napp-it

napp-*it*

napp-it Z-RAID
User's Guide

Setup on OmniOS
with napp-it Pro complete

published: 2016, Sep 14 (c) napp-it.org

Content:

The napp-it Z-RAID SSF functionality is under development with a first preview up from napp-it 16.09 dev.

1. Appliance Z-RAID with SSF (Storage and Service Failover)

Z-RAID vs RAID-Z

ZFS and Raid-Z is a perfect solution when you want data protection against bitrot with end to end data checksums and a crash resistent CopyOnWrite filesystem with snaps and versioning. ZFS Raid-Z protects against disk failures. ZFS replication adds a ultrafast method for async backups even when files are open.

ZFS and Z-RAID is the next step as it adds realtime sync between independent Storage Appliances where ZFS protects against a whole Appliance failure. It also adds Availabilty for NFS and SMB services as it allows a full Storage Server failure with a manual or automatic Pool and NFS/SMB service failover with a shared virtual ip.

RAID-Z

Traditionally you build a ZFS pool from RAID-Z or Raid-1 vdevs from disks. To be protected against a disaster like a fire or flash, you do backups and snaps for daily access of previous versions to access deleted or modified files. In case of a disaster, you can restore data and re-establish services based the last backup state.

Main Problem: there is a delay between your last data state and the backup state. You can reduce the gap with ZFS Async Replication but the problem remains that backup is never up to date. An additional critical point are open files. As ZFS Replication is based on snaps, the last state of a replication is like a sudden poweroff what means that files (or VMs in a virtualisation environment) may be in a corrupted state on the backup.

Another problem is time to re-establish services like NFS or SMB on a server crash. If you need to be back online in a short time, you use a second backup system that is able and prepared to takeover services based on the last backup state. As on Solarish systems, NFS and SMB are integrated in the OS/Kernel/ZFS with the Windows security identifier SID as an extended ZFS attribute (Solarish SMB), this is really troublefree. Even in a Windows AD environment, you only need to import a pool, takeover the ip of the former server, and your clients can access their data with all AD permission settings intact without any additional settings to care about.

Z-RAID SSF

But what about a solution that allows all data on the main server and the backup server to be really in sync? This would mean that you do not use async technologies like backup or replication but sync technologies like mirroring or Raid-Z between your storagervers where ZFS protecs against a whole server or pool failure. This is Z-RAID SSF where you build a ZFS Pool not from disks but from independent storageservers with a local datapool on each with a manual or automatic Z-POOL and service failover in case of problems with the primary server.

What is required to build a ZFS Z-RAID SSF over Appliances on a network?

First you need blockdevices as a ZFS pool is build on blockdevices. A disk is a blockdevice but you can also use files (like on Lofi encrypted ZFS pools) and FC/iSCSI LUNs and the last option is the solution. You know Comstar and iSCSI as a proven technology to create Targets that you may have already used for clients like ESXi, OSX or Windows where you use Comstar to offer LUNs over the network like a local disk.

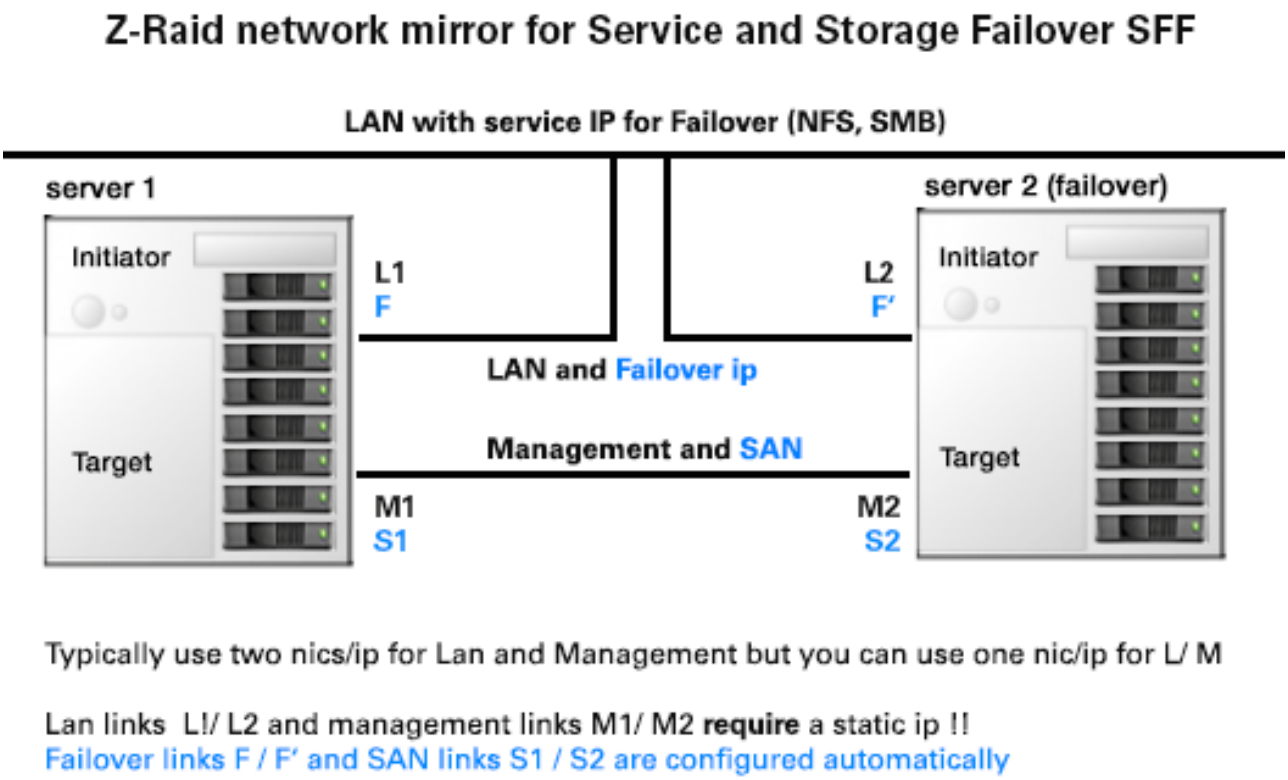
What about using these network LUNs not for other systems but for ZFS itself? You only need a software called Initiator that allows to connect to network LUNs and use them like local disks. If you have not known, this is included in Solarish Comstar as well. When you enable the Initiator with any Target Discovery, it will detect all LUNs from selected Targets and offer them like local disks.

Comstar and iSCSI is proven technology, so the question is, why we have not used this in the past?
The answer is quite simple: Not fast enough on an average 1G network and you need some iSCSI knowledge to set it up. The additional Layer „Disks -> ZFS Pool -> Target -> network -> Initiator -> ZFS Pool" can also slow down compared to a local Pool where the datapath is „Disks -> ZFS Pool".

1.1 Appliance Z-RAID SSF options

Napp-it Z-RAID

Two Storageserver (Master/Slace) where you realtime mirror a Z-RAID pool between them.



Typically use two nics/ip for Lan and Management but you can use one nic/ip for L/ M

Lan links L!/ L2 and management links M1/ M2 **require** a static ip !!
Failover links F / F' and SAN links S1 / S2 are configured automatically

All blue sub-ip settings are defined in menu Pool > Appliance Z-Raid SSF > settings and configured automatically by napp-it. The iSCSI connectivity between them is always on subnet 192.169.101.0 and 192.168.102.0.

Napp-it Z-RAID SSF Basic Setup

For a basic setup for a Z-RAID mirror, you need two ZFS Appliances each with a similar local ZFS pool and a fast link between (10G). You then create a Target on each that offers up to the whole Pool as a LUN. On one of the two Appliances, you enable the Initator to create a ZFS Z-RAID pool as a simple mirror over both LUNs.

Now you have the identical data on both parts of the Z-RAID. If the second Appliance fails, the pool goes to degraded but remains functional with the security of the underlying ZFS pool. If the second server comes back, a resilver starts. If the first Appliance fails completely, you can import the pool on the second Appliance in a degraded state with the security of its local pool. After the import, all services are online. Switch the common virtual LAN IP and your users can continue to work without any further actions needed.

Napp-it Z-RAID Performance

In a 10G network, your performance between your Storageserver (Initiator and Target) is up to 1 GByte/s, with the upcoming 40G Ethernet up to 4 GByte/s what gives you the performance of a very fast local datapool over the network. As you use a whole local ZFS pool as a single network LUN, you can expect a performance level that is only limited by the network with a small extra network latency.

Performance depend on you setup and may require some tunings

Napp-it Z-RAID SSF Advantage

- Realtime Backup/Dataprotection between Appliances without the danger of datacorruption due Snaps
You can place the second Appliance on a different physical location if you have a fast network

-Storage and Service Failover (NFS/SMB)
Even when the whole Master server with its Storage fails, you can switch services to the second storageserver with an up to date data state, This failover can be initiated manually or automatically,

Napp-it Z-RAID Stability

As napp-it Z-RAID is using default iSCSI techniques only, Z-RAID is as stable as the OS or Comstar. As napp-it helps with setup and management, the danger of a misconfiguration on setup or during problems is minimal. During my tests, most problems were timeout problems ex Initiator discovery problems with disk detection waiting very long if a Target is missing. If a Z-RAID pool lost all LUNs you cannot delete the pool without a reboot (pool busy). Of course, Z-RAID does require additional backups.. You can use a second pool on each node.

Napp-it Z-RAID Complexity

HA and cluster solutions add complexity. You must understand the concept of Z-RAID and Comstar iSCSI to setup the cluster and to be prepared for a failure. This includes a regular failover and techniques to recover
a pool on unforeseen problems example with a manual pool import on a Storagehead or switch to a backup either with a data restore or a data share from a backup system. The last is not cluster specific.

Napp-it helps you to reduce the complexity of Z-RAID SSF with a guide and menus for setup and management.

1.2 Appliance Z-RAID SSF setup (napp-it Pro Complete)

The new napp-it Z-RAID SSF functionality is under development and available up from napp-it 17.07 dev

Basic Setup Steps

1. Setup Hardware:
You need at least two napp-it ZFS server, each with an identical pool and one or two nics, best are 10/40G nics, one for LAN and one for SAN and Management. Give the two server hostnames like node1 and node2.
Use a static ip for LAN access and for the nic on the SAN/management network. Avoid ip from subnet 192.168.101.0 or 192.168.102.0 as they are used for traffic between the nodes.

Create a regular ZFS datapool on each server ex tank1 and tank2 with a pool layout according to your use case. Prefer a pool layout that allows any two disks to fail ex n x raid-z2 or mirrors. Prefer SSD only pools for VM usage with high iops and disks for filer/ media usage.

2. Software:
OmniOS and napp-it Pro (17.07 dev or newer) with napp-it Pro complete. Use napp-it menu Services to enable Comstar, Comstar Target and Comstar Initiator service. For tests you can use evaluation keys.

3. Appliance Group
Build an appliance group between the two servers to allow remote control in menu Extensuions > Applince Group

4. Create Target/ LUNs on your nodes:
A LUN is a blockdevice like a disk but a LUN is available in a Target over Ethernet. We will use them to build a Z-RAID. Base of a LUN is a Logical Unit like a ZFS Zvol that is visible in a Target after you set a View.
We will now create the needed Zvol/ LU/ Target/ View combination in two steps:

4.1 Create Zvols, Targets and iSCSI LUNs (you must keep the exact names):
Open napp-it nenu ZFS Filesystem > create on both nodes (node1 and node2 with pools tank1 and tank2) and
Create a ZFS filesystem on tank1. Name the filesystem „zraid-1-lu1" (or zraid-2-lu1 for a second Z-RAID)
Create a ZFS filesystem on tank2. Name the filesystem „zraid-1-lu2" (or zraid-2-lu2 for a second Z-RAID)
Enable iSCSI Sharing on node1 for the filesystem „zraid-1-lu1". Select size ex 90% of capacity of pool tank1.
Enable iSCSI Sharing on node2 for the filesystem „zraid-1-lu2". Select same size like on node1.

If you enable iSCSI Sharing in menu „ZFS filesystems" under iSCSI in the row of the zraid-1-lu filesystem.
use default settings. The iSCSI Target/ LUN is now working. You can connect from an Initiator.

5. iSCSI settings
Enable all Comstar services (Target and Initiator) in menu Srevices > Comstar
Set Initiator to manual discovery of the LUNs from both nodes

4. Z-RAID-Settings:
Open Menu Pools > Appliance Z-RAID > Settings and enter IP adresses for L1, L2, M1 and M2
(Lan interface and SAN/Management interface). This is mandatory.
In a second step you must select the interfaces for L and M to allow auto Cluster ip assignment later.

5. Initiator settings:
Open menu Comstar > Initiator > Static discovery. The selected Targets from 4.) are used per default.
The Targets are bound to to Sub-IP that is generated automatically during a switch Slave > Master (Step 6.).

6. Enable Master
Open menu Pools > Appliance Z-RAID SSF and set one node as a Master. This will enable Target ip.

7. Create one or two Z-RAID
in menu Pools > Appliance Z-RAID SSF > Create Z-RAID.

8. Shares are available over the failover ip, after a manual or auto-failover

1.3 Settings and Management

1.3.1 IP Settings (Menu System > Network ETH)

In a basic config, you need two Storage server for Appliance failover.
Each server has a LAN Interface L1 and L2. This are also the interfaces that are used to offer NFS or SMB services over a failover ip S or S'. You must manually set an IP for this interface to connect the Appliance directly from LAN. The manual IP and the network connectivity is also needed to enable/bring up the interface.

Each server needs a second Interface MH1 and MH2. This are usually the management interfaces. If you manage the appliance over L1/L2, you must set any other ip from an unused ip range to enable the interface/ bring it up, ex 192.168.240.1 and 192.168.240.2. The ip for iSCSO transfer (H1,N1,H2,N2) is set automatically during enabling Z-Raid or a fialover.

Menu Pools > Appliance Z-RAID SFF

This menu shows the state of the Z-RAID and allows to change the state (Master/ Slave)
and to start a manual failover.

napp-it pro napp-it ZFS appliance v. 16.09 dev Aug.26.2016 | logout: admin | sol | Edit | Mon | Acc |

About Help Services System User Disks Pools ZFS Filesystems Snapshots Comstar HA Cluster Jobs Extensions

home » Pools » Appliance Z-RAID SSF PRO Monitor: 10:01 41s Pool Cap Disk Net CPU Job

> Help > Settings > Initiator > Create Z-RAID > HA datapool > poolbased jobs > sub interfaces > Service Log

Z-RAID SSF (complete extension) cmd

Z-RAID settings

Failover mode	Failover IP	Storage Nodes	Heartbeat Head-1	Heartbeat Head-2	Z-RAID Pools on Head-1	Z-RAID Pools Head-2
manual	172.16.100.27	2	-		zraid-1: ONLINE zraid-2: not available	zraid-1: not available zraid-2: not available

Primary Head1 status (this head)

Management ip	Ping	IP adresses	Initiator (iqn)	Z-RAID Targets (iqn)	Initiator Discovery Head1	modify SSF Status
172.16.16.3	is alive	igb6 172.16.16.3 igb6:1 172.16.100.27 (failover ip=Master) i40e2 192.168.1.1 i40e2:2 192.168.101.250 (Target) i40e2:3 192.168.101.1 (Target)	napp-it	tank/zraid-1-lu1	Static: enabled Send Targets: disabled iSNS: disabled	Master

Secondary Head2 status (remote head)

Management ip	Ping	IP adresses	Initiator (iqn)	Z-RAID Targets (iqn)	Initiator Discovery Head2	modify SSF Status
172.16.1.27	is alive	ixgbe2 172.16.1.27 i40e0 192.168.1.2 i40e0:3 192.168.101.2	av-ablage	av/zraid-1-lu2	Static: disabled Send Targets: disabled iSNS: disabled	Slave or undefined

Node1 status (node on this head)

Management ip	Ping	IP adresses	Z-RAID Target (iqn)	Sessions	Initiator (Alias)	Logged in since	SSF Target1 via IP
172.16.16.3	is alive	igb6 172.16.16.3 igb6:1 172.16.100.27 i40e2 192.168.1.1 i40e2:2 192.168.101.250 i40e2:3 192.168.101.1	tank/zraid-1-lu1	1	napp-it	Wed Sep 14 09:59:51 2016	192.168.101.1 alive 192.168.102.1 no answer

Node2 status

Management ip	Ping	IP adresses	Z-RAID Target (iqn)	Sessions	Initiator (Alias)	Logged in since	SSF Target2 via IP
172.16.1.27	is alive	ixgbe2 172.16.1.27 i40e0 192.168.1.2 i40e0:3 192.168.101.2	av/zraid-1-lu2	1	napp-it	Wed Sep 14 09:54:20 2016	192.168.101.2 alive 192.168.102.2 no answer

Current state: manual failover of the netraid-1 pool between the two heads: working
automatic mounting of the Z-Pool on bootup and automatic failover: first tests

For auto-failover: Set Z-RAID to auto in Z-RAID settings and enable failover service (menu services)
Please send an email to community@napp-it.org for questions

2. FAQ and remaining questions

Use cases for Z-RAID SSF

Full Appliance redundancy where
- you want realtime sync between data and backup (usually backup is out of sync, mostly from yesterday)
- you want auto-failover (in case a Master appliance fails on a problem)
- you want manual failover to be able to do service on an appliance (update, repair) without a NFS or SMB service failure.
- you can accept a slighty reduced performance due the realtime network cluster or add a faster network.

How can I debug the manual failover?

The failover mechanism is controlled by:
/var/web-gui/data/napp-it/zfsos/06_Pools/50_Appliance_SSF_Z-RAID=-lin/action.pl (Menu action) and
/var/web-gui/data/napp-it/zfsos/06_Pools/50_Appliance_SSF_Z-RAID=-lin/zraid-lib.pl (common functions)

How can I debug the auto-failover service?

Stop the zraid agent in menu Services and start manually at CLI
perl /var/web-gui/data/napp-it/zfsos/_lib/scripts/agent_zraid_ssf.pl

What should I avoid?

If you are in a situation where all LUNs of a Z-RAID are unavailable, you cannot destroy the Z-RAID pool without a reboot. If you want to destroy a pool, always destroy the Z-RAID pool first then the LUNs.

Can I use the disks on two Appliances directly as base of a Zaid-RAID

Z-RAID is based on network LUNs. This can be created from a Zvol on a base ZFS pool or a raw disk, so yes you can but then the whole storage management must be done by the Master head. With more disks this complicates the whole cluster management. If you use a base ZFS pool instead of raw disks you have an additional ZFS layer that may affect latency but a local ZFS raid and failure management with the option to use the base pool for additional tasks like a backup/ ZFS Replication of the Z-RAID pool over the failover ip. So overall possible but not suggested.

Do I need additional backups with Z-RAID as I aleays have a Failover system that is in sync and up to date

Yes you need. As this is a sync mechanism, there is the ability to destroy them both accidently or intentionally. Only a real external backup can help then. You can use the base pools on Master or Slave for backup or you can use dedicated backupsystems. Simply setup an async replication job with the failover ip as source. If you cannot place your second Slave system on a different physical location, think of an additional backup that is located externally either by an external backup system or removable backup pools that you place externally.

Remaining Problems

Target discovery is using static discovery that is restricted to a sub-ip that is switched automatically on a failover with an easy assignment in Z-RAID settings. Although this reduces timeout/ stall problems compared to a dynamic sendtarget discovery, it happened that the OS is stalled when waiting for targets in a situation when they are not available ex due a node failure or when the node is offering its targets only for the other head. The problem for example is that a format command to show disks never comes to an end.

During regular operation this is not a problem as part of the failover process the target discovery is deactivated/ activated. The target discovery mechanism of the initiator should be improved for situations when the targets become inavailable for whatever reasons. I have added the option to tune timings of the initiator in the Z-RAID menu but the timeout settings is not effective in any case.

2.1 Differences to other Failover solutions
like PaceMaker + Corosync/Hearbeat or RSF-1

A typical OpenSource Cluster solution on Linux is build on PaceMaker as Cluster Resource Manager and Corosync or Heartbeat as a Cluster Communication Manager. This is a very flexible concept as PaceMaker allows to control any sort of services and resources on any node with a lot of possible modifications, scripts and settings to control the services that are involved in the failover process. While you can install PaceMaker on Solaris it is a quite complex solution and requires either a sercice contract with SLA or an IT department to support such a solution inhouse. On Solaris, RSF-1 is a typical commercial Cluster solution with support.

Z-RAID SSF Clusters are different and the approch is „as simple as possible" and „as powerfull as needed".
It reduces the Cluster Resources that are controlled for a failover to „ZFS Pools with NFS and SMB services".
On Solarish you have the unique situation these are pure ZFS resources as SMB and NFS services are fully integrated into the OS/Kernel and ZFS so you do not need a dedicated Cluster Resource Manager for them. For SMB you have the additional advantage that Windows SID in an AD environment are controlled by Solaris ZFS.
A Pool failover on Solarish between two heads that are both AD members allows a failover where all Windows permissions stay intact without any extra efforts or problems.

ZFS itself is the best of all Resource Manager now as a Z-Pool failover (export/import) between the heads automatically controls and enables these services. You only need to add failover management and Cluster Communication that are provided by napp-it. They are quite easy to manage as the Z-Raid SSF concept reduced options to two heads (Master and Slave) with a network mirrored ZFS Pool between them. For higher capacities or performance you can extend this to two heads and up to 6 storage nodes for an iSCSI LUN based Raid-Z2.

Typically you only need to enter some settings like ip, create the LUNs, create a Z-RAID Pool on your Master head and you have a Fileserver with a realtime backup/sync option to a Slave head with manual failover capability. A reboot will keep this situation the same.

You only need to start the failover service on the Slave head for the automatic failover option. If you start the failover service on both heads, the failover process is unidirectionell. Very simple and easy to understand.

2.2 Tuning aspects

Network

Z-RAID performance is limited to network performance. On a untuned 10G network, this means up to 400 MB/s. With some network tuning this may go up to 1000 MB/s (compare http://napp-it.org/doc/downloads/performance_smb2.pdf). I have some Intel X710 (40Gb/s QSFP+) and will add some performance tests with them later.

ARC cache

With Z-RAID we are in a situation where you build a ZFS pool on a ZFS filesystem/zvol each with their own primarycache property. While the ARC cache of the Z-Pool is always local on the Master server, the ARC cache of the underlying Zvol can be on another node. I am undecided if one should modify these settings example to a metadata only for the zvol and all for the Z-RAID to give Z-RAID caching more RAM.

L2ARC cache

L2ARC must be part of the pool. As a Z-RAID pool is a failover pool between Appliances, L2ARC can be either a LUN as well. This will limit access performance to network performance, not perfect. Another option would be a local L2ARC on both heads. In such a case the L2ARC will not be part of the failover what means that the pool lacks the L2ARC after the failover beeing in a degraded state. The solution may be that the failover process removes the former L2ARC cache device and adds the local one after the import.

Sync write/Slog

This question depends on the use case.
If you use a Z-POOL as an SMB filer, you do not need sync, so you can skip this. Keep sync=default on your Z-RAID and writebackcache enabled on the underlaying Locical Unit for best performance,

If you use a Z-POOL as NFS storage for VMs ex in an ESXi environment or for databases, you must use sync write for security reasons or you are in danger that a powerloss or crash failover corrupts your VM or database.

There are mainly four options:
Option1: Use an slog for Z-POOL on both heads and offer as a LUN. Add them to Z-POOL as a mirrorred SLOG. On a storagenode failure, the Slog on the Master keeps available and functional. Although data amount is only a few GB. the latency of your network may ruin your iops values = problem.

Option 2: use a physical Slog on both heads and add to the Z-POOL.
This means that you are safe in a crash/powerloss situation where the same Master comes up or in a Failover with a proper Pool export/import. In a Crash scenario with a Failover the pool lost its Slog with last writes lost.

Option 3: Use the Onpool ZIL for sync write without an extra Slog.
This means that you enable sync (always or default) on your Z-POOL. You also set your Locical Units to Writeback Cache=disabled or the underlying Zvol to sync=always. If you use SSD only pools, you do not need an extra Slog for a good sync write performance. Only care about Enterprise SSDs with poweloss protection.

Option 4: Use an Slog not on your Z-POOL but only on the underlaying datapools that offer the LUNs.
As writes are cached in the RAM of your Master, the storagenodes are not aware of writes. This will lead to a loss of last writes. Not a solution

I would prefer Option 3 especially if you build a Z-RAID from SSDs. If your Z-POOL is build from disks and you need sync write, thgink about SSD only or Option 1.

Pool sync between main Storage and Backup without failover

I only want realtime pool sync between my Storage and my Backup system. Is this possible with napp-it free?

If you do not need the auto or manual failover comfort, you can set up a Z-RAID manually in napp-it Free
- Share your basic pool ex tank (a filesystem) on both server via iSCSI (create a logical unit, a target and a view)
- Enable the Comstar Initiator on your server and enable Target Discovery for your Server and Backup system
- Create a Z-RAID Mirror over both LUNs (each can be your whole pool in size)
- Use your Z-RAID pool for your data. As the Z-RAID keeps the data in sync, you have a realtime backup system.

Name your Z-RAID zraid-1 or zraid-2 to be compatible with napp-it.

2.3 Cluster license:

If you use 50% or more of the usable capacity of your base Pools in the cluster for Z-RAID Failover or HA, you can order a Cluster license. This is napp-it Pro complete for all Appliances in a cluster.

If you use 50% of a base pools for a failover Z-RAID, you can use the other 50% for filer use or a replication back-up with the failover ip as source. You can do this on both heads of a failover mirror to have an additional backup. Add the usable capacity of the base tank-pools. The usable capacity of Z-RAID does not matter.

Example 1:
Head1: Datapool tank1 with 10 TB usable, shared as a LUN for a Z-RAID mirror
Head2: Datapool tank2 with 30 TB usable, 10 TB shared as a LUN for Z-RAID and 20 TB for shares/ backup

This means that you use 10 TB + 10 TB = 20 TB for a Z-RAID ()
and 20 TB for backup or other use cases: A Cluster license is possible

Example 2:
Head1: Datapool tank1 with 20 TB usable, 10 TB shared as a LUN for a Z-RAID mirror and 10 TB for shares/backup
Head2: Datapool tank2 with 20 TB usable, 10 TB shared as a LUN for a Z-RAID mirror and 10 TB for shares/backup

This means that you use 10 TB + 10 TB = 20 TB for a Z-RAID
and 10 TB +10 TB for backup or other use cases: A Cluster license is possible

Example 3:
Head1: Datapool tank1 with 20 TB usable, 10 TB shared as a LUN for a Z-RAID mirror and 10 TB for shares/backup
Head2: Datapool tank2 with 30 TB usable, 10 TB shared as a LUN for a Z-RAID mirror and 20 TB for shares/backup

This means that you use 10 TB + 10 TB = 20 TB for a Z-RAID
and 10 TB +20 TB for backup or other cases: A Cluster license is not possible as this exceeds the Z-RAID capacity.
You then need individual napp-it Pro Cpmplete licenses, either on a per server or per site/location base.

If you use two Z-RAID Pools or more than one datapool, the added base capacity of all pools and the added Z-RAID capacity is relevant.

How should I build a Storage Cluster with Z-RAIF and Storage/Service Failover?

- Use two identical Storage Appliances each witch a similar ZFS pool tank1 and tank2
or two pools each like a disk pool for a Failover SMB pool and SSDs for a Failover NFS pool.

- Place the two Appliances at different physical locations (if possible as you need a fast network between)
- Use redundant PSU, add an UPS on one line only to allow a Power or a UPS failure
- Prefer a direct link between them (Initiator/Target connectivity) or a redundant network

Such a cluster allows a complete server failure (head and storage) without a service failure as there
is a switch from the Master server to the Slave server with a full Storagepool and NFS/SMB Service Failover.

3. High performance 40G Connection between Appliances

For Z-Raid SSF performance, network performance is essential. Ethernet 1G is similar to old ATA disks with around 100MB. This is not fast enough for appliance Z-RAID.
10G Ethernet can give up to 1000MB/s. This is enough for a Z-RAID and offers enough performance for a traditional disk based pool or many use cases.

The upcoming QSFP+ standard offers 40G ethernet or 4 x 10G SFP+ over a breakout cable. This is in the performance area of a local high performance pool or a very fast NVMe solution. As the price of 40G adapters are quite similar with 10G server adapters, this is the best solution beside FC/IB solutions. While FC/IB may have advantages regarding latency, iSCSI is mainstream and QSFP+ is available in a lot of ethernet switches.

For 40G connectivity you can use either a new MPO (multi-fibre push-on) connector with a copper MPO DAC cable up to 5m for local connectivity, a 12 fiber MPO optical tranceiver and MPO fiber patchcables for connectivity inside a serverroom or between different brands. For long distance connectivity you can use CWDM tranceivers with an LC fiber connector. They use 4 different colors simultaniously over a single fiber to achieve 40G over a traditional Multimode (up to 300m) or Singlemode (long distance) fiber.

40G QSFP+ for a same room Appliance <-> Appliance failover setup
- 2 x Intel XL 710 DA1 (around 420 Euro each, Dualport XL 710-DA2 is at around 480 Euro)
- QSFP + DAC copper cable (1 to 5 m), either genuine Intel or compatible
  Intel XLDACBL1 (..BL5) compatible, compatible cables up from 90 Euro

40G QSFP+ for a local Appliance <-> Appliance failover setup
- 2 x Intel XL 710 DA1 (around 420 Euro each, Dualport XL 710-DA2 is at around 480 Euro)
- 2 x Intel Tranceiver Intel E40GQSFPSR or compatible (compatible up from 200 Euro)
- MPO Patchkabel or 2 x breakout cable MPO -> 12 x LC (multimode cabling) -> MPO

40G QSFP+ for a remote Appliance <-> Appliance failover setup
- 2 x Intel XL 710 DA1(around 420 Euro each, Dualport XL 710-DA2 is at around 480 Euro)
- 2 x CWDM Tranceiver (several 10G links over different colors) for 10G over
  Multimode od Singlemode LWL (ask for compatibility with Intel)

40G QSFP+ with a 40G Site to Site or building to building connectivity
- 2 Switches with QSFP+ ports ex H3C
- 2 x CWDM Tranceiver (several 10G links over different colors) for 10G over
  Multimode od Singlemode LWL (ask for compatibility with your switch type)
  ex (H3C) HPEX 140 for 2km (JL286A) or 10km (JG661A) or compatible up from 800 Euro

Connect your Appliance to the QSFP+ port of a switch
- 2 x QSFP+ MPO Tranceiver for your Switch ex HPE X140 MPO (JG709A)
  or compatible up from 200 Euro
- 2 x Intel XL 710 DA1 (around 420 Euro each, Dualport XL 710-DA2 is at around 480 Euro) for the appliances
- 2 x MPO Tranceiver Intel E40GQSFPSR or compatible (Intel compatible up from 200 Euro)
- 2 MPO Patchcable to connect the XL710 to the switch, price depends on length up from 100 Euro

If you simply want 40G between a QSFP+ switch and your storage appliance
- 1 x Intel XL 710 DA1 (around 420 Euro each, Dualport XL 710-DA2 is at around 480 Euro)
- 1 x MPO Tranceiver Intel E40GQSFPSR or compatible up from 200 Euro
- 1 x MPO Tranceiver for your Switch ex H3C: HPE X140 MPO (JG709A) or compatible up from 200 Euro
- 1 MPIO patchcable, price depends on length up from 100 Euro

prices are estimated prices in Germany without vat/ sales tax, current offers for example www.flexoptix.net

4. 40G Raid-Z Benchmarks

4.1 Raid-1 between appliances directly connected via QSFP+ 40G DAC cable

Z-Raid Mirror 100GB, XL710 with Jumboframes enabled
Storage on node1a, 4 x Intel P750-400 in Raid-0
Storage on node1b, v2: 6 x Intel P750-400 in Raid-0
Storage on node2: 2 x 6 disk Raid Z2 vdev of Sandisk Pro extreme 960GB (12 SSD)

Filebench, 1 singlestreamwrite.f; local datapool on node1 (base of LUN1)
Storage on node1: 4 x Intel P740 Raid-0
94754 ops, 3158.278 ops/s, (0/3158 r/w), 3158.2mb/s, 844us cpu/op, 0.3ms latency

Filebench, 1b singlestreamwrite.f; local datapool on node1 (base of LUN1b)
Storage on node1b: 6 x Samsung SSD MZ7KM480 in Raid-Z2
56691 ops, 1888.752 ops/s, (0/1889 r/w), 1888.7mb/s, 2027us cpu/op, 0.5ms latency

Filebench, fivestreamwrite.f; local datapool on node2 (base of LUN2)
2 x 6 disk Raid Z2 vdev of Sandisk Pro extreme 960GB (12 SSD)

Filebench, singlestreamwrite.f; Z-RAID (iSCSI network mirror over LUN1, LUN2)

4.2 Windows 10 -> NAS 40G performance with QSFP+

I made some AjA Benchmarks (4k, RGB) with same tuning options as on 10G
(Jumboframes, ip buffer increased. int throtteling off)
Solaris 11.3 -> Windows 10: around 2200 MB/s write and 300 MB/s read
OmniOS 151018/19 -> Windows 10: around 1400 MB/s write and 140 MB/s read

Especially the low read values are not acceptable. I have seen the same with 10G on the default Windows driver.
The newest Intel 10G drivers from Intel solved that - wait and see!

5. more docs

napp-it Homepage:
http://www.napp-it.org

How to setup the napp-it ZFS storage server
http://www.napp-it.org/doc/downloads/napp-it.pdf

How to setup napp-in-one (virtualized storage server on ESXi)
http://www.napp-it.org/doc/downloads/napp-in-one.pdf

Performancetuning with 10G and SMB2
http://napp-it.org/doc/downloads/performance_smb2.pdf

Download napp-it ToGo (ready to use images for a barebone setup or ESXi template)
http://napp-it.org/downloads/index_en.html

Howto setup OmniOS manually
http://napp-it.org/downloads/omnios_en.html