

? Making MTU 9000 Persistent on XCP-ng (OVS)

? Guide Overview

Title: Making MTU 9000 Persistent on XCP-ng (OVS)

Author: MSLS Partners LLC

Last Updated: April 2025

Description: This guide explains how to configure MTU 9000 for Open vSwitch (OVS) inside XCP-ng environments. It ensures that jumbo frames remain active and survive reboots, greatly improving network performance for storage and virtual machines.

Difficulty Level: Intermediate ☐☐

Estimated Time: 15-20 minutes

? Table of Contents

- [Purpose](#)
- [Requirements](#)
- [Background](#)
- [Steps](#)
 - [☐☐ Step 1: Create the MTU Fix Bash Script](#)
 - [☐☐ Step 2: Create the systemd Service](#)
 - [☐☐ Step 3: Enable the Service](#)
 - [☐☐ Step 4: Validate](#)
- [Expected Results](#)
- [Troubleshooting](#)

- [Bonus: Quick MTU Validation Script](#)
- [Final Notes](#)

? Purpose

This guide explains how to correctly configure **MTU 9000** on **Open vSwitch (OVS)** inside **XCP-ng** to ensure jumbo frames survive across reboots.

? Requirements

- XCP-ng 8.x or newer
- Open vSwitch in use (default in XCP-ng)
- SSH access to the hypervisor
- Root privileges

? Background

- XCP-ng uses Open vSwitch (OVS) to manage VM networking.
- Setting MTU at the Linux NIC level is **not enough**; OVS controls the real MTU behavior.
- XCP-ng resets network bridges at boot, so MTU settings must be **reapplied automatically**.

? Steps

? Step 1: Create the MTU Fix Bash Script

```
sudo nano /usr/local/bin/fix-ovs-mtu.sh
```

Paste this:

```
#!/bin/bash
sleep 20
ovs-vsctl set interface eth4 mtu_request=9000
ovs-vsctl set interface xenbr4 mtu_request=9000
echo "$(date) - MTU 9000 applied to eth4 and xenbr4" >> /var/log/fix-ovs-mtu.log
```

Make it executable:

```
sudo chmod +x /usr/local/bin/fix-ovs-mtu.sh
```

? Step 2: Create the systemd Service

```
sudo nano /etc/systemd/system/fix-ovs-mtu.service
```

Paste this:

```
[Unit]
Description=Fix OVS Interfaces MTU to 9000 After Boot
After=network-online.target openvswitch-switch.service

[Service]
ExecStart=/usr/local/bin/fix-ovs-mtu.sh
Type=oneshot
RemainAfterExit=yes

[Install]
WantedBy=multi-user.target
```

? Step 3: Enable the Service

```
sudo systemctl daemon-reload
sudo systemctl enable fix-ovs-mtu.service
sudo systemctl start fix-ovs-mtu.service
sudo systemctl status fix-ovs-mtu.service
```

? Step 4: Validate

- Check service status:

```
sudo systemctl status fix-ovs-mtu.service
```

- Check MTU settings:

```
ip link show eth4
ip link show xenbr4
```

- Verify inside OVS:

```
ovs-vsctl list interface eth4 | grep mtu  
ovs-vsctl list interface xenbr4 | grep mtu
```

? Expected Results

- eth4 MTU = 9000
- xenbr4 MTU = 9000
- VMs' VIFs inherit MTU 9000
- Jumbo frames supported end-to-end
- Performance improvement (storage, VM transfers)

? Troubleshooting

Symptom	Likely Cause	Solution
MTU resets to 1500 after reboot	Systemd service not enabled	Check service status
Ping fragmentation on large packets	Switch ports not configured for MTU 9000	Enable jumbo frames on switch ports
iperf3 speed low	MTU not properly set at all layers	Double-check NIC, bridge, VM, and switch

? Bonus: Quick MTU Validation Script

```
#!/bin/bash  
echo "Checking eth4:"  
ip link show eth4 | grep mtu  
echo "Checking xenbr4:"  
ip link show xenbr4 | grep mtu  
echo "Checking VMs VIFs:"  
ip link | grep vif | grep mtu
```

? Final Notes

- Always verify MTU from storage to VM.
- Setting `mtu_request` inside OVS and a systemd fix ensures full persistence.

- Better performance, lower CPU usage for 10GbE network operations.
-

? Quick Steps Checklist

- Create the MTU fix script at `/usr/local/bin/fix-ovs-mtu.sh`
- Make the script executable
- Create the systemd service at `/etc/systemd/system/fix-ovs-mtu.service`
- Reload systemd
- Enable and start the service
- Verify MTU settings on `eth4` and `xenbr4` after boot
- Validate VMs inherit MTU 9000
- Test with `iperf3` and `ping -M do -s 8972`

Created and maintained by **MSLS Partners LLC** 

Empowering IT with clear documentation and professional best practices.

Version 1.0 | Last Reviewed: April 2025

Revision #4

Created 29 April 2025 05:10:43 by joliveira

Updated 29 April 2025 13:21:21 by joliveira