

XINNOR

What's New in xiRAID Opus 1.3.0

Contents

xiRAID Opus 1.3.0 Release Notes.....	3
---	----------

xiRAID Opus 1.3.0 Release Notes

NVMe-oF Host Access Control

We've introduced the ability to control which hosts (initiators) can access the subsystem exposed over NVMe-oF.

Persistent Configuration for BDEVs

This feature ensures that vhost configurations and namespaces are restored when configured on top of a local or network disk (or its partition) without RAID. It also supports recovery if the disk is removed and reinserted.

Improvements to xnr_cli

All CLI show commands have been upgraded to provide output in a table format.

SPDK Updated to 25.05

xiRAID Opus has been upgraded to SPDK version 25.05.

Known Issues

1. Vhost Device IO Recovery after RAID or Disk Failure

If a RAID or single disk hosting a vhost device becomes unavailable and later becomes available, you must restart the virtual machine or manually restart the device within the virtual machine. Without this step, IO will not reconnect properly, and the vhost will appear in the `detached` state in the CLI.

2. IOMMU Group Issues on Kernels < 5.19

There is a known bug in Linux kernel versions earlier than 5.19 that can cause user session termination and error messages to appear in `dmesg` or `journalctl`. The issue occurs when multiple devices are present in a single IOMMU group, and only some of them are attached to xiRAID Opus. If a device from the group that is not attached to SPDK is removed from the system and then reinserted, it can trigger session termination. After this point, any subsequent attempt to rescan the PCI bus or remove a device will cause the current process to hang.

3. Issues with Connected Network Devices

In a setup where xiRAID Opus connects to NVMe-oF network drives backed by VirtIO drives (on a virtual machine on Proxmox with Ubuntu 22.04 LTS, where disks are exposed using `nvmef`), detaching a drive from the sharing VM does not immediately mark the associated namespace as offline. Instead, the namespace remains online in xiRAID Opus until I/O operations are issued on the RAID, after which the detached namespace is correctly recognized as offline. Additionally, when the detached drive is returned, the namespace remains offline until `nvmef disable` and `nvmef enable` commands are executed. IO operations cannot be performed on the namespace until this step is completed.

4. Insufficient I/O Queues on NVMe Drives May Prevent Opus Namespace Configuration

Some NVMe drives may expose fewer I/O queues than expected, potentially causing issues when running xiRAID Opus with a high number of CPU cores. Modern NVMe drives typically offer 128 or more I/O queues, but certain models may provide fewer. For stable operation, Opus requires that the number of I/O queues per drive be at least twice the number of CPU cores it uses. If the I/O queue count is insufficient, errors like the following may occur when adding a namespace to an NVMe-oF subsystem:

```
ERROR Fail to add namespace to a NVMF subsystem: unable to add namespace, subsystem is in active state; nsid: 1
```

In such cases, the solution is to reduce the number of CPU cores allocated to Opus so that the I/O queue count meets the rule of "I/O queues # $2 \times$ cores." Alternatively, deploying drives with more I/O queues will resolve the issue. To check the number of I/O queues on a drive, run the command:

```
nvmf get-feature /dev/nvme0 -f 7 -H
```

This will return the **Number of Submission Queues Allocated (NSQA)** and **Number of Completion Queues Allocated (NCQA)**. The effective number of I/O queues is the minimum of these two values. For example, if both NSQA and NCQA are 64, the drive provides 64 I/O queues, which would support a maximum of 32 Opus CPU cores. If Opus is running with 48 cores, either reduce the core count to 32 or use a configuration with drives that expose more I/O queues.