ID dari container yang akan dibuat yaitu **113**.
- Pada parameter **Hostname:**, masukkan nama komputer dan nama domain dari Container CentOS 8, sebagai contoh “**server113.ubg.local**”.
- Parameter **Password:** dan **Confirm password:**, masukkan sandi login dari user “**root**” untuk *container CentOS 8*, sebagai contoh “**12345678**”.
- Klik tombol **Next** untuk melanjutkan.

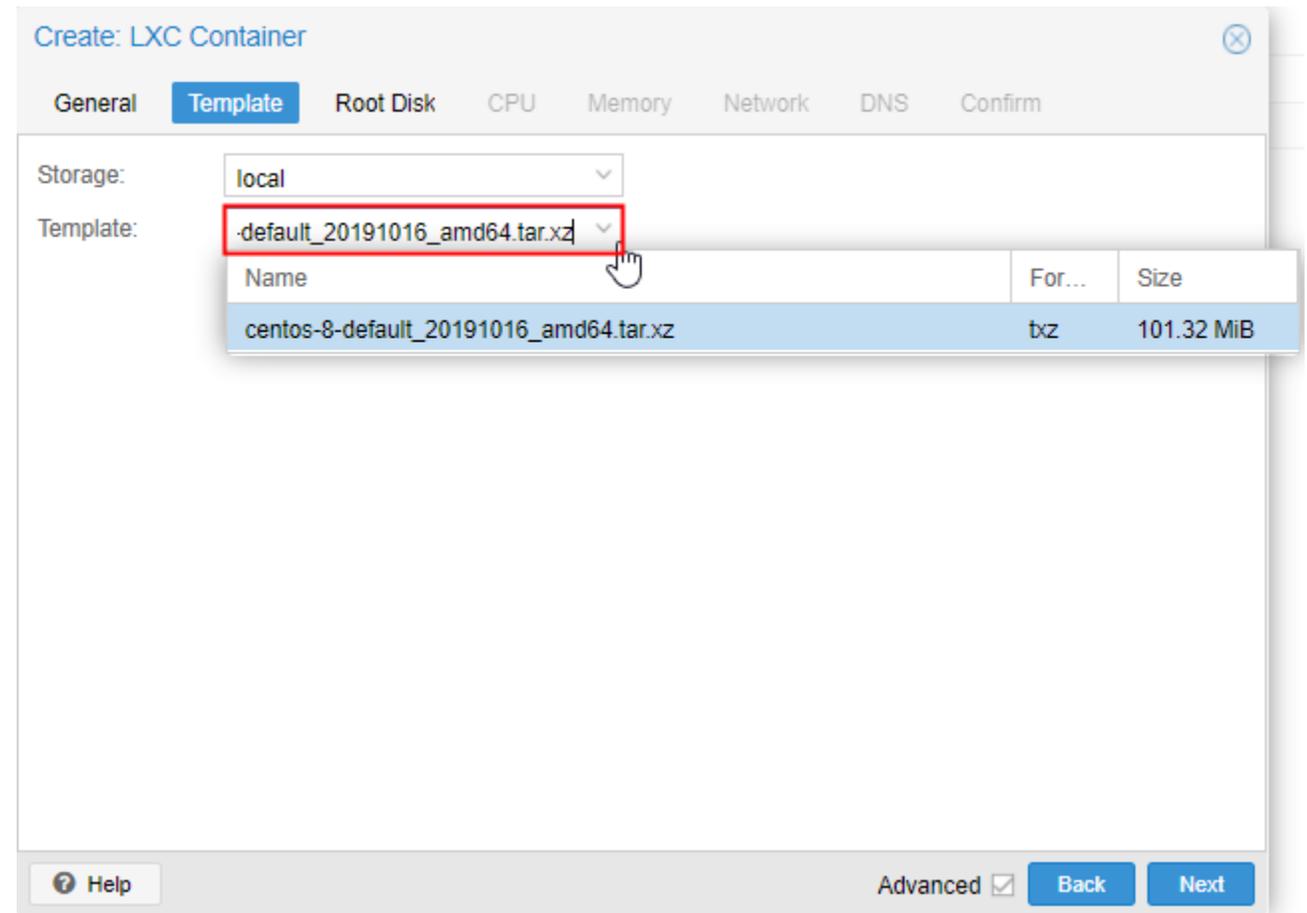
The screenshot shows the 'Create: LXC Container' dialog box with the following fields and values:

- Node:** pve2
- CT ID:** 113
- Hostname:** server113.ubg.local
- Unprivileged container:**
- Resource Pool:** (empty)
- Password:** .....
- Confirm password:** .....
- SSH public key:** (empty)

Buttons: **Load SSH Key File**, **Back**, **Next**

# MEMBUAT CONTAINER PADA NODE PVE2 (3)

- Tampil kotak dialog pengaturan bagian **Template** dari **LNX Container**. Pilih **centos-8-default\_20191016\_amd64.tar.xz** pada parameter **Template**:
- Klik tombol **Next** untuk melanjutkan.



## MEMBUAT CONTAINER PADA NODE PVE2 (4)

- Tampil kotak dialog pengaturan bagian **Root Disk** dari **LNX Container**. Lakukan penyesuaian ukuran hardisk yang digunakan pada parameter **Disk size (GB)**:, sebagai contoh menggunakan **3 GB**
- Klik tombol **Next** untuk melanjutkan.

Create: LXC Container

General Template **Root Disk** CPU Memory Network DNS Confirm

Storage: local-lvm

Disk size (GiB): 3

Enable quota:

ACLs: Default

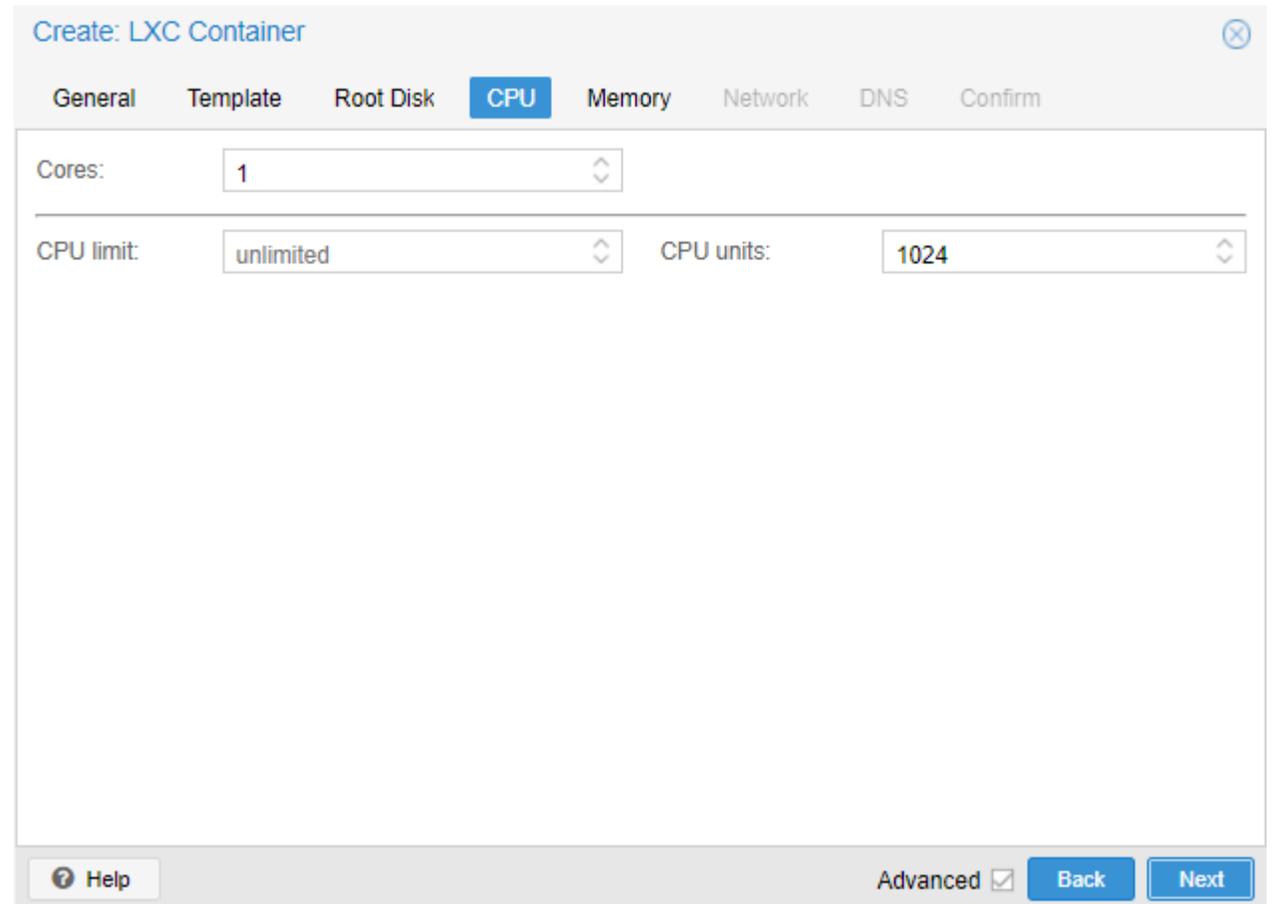
Mount options:

Skip replication:

Help Advanced  Back Next

## MEMBUAT CONTAINER PADA NODE PVE2 (5)

- Tampil kotak dialog pengaturan **CPU** dari **LNX Container**. Pada parameter **Cores**:, lakukan penyesuaian jumlah Core CPU yang digunakan apabila diperlukan. Secara default bernilai **1** (satu).
- Klik tombol **Next** untuk melanjutkan.

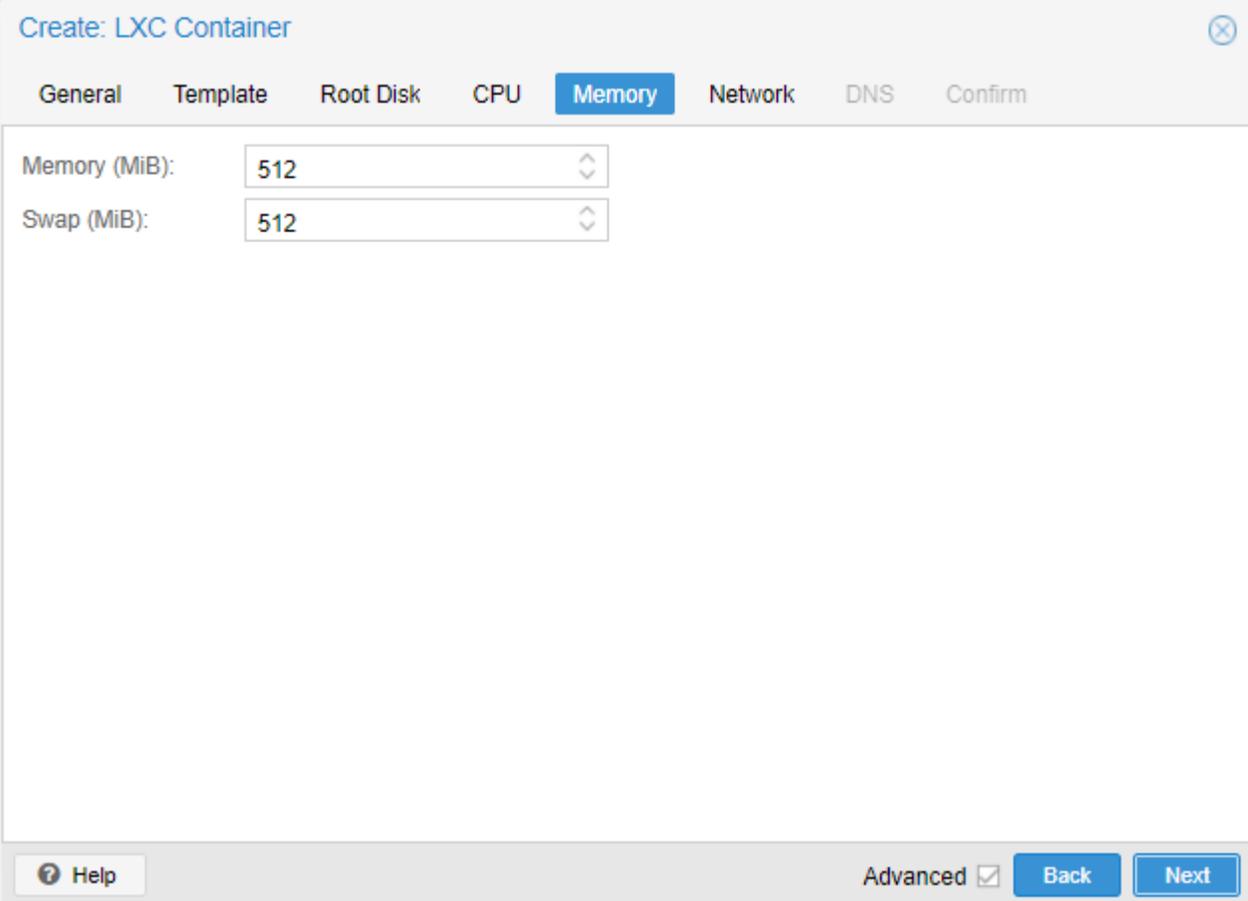


The screenshot shows the 'Create: LXC Container' dialog box with the 'CPU' tab selected. The 'Cores' field is set to 1, 'CPU limit' is unlimited, and 'CPU units' is 1024. The 'Next' button is highlighted.

Field	Value
Cores:	1
CPU limit:	unlimited
CPU units:	1024

## MEMBUAT CONTAINER PADA NODE PVE2 (6)

- Tampil kotak dialog pengaturan **Memory** dari **LNX Container**. Terdapat 2 (dua) parameter yang dapat diatur yaitu **Memory (MB)** dan **Swap (MB)**. Secara default masing-masing parameter tersebut bernilai **512 MB**.
- Klik tombol **Next** untuk melanjutkan.



The screenshot shows a web-based interface for creating an LXC container. The title bar reads "Create: LXC Container" with a close button. Below the title bar are several tabs: "General", "Template", "Root Disk", "CPU", "Memory" (which is selected and highlighted in blue), "Network", "DNS", and "Confirm". The main content area contains two input fields: "Memory (MiB):" with a value of "512" and "Swap (MiB):" with a value of "512". Both fields have a small up/down arrow icon to their right. At the bottom of the dialog, there is a "Help" button with a question mark icon, an "Advanced" checkbox which is checked, and two buttons: "Back" and "Next".

# MEMBUAT CONTAINER PADA NODE PVE2 (7)

- Tampil kotak dialog pengaturan **Network** dari **LNX Container**.
- Pada parameter **Bridge:**, pilih **vmbr2**.
- Pada parameter **VLAN Tag:**, masukkan VLAN ID **11**.
- Pada parameter **IPv4/CIDR:** masukkan alamat IP dan subnetmask yang digunakan oleh *container CentOS 8* yaitu **192.168.11.113/24**. Sedangkan pada bagian **Gateway (IPv4):**, masukkan alamat IP **192.168.11.253**.
- Klik tombol **Next** untuk melanjutkan.

Create: LXC Container

General Template Root Disk CPU Memory **Network** DNS Confirm

Name: eth0 IPv4:  Static  DHCP

MAC address: auto IPv4/CIDR: 192.168.11.113/24

Bridge: vmbr2 Gateway (IPv4): 192.168.11.253

VLAN Tag: 11 IPv6:  Static  DHCP  SLAAC

Rate limit (MB/s): unlimited IPv6/CIDR: None

Firewall:  Gateway (IPv6):

Help Advanced  Back Next

## MEMBUAT CONTAINER PADA NODE PVE2 (8)

- Tampil kotak dialog pengaturan **DNS** dari **LNX Container**.
- Pada parameter **DNS domain**: masukkan nama domain yang digunakan oleh *container CentOS 8*, sebagai contoh menggunakan “**ubg.local**”. Sedangkan pada bagian **DNS server 1**:, masukkan alamat IP dari *Primary Name Server*, sebagai contoh menggunakan alamat IP **192.168.169.254**.
- Klik tombol **Next** untuk melanjutkan.

Create: LXC Container

General Template Root Disk CPU Memory Network **DNS** Confirm

DNS domain:

DNS servers:

Advanced  Back Next

# MEMBUAT CONTAINER PADA NODE PVE2 (9)

- Tampil kotak dialog **Confirm** dari **LNX Container** yang menampilkan ringkasan pengaturan yang telah dilakukan terkait pembuatan *container CentOS 8*.
- Klik tombol **Finish**.

Create: LXC Container

General Template Root Disk CPU Memory Network DNS Confirm

Key ↑	Value
cores	1
hostname	server113.ubg.local
memory	512
nameserver	192.168.169.254
net0	bridge=vibr2,name=eth0,ip=192.168.11.113/24,gw=192.168.11.254,firewall=1,ta...
nodename	pve2
ostemplate	local:vztmpl/centos-8-default_20191016_amd64.tar.xz
pool	
rootfs	local-lvm:3
searchdomain	ubg.local
swap	512
unprivileged	1
vmid	113

Start after created

Advanced  Back Finish

# MEMBUAT CONTAINER PADA NODE PVE2 (10)

- Tampil kotak dialog **Task viewer: CT 113 – Create**. Tunggu hingga proses pembuatan container *CentOS 8* selesai dibuat dimana ditandai dengan pesan “**TASK OK**” pada bagian **Output** dari kotak dialog **Task viewer: CT 113 – Create**.

```
Task viewer: CT 113 - Create

Output Status

Stop

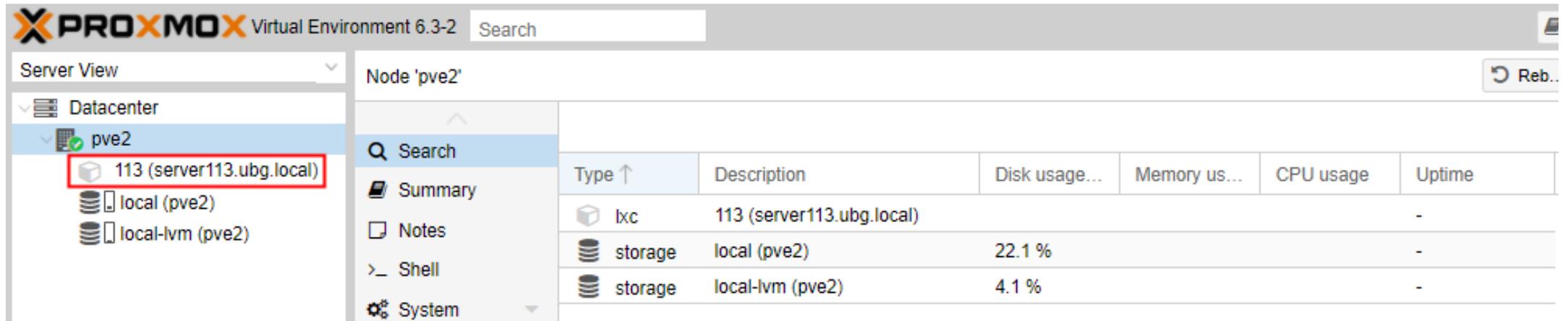
Allocating group tables: 0/24 [progress] done
Writing inode tables: 0/24 [progress] done
Creating journal (16384 blocks): done
Multiple mount protection is enabled with update interval 5 seconds.
Writing superblocks and filesystem accounting information: 0/24 [progress] done

extracting archive '/var/lib/vz/template/cache/centos-8-default_20191016_amd64.tar.xz'
Total bytes read: 673566720 (643MiB, 16MiB/s)
Detected container architecture: amd64
Creating SSH host key 'ssh_host_rsa_key' - this may take some time ...
done: SHA256:d5bdgPZ+O5gfyPsowKu/OxKhiefqavNvTYMApbqWm4 root@server113
Creating SSH host key 'ssh_host_dsa_key' - this may take some time ...
done: SHA256:0iFfXXacmUN2ELnLdYawl7K8WuivBRJt/DUG/K4bXrs root@server113
Creating SSH host key 'ssh_host_ecdsa_key' - this may take some time ...
done: SHA256:uqIG2erMjDEc6DEUGrEVZi9k8qI8N/VTXT/CvAHPF28 root@server113
Creating SSH host key 'ssh_host_ed25519_key' - this may take some time ...
done: SHA256:j8eiyXCziq2iCb+Zaof50pM3WF9uARwM0R/SRzL/DzU root@server113
TASK OK
```

- Tutup kotak dialog **Task viewer: CT 113 – Create**.

# VERIFIKASI HASIL DARI PEMBUATAN CONTAINER PADA NODE PVE2

- Pada panel sebelah kiri dibawah *node pve2* terlihat hasil dari pembuatan *container CentOS 8* dengan ID 113.



The screenshot displays the Proxmox VE interface for node 'pve2'. On the left sidebar, the 'Datacenter' is expanded to show 'pve2', which contains three items: '113 (server113.ubg.local)' (highlighted with a red box), 'local (pve2)', and 'local-lvm (pve2)'. The main panel shows a table of resources for the selected container.

Type ↑	Description	Disk usage...	Memory us...	CPU usage	Uptime
lxc	113 (server113.ubg.local)				-
storage	local (pve2)	22.1 %			-
storage	local-lvm (pve2)	4.1 %			-

## PEMBUATAN 3 CONTAINER (CT) BARU PADA NODE PVE2

Dengan cara yang sama, lakukan pembuatan 3 (tiga) CT baru dengan ketentuan sebagai berikut:

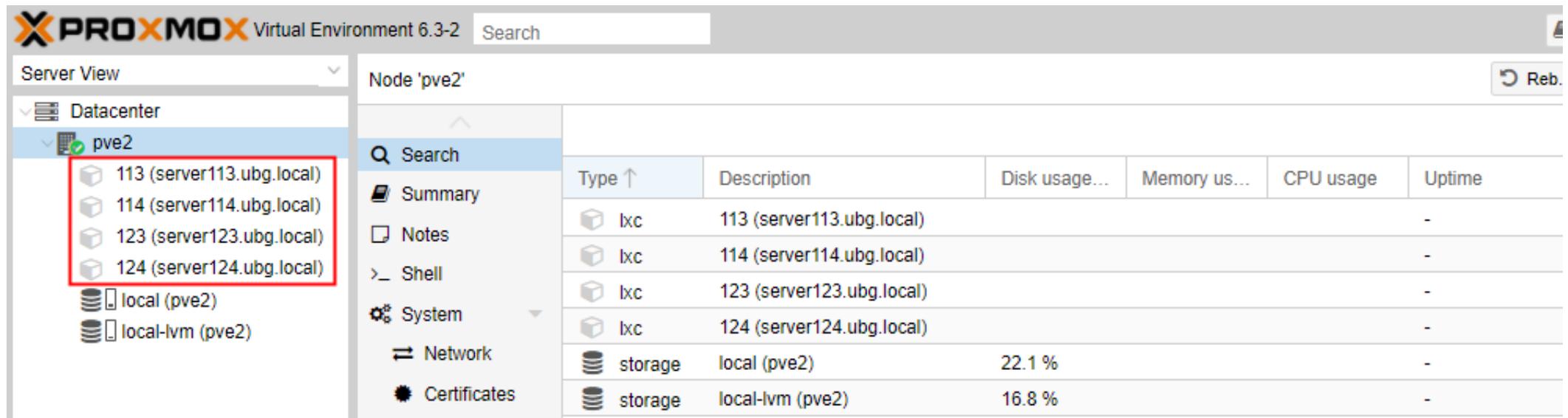
CT ID	Hostname	VLAN ID	IP Address	Gateway
114	server114.ubg.local	11	192.168.11.114/24	192.168.11.253
123	server123.ubg.local	12	192.168.12.123/24	192.168.12.253
124	server124.ubg.local	12	192.168.12.124/24	192.168.12.253

Keseluruhan CT menggunakan nilai yang sama untuk parameter-parameter berikut:

- **Password root** menggunakan nilai “12345678”.
- **Container Template** menggunakan `centos-8-default_20191016_amd64.tar.xz` yang tersimpan di storage local.
- Lokasi penyimpanan *container* adalah di storage `local-lvm` dengan ukuran hardisk sebesar **3 GB**.
- **CPU** menggunakan **1 Core**.
- **Memory** dan swap masing-masing menggunakan **512 MB**.
- **DNS Domain** menggunakan `ubg.local`.
- **DNS Servers** menggunakan `192.168.169.254`.

# HASIL PEMBUATAN KESELURUHAN CONTAINER (CT) DI NODE PVE2

- Terlihat pada *node PVE2* telah terbuat 4 (empat) CT yaitu masing-masing dengan ID 113, 114, 123 dan 124.



PROXMOX Virtual Environment 6.3-2

Server View

Node 'pve2'

Search

Search

Summary

Notes

Shell

System

Network

Certificates

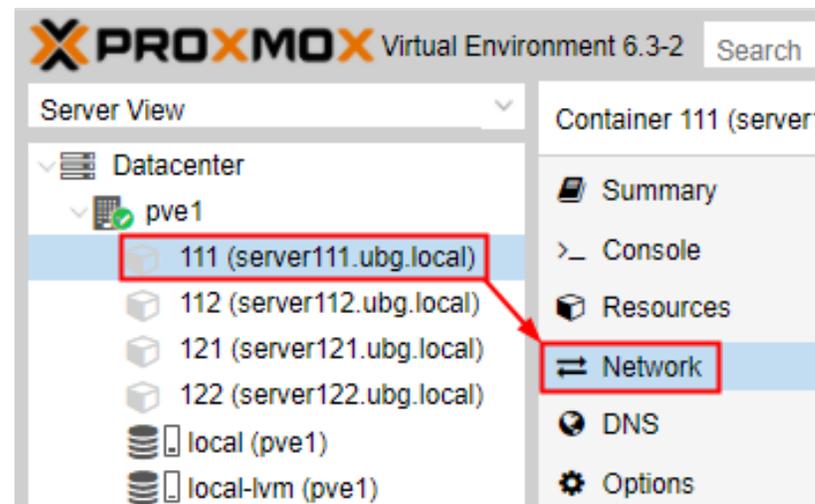
Type ↑	Description	Disk usage...	Memory us...	CPU usage	Uptime
lxc	113 (server113.ubg.local)				-
lxc	114 (server114.ubg.local)				-
lxc	123 (server123.ubg.local)				-
lxc	124 (server124.ubg.local)				-
storage	local (pve2)	22.1 %			-
storage	local-lvm (pve2)	16.8 %			-



# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT)

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 111

- Diasumsikan **CT ID 111** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE1**.
- Navigasi ke menu **Datacenter > pve1** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 111 (server111.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 111

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 111** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 11**.
  - **IP Address 192.168.11.111/24** dengan **Gateway 192.168.11.254**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 111 (server111.ubg.local) on node 'pve1' ▶ St... ⏻ Shutdo...

ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
net0	eth0	vmbr2	Yes	11	5E:80:80:6E:C8:24	192.168.11.111/24	192.168.11.254

- Untuk menjalankan **CT** maka klik kanan pada **111 (server111.ubg.local)** dibawah *node PVE1* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 112

- Diasumsikan **CT ID 112** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE1**.
- Navigasi ke menu **Datacenter > pve1** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 112 (server112.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 112

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 112** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vubr2** dengan **VLAN Tag 11**.
  - **IP Address 192.168.11.112/24** dengan **Gateway 192.168.11.254**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 112 (server112.ubg.local) on node 'pve1' ▶ St... ⏻ Shutdo...

ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
net0	eth0	vubr2	Yes	11	16:FB:6A:5E:BF:44	192.168.11.112/24	192.168.11.254

- Untuk menjalankan **CT** maka klik kanan pada **112 (server112.ubg.local)** dibawah *node PVE1* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 121

- Diasumsikan **CT ID 121** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE1**.
- Navigasi ke menu **Datacenter > pve1** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 121 (server121.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 121

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 121** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 12**.
  - **IP Address 192.168.12.121/24** dengan **Gateway 192.168.12.254**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 121 (server121.ubg.local) on node 'pve1' ▶ St... ⏻ Shutdo...

ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
net0	eth0	vmbr2	Yes	12	BE:FA:66:0B:C1:1B	192.168.12.121/24	192.168.12.254

- Untuk menjalankan **CT** maka klik kanan pada **121 (server121.ubg.local)** dibawah *node PVE1* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 122

- Diasumsikan **CT ID 122** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE1**.
- Navigasi ke menu **Datacenter > pve1** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 122 (server122.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 122

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 122** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 12**.
  - **IP Address 192.168.12.122/24** dengan **Gateway 192.168.12.254**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

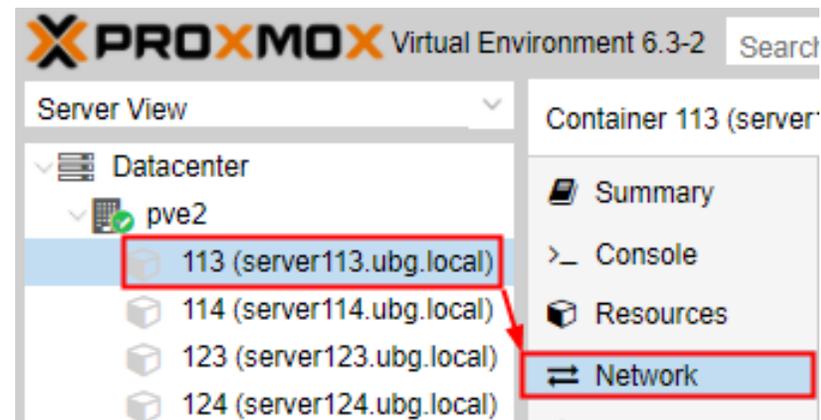
Container 122 (server122.ubg.local) on node 'pve1' ▶ St... ⏻ Shutdo...

ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
net0	eth0	vmbr2	Yes	12	F6:8C:65:BF:84:A2	192.168.12.122/24	192.168.12.254

- Untuk menjalankan **CT** maka klik kanan pada **122 (server122.ubg.local)** dibawah *node PVE1* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 113

- Diasumsikan **CT ID 113** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE2**.
- Navigasi ke menu **Datacenter > pve2** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 113 (server113.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 113

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 113** yang terdapat di *node PVE2* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 11**.
  - **IP Address 192.168.11.113/24** dengan **Gateway 192.168.11.253**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 113 (server113.ubg.local) on node 'pve2'

Summary	Add	Remove	Edit					
Console	ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
Resources	net0	eth0	vmbr2	Yes	11	A6:F5:73:64:...	192.168.11.113/24	192.168.11.253
Network								

- Untuk menjalankan **CT** maka klik kanan pada **113 (server113.ubg.local)** dibawah *node PVE2* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 114

- Diasumsikan **CT ID 114** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE2**.
- Navigasi ke menu **Datacenter > pve2** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 114 (server114.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 114

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 114** yang terdapat di *node PVE2* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 11**.
  - **IP Address 192.168.11.114/24** dengan **Gateway 192.168.11.253**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

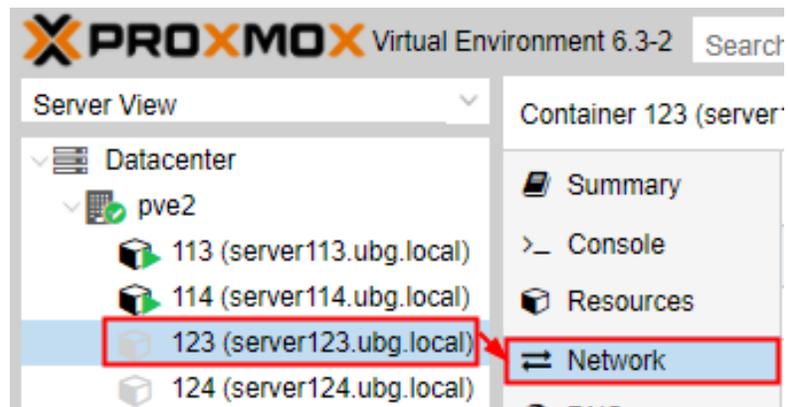
Container 114 (server114.ubg.local) on node 'pve2'

Summary	Add	Remove	Edit					
Console	ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
Resources	net0	eth0	vmbr2	Yes	11	42:59:E7:A0:...	192.168.11.114/24	192.168.11.253
Network								

- Untuk menjalankan **CT** maka klik kanan pada **114 (server114.ubg.local)** dibawah *node PVE2* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 123

- Diasumsikan **CT ID 123** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE2**.
- Navigasi ke menu **Datacenter > pve2** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 123 (server123.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 123

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 123** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 12**.
  - **IP Address 192.168.12.123/24** dengan **Gateway 192.168.12.253**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 123 (server123.ubg.local) on node 'pve2'

Summary	Add	Remove	Edit					
Console	ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway
Resources	net0	eth0	vmbr2	Yes	12	5E:29:D9:42:...	192.168.12.123/24	192.168.12.253
Network								

- Untuk menjalankan **CT** maka klik kanan pada **123 (server123.ubg.local)** dibawah *node PVE2* dari menu **Datacenter** dan pilih **Start**.

# VERIFIKASI KONFIGURASI NETWORK PADA CONTAINER (CT) ID 124

- Diasumsikan **CT ID 124** dengan *container template* **centos-8-default\_20191016\_amd64.tar.xz** telah terbuat pada *node* **PVE2**.
- Navigasi ke menu **Datacenter > pve2** pada panel sebelah kiri dari halaman **Server View** dan memilih **CT ID 124 (server124.ubg.local)**.
- Pada panel detail sebelah kanan pilih **Network**.



- Pilih *interface* **net0** dan klik tombol **Edit** untuk melakukan penyesuaian.

# KONFIGURASI NETWORK PADA CONTAINER (CT) ID 124

- Lakukan verifikasi konfigurasi **network** pada **interface net0** dari **CT ID 124** yang terdapat di *node PVE1* dan pastikan agar menggunakan nilai berikut:
  - **Bridge vmbr2** dengan **VLAN Tag 12**.
  - **IP Address 192.168.12.124/24** dengan **Gateway 192.168.12.253**.
- Hasil akhir dari pengaturan **network** tersebut adalah sebagai berikut:

Container 124 (server124.ubg.local) on node 'pve2'

		Add	Remove	Edit				
ID.↑	Name	Bridge	Firewall	VLAN Tag	MAC address	IP address	Gateway	
net0	eth0	vmbr2	Yes	12	BA:53:D3:A9...	192.168.12.124/24	192.168.12.253	

- Untuk menjalankan **CT** maka klik kanan pada **124 (server124.ubg.local)** dibawah *node PVE2* dari menu **Datacenter** dan pilih **Start**.



# VERIFIKASI KONEKSI ANTAR CONTAINER (CT)

# VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 11 (1)

- Akses **Console** dari **CT111** dan **login** menggunakan *user* “**root**” serta sandi “**12345678**”.
- Verifikasi koneksi dari **CT111** ke **CT112** menggunakan perintah **ping 192.168.11.112**. Kedua CT tersebut terdapat pada *node* **PVE1** dan menjadi anggota dari **VLAN ID 11**.

```
CentOS Linux 8 (Core)
Kernel 5.4.73-1-pve on an x86_64

server111 login: root
Password:
Last login: Fri Dec 25 04:59:19 on tty1
[root@server111 ~]# ping 192.168.11.112
PING 192.168.11.112 (192.168.11.112) 56(84) bytes of data.
64 bytes from 192.168.11.112: icmp_seq=1 ttl=64 time=5.32 ms

--- 192.168.11.112 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 5.319/5.319/5.319/0.000 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

## VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 11 (2)

- Verifikasi koneksi dari **CT111** yang terdapat pada *node PVE1* ke **CT113** yang terdapat di *node PVE2* menggunakan perintah **ping 192.168.11.113**. Kedua CT tersebut menjadi anggota dari **VLAN ID 11**.

```
[root@server111 ~]# ping 192.168.11.113
PING 192.168.11.113 (192.168.11.113) 56(84) bytes of data.
64 bytes from 192.168.11.113: icmp_seq=1 ttl=64 time=3.40 ms
64 bytes from 192.168.11.113: icmp_seq=2 ttl=64 time=0.749 ms

--- 192.168.11.113 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 2ms
rtt min/avg/max/mdev = 0.749/2.072/3.395/1.323 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

## VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 11 (3)

- Verifikasi koneksi dari **CT111** yang terdapat pada *node PVE1* ke **CT114** yang terdapat di *node PVE2* menggunakan perintah **ping 192.168.11.114**. Kedua CT tersebut menjadi anggota dari **VLAN ID 11**.

```
[root@server111 ~]# ping 192.168.11.114
PING 192.168.11.114 (192.168.11.114) 56(84) bytes of data.
64 bytes from 192.168.11.114: icmp_seq=1 ttl=64 time=3.59 ms
64 bytes from 192.168.11.114: icmp_seq=2 ttl=64 time=1.49 ms

--- 192.168.11.114 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 1.494/2.541/3.589/1.048 ms
[root@server111 ~]#
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# VERIFIKASI KONEKSI KE DEFAULT GATEWAY DARI VLAN ID 11

- Verifikasi koneksi dari **CT111** ke **default gateway** dari **VLAN ID 11** yang terdapat pada *node PVE1* menggunakan perintah **ping 192.168.11.254**.

```
[root@server111 ~]# ping 192.168.11.254
PING 192.168.11.254 (192.168.11.254) 56(84) bytes of data.
64 bytes from 192.168.11.254: icmp_seq=1 ttl=64 time=0.974 ms
64 bytes from 192.168.11.254: icmp_seq=2 ttl=64 time=0.108 ms

--- 192.168.11.254 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 0.108/0.541/0.974/0.433 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# VERIFIKASI KONEKSI KE INTERNET DARI CONTAINER PADA VLAN ID 11

- Verifikasi koneksi dari **CT111** ke salah satu situs yang ada di Internet, sebagai contoh ke **Google.com** dengan mengeksekusi perintah **ping google.com**.

```
[root@server111 ~]# ping google.com
PING forcesafesearch.google.com (216.239.38.120) 56(84) bytes of data.
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=1 ttl=126 time=69.9 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=2 ttl=126 time=86.5 ms

--- forcesafesearch.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 5ms
rtt min/avg/max/mdev = 69.857/78.168/86.479/8.311 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.
- Apabila koneksi ke salah satu situs di *Internet* gagal dilakukan maka lakukan pengaktifan fitur **IP Forwarding** di file **/etc/sysctl.conf**.

# MENGAKTIFKAN FITUR IP FORWARDING DI NODE PVE1

- Membuka *file* `/etc/sysctl.conf` menggunakan *editor* `nano`.

```
# nano /etc/sysctl.conf
```

- Menghilangkan tanda `#` di awal dari parameter yang terdapat pada **baris 28**.

```
26  
27 # Uncomment the next line to enable packet forwarding for IPv4  
28 #net.ipv4.ip_forward=1
```

- Simpan perubahan dengan menekan tombol **CTRL+O** dan tekan **Enter**. Tekan tombol **CTRL+X** untuk keluar dari *editor* `nano`.

- Mengaktifkan perubahan pada *file* `/etc/sysctl.conf` dengan mengeksekusi perintah:

```
# sysctl -p
```

- Hasilnya terlihat seperti berikut:

```
root@pve1:~# sysctl -p  
net.ipv4.ip_forward = 1
```

- Selanjutnya lakukan verifikasi koneksi kembali ke salah satu situs di *Internet* pada salah satu CT yang terdapat di *node* `pve1`.

# VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 12 (1)

- Akses **Console** dari **CT121** dan **login** menggunakan *user* “**root**” serta sandi “**12345678**”.
- Verifikasi koneksi dari **CT121** ke **CT122** menggunakan perintah **ping 192.168.12.122**. Kedua CT tersebut terdapat pada *node* **PVE1** dan menjadi anggota dari **VLAN ID 12**.

```
CentOS Linux 8 (Core)
Kernel 5.4.73-1-pve on an x86_64

server121 login: root
Password:
Last login: Fri Dec 25 04:58:51 on tty1
[root@server121 ~]# ping 192.168.12.122
PING 192.168.12.122 (192.168.12.122) 56(84) bytes of data.
 64 bytes from 192.168.12.122: icmp_seq=1 ttl=64 time=1.72 ms
 64 bytes from 192.168.12.122: icmp_seq=2 ttl=64 time=0.163 ms

--- 192.168.12.122 ping statistics ---
 2 packets transmitted, 2 received, 0% packet loss, time 4ms
 rtt min/avg/max/mdev = 0.163/0.942/1.722/0.780 ms
[root@server121 ~]#
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

## VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 12 (2)

- Verifikasi koneksi dari **CT121** yang terdapat pada *node PVE1* ke **CT123** yang terdapat di *node PVE2* menggunakan perintah **ping 192.168.11.123**. Kedua CT tersebut menjadi anggota dari **VLAN ID 12**.

```
[root@server121 ~]# ping 192.168.12.123
PING 192.168.12.123 (192.168.12.123) 56(84) bytes of data.
64 bytes from 192.168.12.123: icmp_seq=1 ttl=64 time=3.40 ms
64 bytes from 192.168.12.123: icmp_seq=2 ttl=64 time=1.39 ms

--- 192.168.12.123 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 1.388/2.392/3.397/1.005 ms
[root@server121 ~]#
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

## VERIFIKASI KONEKSI ANTAR CONTAINER PADA VLAN ID 12 (3)

- Verifikasi koneksi dari **CT121** yang terdapat pada *node PVE1* ke **CT124** yang terdapat di *node PVE2* menggunakan perintah **ping 192.168.11.124**. Kedua CT tersebut menjadi anggota dari **VLAN ID 12**.

```
[root@server121 ~]# ping 192.168.12.124
PING 192.168.12.124 (192.168.12.124) 56(84) bytes of data.
64 bytes from 192.168.12.124: icmp_seq=1 ttl=64 time=3.34 ms
64 bytes from 192.168.12.124: icmp_seq=2 ttl=64 time=1.24 ms

--- 192.168.12.124 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 1.242/2.290/3.338/1.048 ms
[root@server121 ~]#
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# VERIFIKASI KONEKSI KE DEFAULT GATEWAY DARI VLAN ID 12

- Verifikasi koneksi dari **CT121** ke **default gateway** dari **VLAN ID 12** yang terdapat pada *node* **PVE2** menggunakan perintah **ping 192.168.12.254**.

```
[root@server121 ~]# ping 192.168.12.254
PING 192.168.12.254 (192.168.12.254) 56(84) bytes of data.
64 bytes from 192.168.12.254: icmp_seq=1 ttl=64 time=0.387 ms
64 bytes from 192.168.12.254: icmp_seq=2 ttl=64 time=0.156 ms

--- 192.168.12.254 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 0.156/0.271/0.387/0.116 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# VERIFIKASI KONEKSI KE INTERNET DARI CONTAINER PADA VLAN ID 12

- Verifikasi koneksi dari **CT121** ke salah satu situs yang ada di *Internet*, sebagai contoh ke **Google.com** dengan mengeksekusi perintah **ping google.com**.

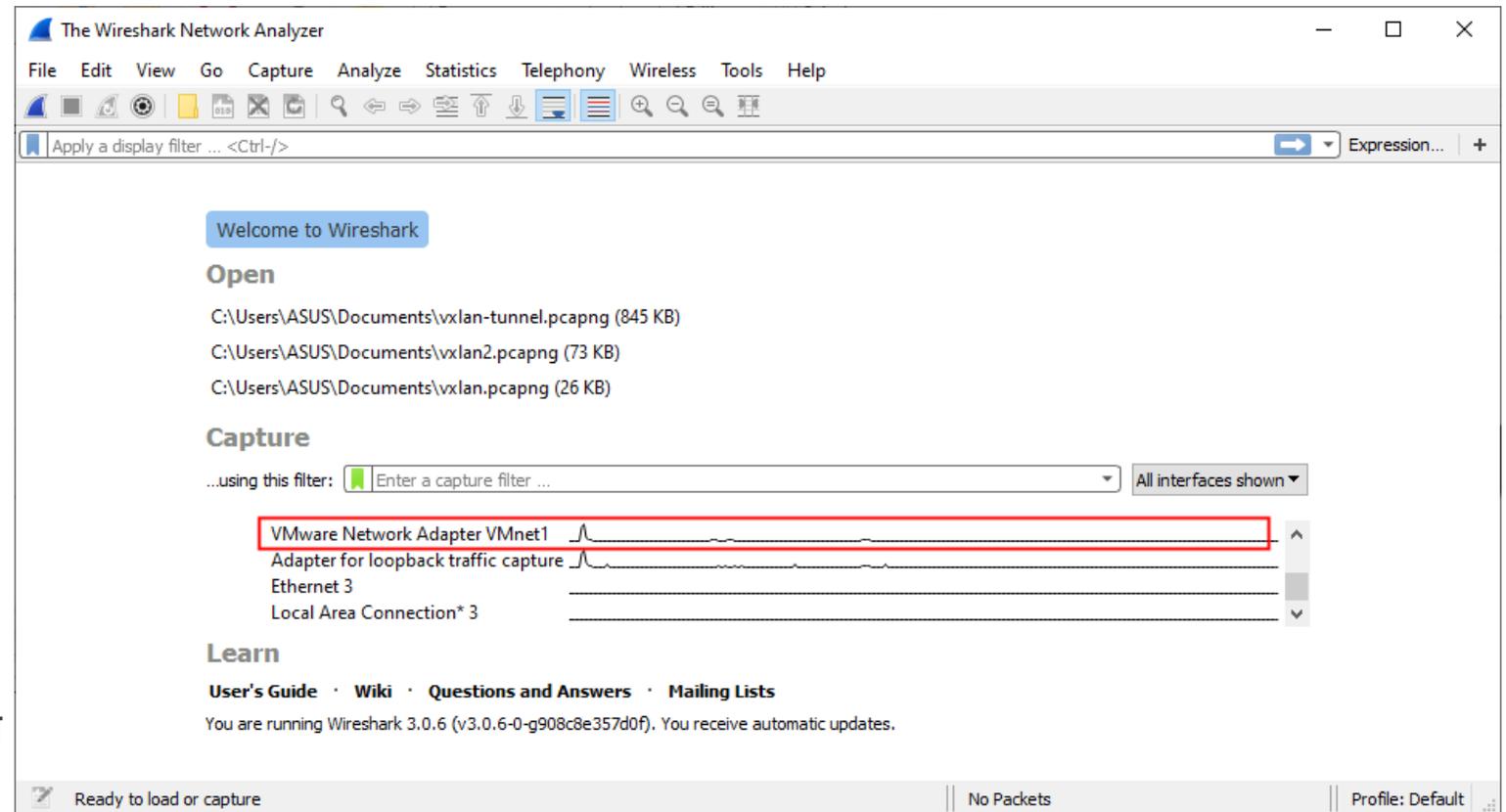
```
[root@server121 ~]# ping google.com
PING forcesafesearch.google.com (216.239.38.120) 56(84) bytes of data.
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=1 ttl=126 time=74.9 ms
64 bytes from any-in-2678.1e100.net (216.239.38.120): icmp_seq=2 ttl=126 time=73.0 ms

--- forcesafesearch.google.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 3ms
rtt min/avg/max/mdev = 73.022/73.955/74.888/0.933 ms
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# ANALISA VXLAN PACKET MENGGUNAKAN WIRESHARK (1)

- Wireshark merupakan aplikasi yang dapat digunakan untuk menganalisa protokol jaringan.
- Aplikasi **Wireshark** dapat diunduh di [www.wireshark.org](http://www.wireshark.org).
- Jalankan aplikasi **Wireshark** yang telah terinstalasi pada komputer yang digunakan.
- Pada kotak dialog **The Wireshark Network Analyzer** yang tampil, pilih interface **VMware Network Adapter VMnet1** pada bagian **Capture** untuk melakukan sniffing **VXLAN packet** pada *interface* tersebut.



# ANALISA VXLAN PACKET MENGGUNAKAN WIRESHARK (1)

- Lakukan verifikasi kembali koneksi dari **Console CT111** yang terdapat pada *node PVE1* ke **CT114** yang terdapat di *node PVE2* menggunakan perintah **ping 192.168.11.114**. Kedua CT tersebut menjadi anggota dari **VLAN ID 11**.

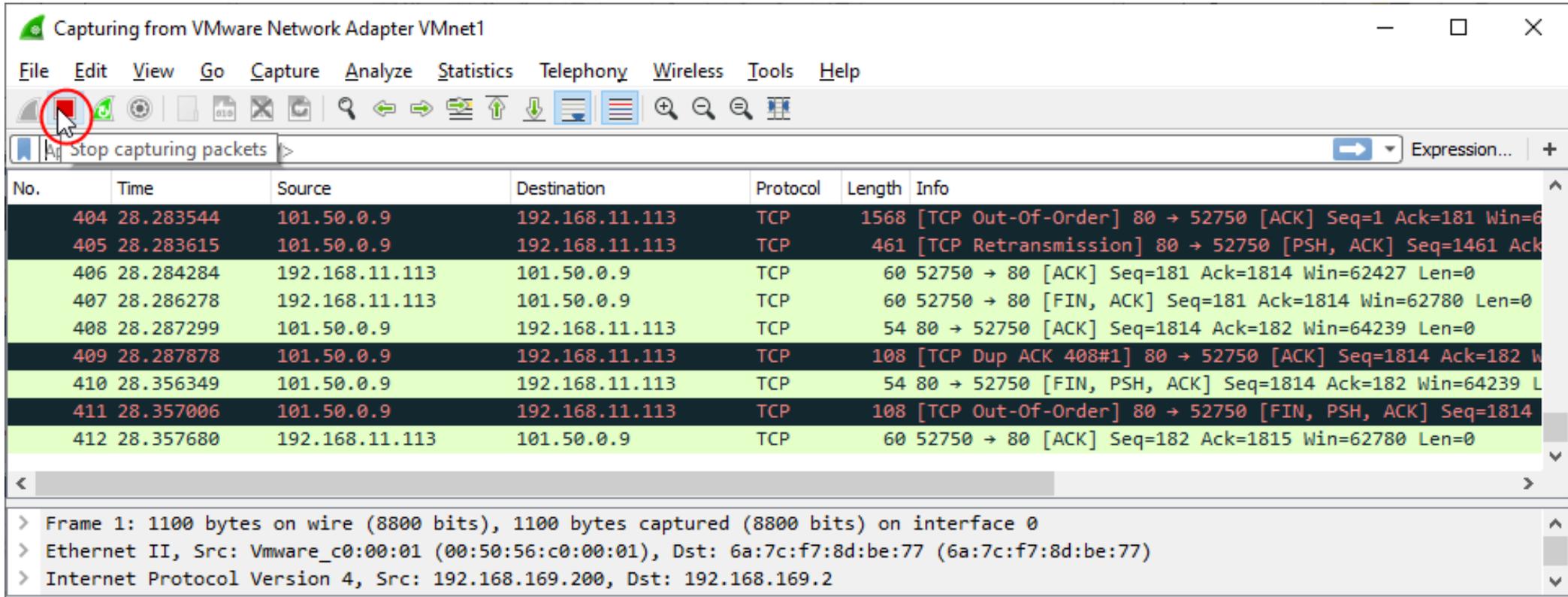
```
[root@server111 ~]# ping 192.168.11.114
PING 192.168.11.114 (192.168.11.114) 56(84) bytes of data.
64 bytes from 192.168.11.114: icmp_seq=1 ttl=64 time=3.59 ms
64 bytes from 192.168.11.114: icmp_seq=2 ttl=64 time=1.49 ms

--- 192.168.11.114 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 4ms
rtt min/avg/max/mdev = 1.494/2.541/3.589/1.048 ms
[root@server111 ~]#
```

- Terlihat koneksi berhasil dilakukan. Tekan **CTRL+C** untuk menghentikan ping.

# MENGHENTIKAN CAPTURING PACKETS PADA WIRESHARK

- Pada *toolbar* dari kotak dialog aplikasi **Wireshark**, klik tombol  untuk menghentikan (**stop**) **capturing packets**.



Capturing from VMware Network Adapter VMnet1

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

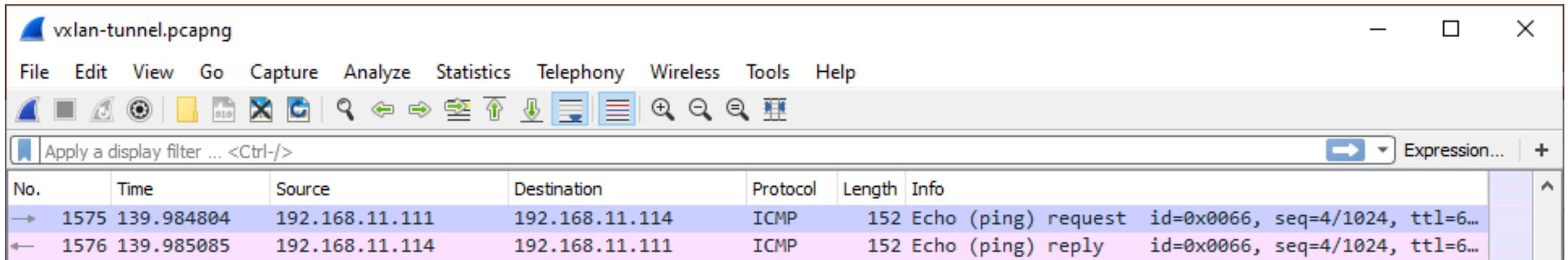
Stop capturing packets

No.	Time	Source	Destination	Protocol	Length	Info
404	28.283544	101.50.0.9	192.168.11.113	TCP	1568	[TCP Out-Of-Order] 80 → 52750 [ACK] Seq=1 Ack=181 Win=6
405	28.283615	101.50.0.9	192.168.11.113	TCP	461	[TCP Retransmission] 80 → 52750 [PSH, ACK] Seq=1461 Ack
406	28.284284	192.168.11.113	101.50.0.9	TCP	60	52750 → 80 [ACK] Seq=181 Ack=1814 Win=62427 Len=0
407	28.286278	192.168.11.113	101.50.0.9	TCP	60	52750 → 80 [FIN, ACK] Seq=181 Ack=1814 Win=62780 Len=0
408	28.287299	101.50.0.9	192.168.11.113	TCP	54	80 → 52750 [ACK] Seq=1814 Ack=182 Win=64239 Len=0
409	28.287878	101.50.0.9	192.168.11.113	TCP	108	[TCP Dup ACK 408#1] 80 → 52750 [ACK] Seq=1814 Ack=182 W
410	28.356349	101.50.0.9	192.168.11.113	TCP	54	80 → 52750 [FIN, PSH, ACK] Seq=1814 Ack=182 Win=64239 L
411	28.357006	101.50.0.9	192.168.11.113	TCP	108	[TCP Out-Of-Order] 80 → 52750 [FIN, PSH, ACK] Seq=1814
412	28.357680	192.168.11.113	101.50.0.9	TCP	60	52750 → 80 [ACK] Seq=182 Ack=1815 Win=62780 Len=0

> Frame 1: 1100 bytes on wire (8800 bits), 1100 bytes captured (8800 bits) on interface 0  
> Ethernet II, Src: Vmware\_c0:00:01 (00:50:56:c0:00:01), Dst: 6a:7c:f7:8d:be:77 (6a:7c:f7:8d:be:77)  
> Internet Protocol Version 4, Src: 192.168.169.200, Dst: 192.168.169.2

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (1)

- Cuplikan hasil **capture** pada *interface* **VMware Network Adapter VMNet1** menggunakan **Wireshark** ketika **CT111** dengan alamat IP sumber **192.168.11.111** melakukan **ping** ke **CT114** dengan alamat IP tujuan **192.168.11.114**.



The screenshot shows the Wireshark interface with a capture of VXLAN frames. The packet list pane displays two ICMP frames:

No.	Time	Source	Destination	Protocol	Length	Info
→ 1575	139.984804	192.168.11.111	192.168.11.114	ICMP	152	Echo (ping) request id=0x0066, seq=4/1024, ttl=6...
← 1576	139.985085	192.168.11.114	192.168.11.111	ICMP	152	Echo (ping) reply id=0x0066, seq=4/1024, ttl=6...

- Untuk menampilkan informasi detail dari *frame* maka klik dua kali pada salah satu dari hasil *frame capture*, sebagai contoh nomor **1575**.

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (2)

- Terlampir cuplikan penampilan *frame* yang di *capture* secara lebih detail.
- VXLAN menggunakan enkapsulasi dan tunneling MAC di dalam UDP untuk memperluas segmen layer 2 melalui jaringan di layer 3.
- Frame ethernet layer 2 yang dikirim oleh CT111 ke CT114 ditempatkan di dalam paket UDP-IP dan dikirim melalui VXLAN tunnel.

```
Wireshark · Packet 1575 · VMware Network Adapter VMnet1

> Frame 1575: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
> Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e) I
> Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2 H
> User Datagram Protocol, Src Port: 39245, Dst Port: 4789 G
  Source Port: 39245
  Destination Port: 4789
  Length: 118
  [Checksum: [missing]]
  [Checksum Status: Not present]
  [Stream index: 11]
  > [Timestamps]
> Virtual eXtensible Local Area Network F
  > Flags: 0x0800, VXLAN Network ID (VNI)
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 100 E
  Reserved: 0
> Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b) D
  > Destination: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
  > Source: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24)
  Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11 C
> Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114 B
> Internet Control Message Protocol
  Type: 8 (Echo (ping) request) A
```

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (3)

- Pada bagian **A** dan **B** memperlihatkan informasi terkait jenis paket **Internet Control Message Protocol (ICMP)** yang dikirim oleh **CT111** dengan IP sumber **192.168.11.111** ke **CT114** dengan IP tujuan **192.168.11.114**. adalah **Echo request (8)**.
- Pada bagian **C** memperlihatkan **VLAN ID 11** yang digunakan oleh **CT111**.

```
Wireshark · Packet 1575 · VMware Network Adapter VMnet1
> Frame 1575: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
> Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e) I
> Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2 H
> User Datagram Protocol, Src Port: 39245, Dst Port: 4789 G
  Source Port: 39245
  Destination Port: 4789
  Length: 118
  [Checksum: [missing]]
  [Checksum Status: Not present]
  [Stream index: 11]
  > [Timestamps]
> Virtual eXtensible Local Area Network F
  > Flags: 0x0800, VXLAN Network ID (VNI)
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 100 E
  Reserved: 0
> Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b) D
  > Destination: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
  > Source: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24)
  Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11 C
> Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114 B
> Internet Control Message Protocol
  Type: 8 (Echo (ping) request) A
```

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (4)

- Pada bagian **D** memperlihatkan frame **Ethernet II** yang dikirim oleh **CT111** dengan alamat **Media Access Control (MAC)** sumber **5E:80:80:6E:C8:24** ke **CT114** dengan alamat MAC tujuan **42:59:E7:A0:90:2B**.
- Pada bagian **E** dan **F** memperlihatkan informasi terkait **VXLAN Network Identifier (VNI)** dengan nilai **100** yang berkaitan dengan trafik yang bersumber dari **CT111** dengan tujuan ke **CT114**.

```
Wireshark · Packet 1575 · VMware Network Adapter VMnet1

> Frame 1575: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
> Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e) I
> Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2 H
> User Datagram Protocol, Src Port: 39245, Dst Port: 4789 G
  Source Port: 39245
  Destination Port: 4789
  Length: 118
  [Checksum: [missing]]
  [Checksum Status: Not present]
  [Stream index: 11]
  [Timestamps]
> Virtual eXtensible Local Area Network F
  Flags: 0x0800, VXLAN Network ID (VNI)
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 100 E
  Reserved: 0
> Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b) D
  Destination: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
  Source: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24)
  Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11 C
> Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114 B
> Internet Control Message Protocol
  Type: 8 (Echo (ping) request) A
```

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (5)

- Pada bagian **G** memperlihatkan **UDP header** dengan port sumber **39245** dan port tujuan **4789** yang merupakan port dari **VXLAN**.
- Pada bagian **H** memperlihatkan **IP header** dengan alamat IP sumber **192.168.170.1** dan alamat IP tujuan **192.168.170.2** **VXLAN Tunnel**.

```
Wireshark · Packet 1575 · VMware Network Adapter VMnet1

> Frame 1575: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
> Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e) I
> Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2 H
▼ User Datagram Protocol, Src Port: 39245, Dst Port: 4789 G
  Source Port: 39245
  Destination Port: 4789
  Length: 118
  [Checksum: [missing]]
  [Checksum Status: Not present]
  [Stream index: 11]
  > [Timestamps]
▼ Virtual eXtensible Local Area Network F
  > Flags: 0x0800, VXLAN Network ID (VNI)
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 100 E
  Reserved: 0
▼ Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b) D
  > Destination: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
  > Source: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24)
  Type: 802.1Q Virtual LAN (0x8100)
> 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11 C
> Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114 B
▼ Internet Control Message Protocol
  Type: 8 (Echo (ping) request) A
```

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (6)

- Pada bagian **I** memperlihatkan **frame Ethernet II** yang dikirim oleh **VTEP** pada **PVE1** dengan alamat **Media Access Control (MAC)** sumber **ba:95:26:a5:a2:72** ke **VTEP** dari **PVE2** dengan alamat MAC tujuan **be:49:ec:11:e0:7e**.

```
Wireshark · Packet 1575 · VMware Network Adapter VMnet1
> Frame 1575: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
> Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e) I
> Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2 H
▼ User Datagram Protocol, Src Port: 39245, Dst Port: 4789 G
  Source Port: 39245
  Destination Port: 4789
  Length: 118
  [Checksum: [missing]]
  [Checksum Status: Not present]
  [Stream index: 11]
  > [Timestamps]
▼ Virtual eXtensible Local Area Network F
  > Flags: 0x0800, VXLAN Network ID (VNI)
  Group Policy ID: 0
  VXLAN Network Identifier (VNI): 100 E
  Reserved: 0
▼ Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b) D
  > Destination: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
  > Source: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24)
  Type: 802.1Q Virtual LAN (0x8100)
  > 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11 C
  > Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114 B
▼ Internet Control Message Protocol
  Type: 8 (Echo (ping) request) A
```

# ANALISA VXLAN FRAME ENCAPSULATION DARI WIRESHARK (7)

Wireshark · Packet 1557 · vxlan-tunnel.pcapng

- > Frame 1557: 152 bytes on wire (1216 bits), 152 bytes captured (1216 bits) on interface 0
- ▼ Ethernet II, Src: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72), Dst: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e)
  - > Destination: be:49:ec:11:e0:7e (be:49:ec:11:e0:7e)
  - > Source: ba:95:26:a5:a2:72 (ba:95:26:a5:a2:72)
  - Type: IPv4 (0x0800)
- ▼ Internet Protocol Version 4, Src: 192.168.170.1, Dst: 192.168.170.2
  - 0100 .... = Version: 4
  - .... 0101 = Header Length: 20 bytes (5)
  - > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  - Total Length: 138
  - Identification: 0xb2e2 (45794)
  - > Flags: 0x4000, Don't fragment
  - ...0 0000 0000 0000 = Fragment offset: 0
  - Time to live: 64
  - Protocol: UDP (17)
  - Header checksum: 0xb22b [validation disabled]
  - [Header checksum status: Unverified]
  - Source: 192.168.170.1
  - Destination: 192.168.170.2
- ▼ User Datagram Protocol, Src Port: 39245, Dst Port: 4789
  - Source Port: 39245
  - Destination Port: 4789
  - Length: 118
  - [Checksum: [missing]]
  - [Checksum Status: Not present]
  - [Stream index: 11]
  - > [Timestamps]
- > Virtual eXtensible Local Area Network
- > Ethernet II, Src: 5e:80:80:6e:c8:24 (5e:80:80:6e:c8:24), Dst: 42:59:e7:a0:90:2b (42:59:e7:a0:90:2b)
- > 802.1Q Virtual LAN, PRI: 0, DEI: 0, ID: 11
- > Internet Protocol Version 4, Src: 192.168.11.111, Dst: 192.168.11.114
- > Internet Control Message Protocol

**UNDERLAY**

**OVERLAY**



**ADA PERTANYAAN?**



# TERIMAKASIH