# OviOS HA Cluster — Unplanned Failover Report

Nov 6, 2026 · @ovi

**Storage resiliency & high availability** · OviOS Linux · Status: validated in lab testing, 6 November 2026

## 1. Executive summary

This report documents unplanned-failover testing of a three-node OviOS Linux HA storage cluster. For each protocol, the network cable was physically pulled from the active node while a Linux client ran a sustained 4 KiB random read/write workload against it.

**Key finding:** across iSCSI, SMB 3.1.1 and NFS v4.2, the replica took over automatically and the client application saw **zero I/O errors**. Client I/O was held during the takeover — about 35 seconds for iSCSI and SMB, about 60 seconds for NFS — then resumed without any administrator action.

| CLUSTER ARCHITECTURE | FAILURE INJECTED | CLIENT WORKLOAD | APPLICATION I/O ERRORS |
|---|---|---|---|
| **3 nodes** | **Cable pull** | **4 KiB random** | **0** |
| Primary · Replica · Witness | Active node, under full load | 64 I/Os in flight, 70/30 read/write | iSCSI · SMB · NFS |

## 2. Tested protocols and mount configurations

Each protocol was tested on its own, with its own cable pull, from a Linux client using standard kernel clients and no special software.

| Protocol | Mount point | Filesystem / version | Key client options |
|---|---|---|---|
| iSCSI LUN (block) | /mnt/iscsi | ext4 on an OviOS LUN, open-iscsi initiator | Defaults |
| SMB share | /mnt/smb | CIFS, SMB 3.1.1 | `vers=3.1.1,cache=none,actimeo=0,hard,echo_interval=5,nobrl` |
| NFS share | /mnt/nfs | NFS v4.2 over TCP | `vers=4.2,hard,proto=tcp,timeo=600,retrans=2,rsize=1M,wsize=1M` |

On the cluster, the NFSv4 grace and lease time were set to 15 seconds instead of the Linux default of 90 (`options nfs.grace.time 15`, `options nfs.lease.time 15`). SMB required SMB3 as the minimum protocol, with multichannel on.

## 3. Failover performance summary

The figure that matters to an application is the **longest single I/O pause**: how long one request waited while the replica took over. Every request then completed normally.

iSCSI TARGET (BLOCK)

**35.6 s**

longest I/O pause

0 application I/O errors

ext4 stayed writable throughout

SMB 3.1.1 SHARE

**35.0 s**

longest I/O pause

0 application I/O errors

Open files stayed usable

NFS v4.2 SHARE

**59.8 s**

longest I/O pause

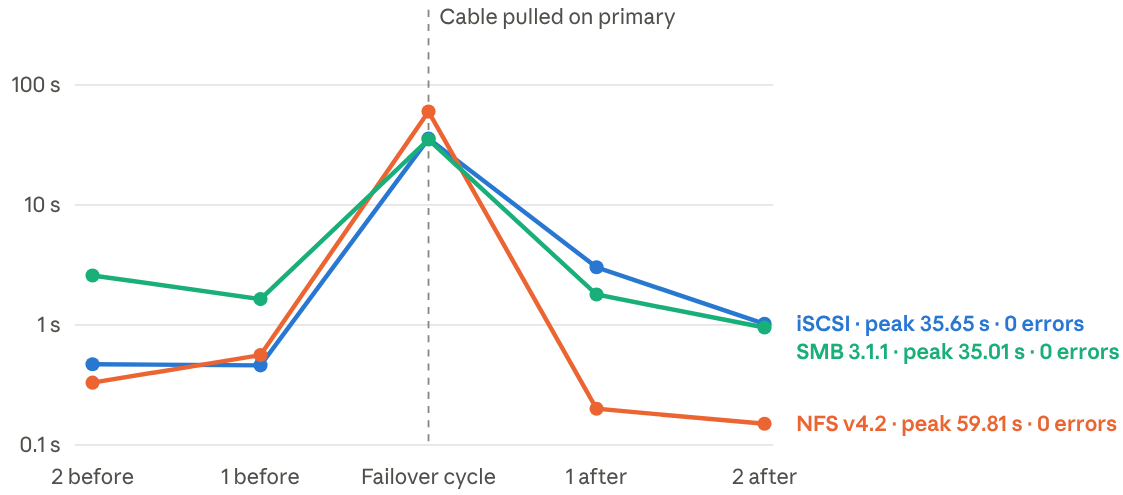
0 application I/O errors

No stale file handles

Outside the failover cycle, the slowest I/O stayed in the sub-second to low-second range on every protocol; the cable pull shows as one spike, and the next cycle is back to normal.

**All three protocols paused once, at the cable pull, and resumed with zero errors**

Slowest single I/O in each 30-second cycle, seconds (log scale)




Two cycles either side of the failover. The cable was pulled in cycle 3 for iSCSI and NFS, and in cycle 5 for SMB.

Client-side fio logs, 6 November 2026 · two cycles either side of each failover

# 4. Detailed protocol breakdown

## 4.1 iSCSI block target

The cable was pulled about 5 seconds into cycle 3 (16:44:46). The Linux iSCSI initiator held its 64 outstanding requests instead of failing them; the longest waited **35.6 s**, and the 30-second cycle completed in 55.8 s with **0 errors**. Writes succeeded in every following cycle, so the ext4 filesystem never switched to read-only. A few I/Os in the first cycle on the replica took up to 3 s as it settled; by the next cycle the slowest was about 1 s.

## 4.2 SMB 3.1.1 share

The cable was pulled a few seconds into cycle 5 (22:14:40). With the `hard` and `echo_interval=5` mount options, the client detected the lost server quickly, held its requests and reconnected to the replica on the same address. The longest request waited **35.0 s**; the cycle completed in 58.1 s with **0 errors**, and all four workers kept their open files. By two cycles later the slowest I/O was under 1 s.

### 4.3 NFS v4.2 share

The cable was pulled about 1 second into cycle 3 (19:29:05). The `hard` mount kept the client retrying; the longest request waited **59.8 s**, and the cycle completed in 72.5 s with **0 errors** and no stale file handles. The following cycles ran normally, with no I/O slower than 0.2 s.

NFS pauses longer than iSCSI and SMB because of the NFSv4 grace period: after a takeover, the server gives clients a window to reclaim their open files before it accepts new ones. With the Linux default of 90 seconds the same test paused for about 139 s; setting grace and lease to 15 seconds removed about 78 s of that.

## 5. Stress-cycle metrics matrix

Every 30-second cycle of every test, from the client-side fio output on 6 November 2026. The failover cycle stretches past 30 s because its requests were held, not failed. In each test the final cycle was stopped by hand after a few seconds, with 0 errors, and is not listed.

| Protocol | Cycle | Runtime | Read IOPS | Write IOPS | Slowest I/O | Errors | Status |
|---|---|---|---|---|---|---|---|
| iSCSI | 1 | 30.0 s | 3,995 | 1,727 | 0.47 s | 0 | Primary |
| iSCSI | 2 | 30.2 s | 2,802 | 1,213 | 0.46 s | 0 | Primary |
| **iSCSI** | **3** | **55.8 s** | **1,509** | **653** | **35.65 s** | **0** | **Cable pull — failover absorbed** |
| iSCSI | 4 | 30.0 s | 3,473 | 1,500 | 3.02 s | 0 | Replica |
| iSCSI | 5 | 30.0 s | 5,027 | 2,170 | 1.02 s | 0 | Replica |
| SMB 3.1.1 | 1 | 30.4 s | 4,584 | 1,980 | 3.88 s | 0 | Primary |
| SMB 3.1.1 | 2 | 33.4 s | 2,427 | 1,048 | 4.21 s | 0 | Primary |
| SMB 3.1.1 | 3 | 30.0 s | 3,228 | 1,398 | 2.58 s | 0 | Primary |
| SMB 3.1.1 | 4 | 30.0 s | 2,470 | 1,068 | 1.64 s | 0 | Primary |
| **SMB 3.1.1** | **5** | **58.1 s** | **1,139** | **492** | **35.01 s** | **0** | **Cable pull — failover absorbed** |
| SMB 3.1.1 | 6 | 30.3 s | 2,330 | 1,008 | 1.79 s | 0 | Replica |
| SMB 3.1.1 | 7 | 30.1 s | 3,599 | 1,557 | 0.95 s | 0 | Replica |
| NFS v4.2 | 1 | 30.1 s | 1,559 | 676 | 0.33 s | 0 | Primary |
| NFS v4.2 | 2 | 30.2 s | 1,180 | 515 | 0.56 s | 0 | Primary |

| Protocol | Cycle | Runtime | Read IOPS | Write IOPS | Slowest I/O | Errors | Status |
|---|---|---|---|---|---|---|---|
| **NFS v4.2** | **3** | **72.5 s** | **155** | **69** | **59.81 s** | **0** | **Cable pull — failover absorbed** |
| NFS v4.2 | 4 | 30.1 s | 1,143 | 500 | 0.20 s | 0 | Replica |
| NFS v4.2 | 5 | 30.1 s | 1,325 | 577 | 0.15 s | 0 | Replica |

## 6. High-availability architecture

An OviOS HA cluster runs three roles, so it can tell a failed node from a broken link and recover with nobody at the console.

| Role | What it does |
|---|---|
| Primary | Serves iSCSI, NFS and SMB on the cluster service address |
| Replica | Stands by, ready to take over the storage and services |
| Witness | Breaks the tie when primary and replica lose sight of each other |

When the primary drops off the network:

1. The replica and the witness agree the primary is gone.
2. The replica takes the active role and brings the storage, the service address, the LUNs and the shares online.
3. The isolated primary detects that it has lost the cluster, and its watchdog fences it: the node reboots rather than keep serving data. This happened as designed in every test.
4. Clients reconnect to the same address and continue. Nothing is reconfigured on the client side.

Self-fencing is what makes this safe. Two nodes writing to the same data at once ("split brain") is the classic way HA storage corrupts data; the watchdog removes the losing node before that can happen.

# 7. Recommended client and cluster configuration

These are the settings used in this test, with the SMB password moved into a credentials file. The one that matters most is `hard`: it makes the client wait for the replica instead of reporting an error.

## Linux clients

```
# SMB 3.1.1 share
mount -t cifs //<service-address>/<share> /mnt/smb \
  -o vers=3.1.1,credentials=/etc/smb-
credentials,cache=none,actimeo=0,hard,echo_interval=5,nobrl

# NFS v4.2 share
mount -t nfs4 <service-address>:/ /mnt/nfs \
  -o vers=4.2,hard,proto=tcp,timeo=600,retrans=2

# iSCSI LUN: log in with open-iscsi, then mount as usual
iscsiadm -m node -T <target-iqn> -p <service-address> --login
mount /dev/<lun-partition> /mnt/iscsi
```

## OviOS cluster

Set once, with the OviOS `options` command:

```
options nfs.grace.time 15   # NFSv4 grace period, seconds (Linux default: 90)
options nfs.lease.time 15   # NFSv4 lease time, seconds (Linux default: 90)
```

The SMB service ran with `smb.min.protocol SMB3` and `smb.multichannel on`; iSCSI ran on the standard port 3260.

## Planning notes

- Always point clients at the cluster service address, never at an individual node.
- Applications and hypervisors have their own I/O timeouts. Allow at least about 60 seconds so a failover is a pause, not an error.

# 8. Conclusion

An OviOS Linux HA cluster turns a hard failure of the active node into a pause, not an outage. With the network cable pulled mid-workload, iSCSI, SMB and NFS clients all kept their connections, saw zero I/O errors and carried on once the replica took over — about 35 seconds for block and SMB storage, about 60 seconds for NFS. The failed node fenced itself every time, so the data was only ever served from one node.

## Test scope

- One unplanned failure per protocol in the final configuration, injected by pulling the primary's network cable under load.
- During tuning, NFS paused for about 139 s with the default 90-second grace period.
- Linux clients only. Windows SMB clients and VMware or Hyper-V iSCSI initiators were not part of this round; Windows handles open files across a server change differently.
- The test measured service continuity and I/O errors. It did not checksum the data written before and after the failover.
- Failback to the original primary was not part of this test.

---

*OviOS Linux — Storage made simple.* ovios.org