napp-it

One-Pagers for a napp-it ZFS Storageserver

Checklist for beginners

published: 2023-Nov-22

One Pagers

1. Setup OmniOS and napp-it
2. Server, Pool and Filesystems options
3. SMB filer
4. NFS filer
5. iSCSI filer
6. Essential filer and maintenance settings
7. Trouble fixing
8. Tuning
9. Hybrid Pool Tiering

1. Setup OS and napp-it

1.1 Use a server with a 30 GB min bootdisk (Sata, M.2, ESXi vdisk) and at least 4 GB RAM

prefer server class hardware with ECC RAM and Sata/AHCI or
LSI/Broadcom SAS HBAs with 2008, 2307, 9003, 93xx, 94xx series

https://www.broadcom.com/products/storage/host-bus-adapters
https://forums.servethehome.com/index.php?threads/lsi-raid-controller-and-hba-complete-listing-plus-oem-models.599/

1.2 Download OmniOS/OI/Solaris (usb-image, on OI or Solaris text or GUI edition)

OmniOS from https://omnios.org/download.html
OpenIndiana from https://www.openindiana.org/downloads/
Solaris https://www.oracle.com/de/solaris/solaris11/downloads/solaris-downloads.html

1.3 Download USB imager for Windows from
https://www.napp-it.org/doc/downloads/usb_image.zip

1.4 Use the imager to create a bootable USB stick (1GB OmniOS, 4 GB for Solaris or OI GUI edition)

Boot your system from the USB stick
Select install to disk and answer the setup questions. Confirm the bootdisk name=rpool
Configure network settings, preferred to auto/dhcp (requires an active DHCP server)
Manual network setup, see https://napp-it.org/downloads/omnios_en.html
reboot

1.4 Install napp-it (working internet connectivity required)

login as root and enter: wget -O - www. napp-it.org/nappit | perl

when setup is finished, (re)enter your root pw: passwd root
a reboot is now recommended

login into napp-it. Open a browser and enter: http://serverip:81
(replace serverip with the ip of your server as shown after setup or via ifconfig -a)

1.5 Update OmniOS and OmniOS to newest state:

login as root at console: pkg update
if pkg itself is outdated: pkg update pkg

Open napp-it and select menu About > Update
update to newest version, either Free or newest Pro edition (dev edition when needed)

1.6 Problems

If something happens, reboot and select a former bootenvironment.
Or reinstall napp-it via wget command. This preserves all setting.

1.7 Manuals
see https://napp-it.org/manuals/index_en.html

2. Server, Pools and filesystems options

2.1 Server concept

- one filer + backup disks
- one filer + one or more backup server/ disaster failover server
- two filers in a HA/Cluster setup + one or more backup server/ disaster failover server

2.2 Disk/ Pool concept

- Sata disks (for a few local disks, hot removeable, avoid Expanders)
- SAS disks (up to hundreds, hot removeable, Expander is a good option, prefer 12G)
- Multipath SAS disks or Multipath SAS Jbods with dual expander, suggested for a ZFS HA Cluster
- Care about redundancy (Best is to use a pool layout thats allows a failure of any two disks)

2.1 Basic Filer

- use a dedicated bootdisk (30GB min, suggested 60GB+) for rpool (system pool)
- create a ZFS pools from disks (high capacity, cheap, slow, hot removable disks)
- create a ZFS pool from Sata/SAS SSDs (medium size capacity, affordable, fast, hot removeable)
- create a ZFS pool from NVMe (medium size, expensive, very fast, non hot removeable)
- create a ZFS pool from Intel Optane (small size, expensive, simply the fastest, non hot removeable)

For very different use cases, create two pools, one from disks, one from SSDs/NVMe

2.2 VM storage, mailserver, database server

- use an SSD/NVMe pool
- acticate sync and add an Slog on diskpools (Intel Optane, WD Ultrastar SS530)
  or use a fast enough pool without Slog and sync enabled (Intel Optane, WD Ultrastar SS530)

2.3 Care about backup, data security and availability

- basic: use a server with a backplane with enough bays for a backup pool or external USB disks
  Replicate your datapool regularly. After say a month replace the backup pool with a second one.
  Place the backup disks disks at a different location. After a month swap the backup pool again and
  continue replication. Use snaps for versioning/ protection against Ransomware.

- advanced: use a second backup server, best at a different physical location/ fire department
  Replicate datapool regularly, can be down to a minute delay. Use snaps for versioning/ protection
  against Ransomware. Use a second backupserver if you cannot afford a backup interruption.

- superiour: use a ZFS HA cluster setup with two server heads and a common multipath storage.
  This allows server maintenance/ backup/ security fixes while the second head takes over.
  With two physical head servers and two dual expander SAS Jbod boxes, any head or Jbod can fail
  without a service outage (create a pool that is redundant over Jbods ex 2way mirror or 4way mirror)

- Network replication or a ZFS cluster requires napp-it Pro.
  see https://www.napp-it.org/extensions/quotation_en.html
  http://www.napp-it.org/doc/downloads/featuresheet.pdf

2.4 Pool move

You can move ZFS pools between different servers with different disk controllers.
You must only care that the new server supports the activated ZFS features.

2.5 Manuals, see https://napp-it.org/manuals/index_en.html

3. SMB filer

3.1 SMB Server options

- SAMBA (available either in the OS repo or from Joyent, http://pkgsrc.joyent.com/packages/SmartOS/)
- ZFS/ kernelbased and multithreaded Solarish SMB server with full NFS v4 ACL support in ZFS that allows a file backup/restore/move with AD permissions remain intact without mapping uid-SID

Mostly you use the kernelbased SMB server due its simplicity (it just works), better integration of Windows ntfs alike ACL permissions with support of Windows SID and perfectly working ZFS snaps=Windows previous versions and local SMB groups. AD integration is built in as default.

The kernelbased SMB server is supported by napp-it

3.2 Enable an SMB share

On Solarish, a SMB share is a property of a ZFS filesystem. To enable a share, just set the ZFS SMB share property to on. This can be done when you create a filesystem or in menu ZFS filesystems when you click to „off" in the row of a filesystem under SMB. As an option you can enable guest access or ABE/ access based enumeration. You can share under the filesystem name or with a different name. If you want to make the share invisible, add a $ ex share$.

3.1 SMB Server properties

Unlike SAMBA, there is no config file. Properties are either ZFS properties or can be set in menu Services > SMB > properties.

3.2 User

After enabling a share in napp-it, the default is that any lokal Solarish user can access the share. You can either setup a share with guest access where no login/password is required or you can add local users on your Solarish server. If you want a centralized user database like Active Directory, join the domain in menu Services > SMB > Active Directory. Optionally add an id-mapping then winuser:domainadmin=Unixuser root (suggested but not required, SMB connect as root then).

If you create the same user/pw than you use on Windows, you can connect without entering a password again. This is similar to connecting AD servers without extra login.

3.3 Permissions

You can restrict permissions either from napp-it in Menu ZFS filesystems and folder ACL or from Windows after you SMB connect as root. Klick on a folder with a right mouse + property > security.

Beside File and folder ACL, you can add ACLs to a share. Use this mainly to set an additional restriction ex a temporary readonly. Share ACL are per default everyone=full. When you disable/enable a share the setting reverts to default.

3.4 Backup/restore pools with permissions intact

If you move a pool from a Solarish server that is a memer of an AD domain to another member server, all permissions remain intact as Solarish use the genuine Windows AD SIDs for file permissions.
If you move a pool from one server to another in workgroup mode, all permissions remain intact when you mirror all Unix users, Unix groups and SMB groups on the new server - otherwise you must reset/ set all permissions newly.

3.5 Manuals, see SMB sharing on Solaris or https://napp-it.org/manuals/index_en.html

4. NFS filer

4.1 NFS Server options

Illumos (OmniOS, OpenIndiana) supports NFS3 and NFS4
Solaris 11.4 supports NFS3 and NFS4.2

4.2 NFS3

NFS3 is a network filesystem that gives you a high performance access to a remote storage. It was developped by Sun. You should use it in secure environments only as there is no authentication (login) or authorisation (permissions). Access is granted based on client ip and Unix client user uid (can be nobody on some systems).

Typical use case is VM storage ex in an ESXi environment. If security is a concern, use it in a dedicated vlan/lan without access from unsecure lans.

In an environment where you want concurrent access via NFS and SMB ex for easy access to data and snaps, set file permissions to everyone@)modify. If you need to restrict SMB access, use share permissions.

4.3 Enable and restrict access to shares

You can enable NFS shares in menu ZFS filesystems when you click on „off" in the row of a filesystem under NFS. Set it simply to „on" to enable.

You can enter share options instead the simple „on".
rw=host1:host2:host3,root=host1:host2:host3,ro=host4 ex
rw=@192.168.1.0/24,root=@192.168.1.0/24

4.4 NFS Tuning

Open menu Appliance > Tuning and increase tcp, NFS buffers and settings.
This is mainly an option for networks > 1G

If you use NFS with ESXi, you should use a vmxnet3s vnic where you should increase the buffer settings as well. Optionally Jumbo Frames can improve performance as well

Jumbo Frames can improve performance but can be a source of troubles.
On any comminication problems, disable Jumbo and test again.

4.5 Client access

When you connect an SMB share, you must enter the full path
Connect to NFS share via serverip/pool/filesystem

On a Mac, connect via nfs://serverip/pool/filesystem

4. Manuals

NFS sharing on Solaris or
https://napp-it.org/manuals/index_en.html

5. iSCSI filer

5.1 Use case

Unlike NFS and SMB that are multiuser access sharing options, iSCSI is a method to share parts of a filer storage to single client computers where it is connected like a local disk and formatted with the client filesystem ex ext4, ntfs or vmfs. Concurrent client access is only allowed under control of a Cluster management software.

So the use case of iSCSI is to replace a local client disk with a LUN in a SAN server. The main advantage over a physical local disk is expandability (disk can grow), security (ZFS based), backup (ZFS replication) and moveability (disconnect/connect on another client).

Most use cases of iSCSI can be covered by NFS in a much simpler way but some client applications want local disks. This is where iSCSI is unique in its flexability.

5.2 Comstar

The Comstar FC/iSCSI framework is an enterprise ready base for iSCSI sharing that can scale from small setups up to very large installations. It supports Logical Units (zvols, files, disks), Targets, Target Groups, Host Groups, Target Portal Groups and client authentication via Radius and Chap. Views can be defined to make a Logical Unit visible in a target. You can then access the Logical Units as a LUN from a client like ESXi, Mac, Linux, Solaris or Windows.

5.3 Enable an iSCSI share

Enable Comstar target and optionally initiator service (can connect a LUN from another server) in menu Services > Comstar.
You can enable iSCSI LUNs in menu ZFS filesystems in menu ZFS filesystems when you click on „ZFS unset“ in the row of a filesystem under iSCSI. Set iSCSI sharing to on with a desired size setting.

You can now access the LUN from a client. A target and a view is created automatically
For more special settings, see menu Comstar.

5.4 Restrict LUN access

Chap: Goto menu Comstar > Targets > allow initiator chap and enter an initiator iqn and a password
Initiators: Create a Host group with your initiators
Client-ip range: Create a Target Portal group that listens to a given server ip

5.5 Backup/ restore a LUN

If your LUN is based on a zvol (a ZFS filesystem treated as blockdevice), you can move/backup the zvol via ZFS replication. To re-enable it as a Logical Unit, simply activate it in menu ZFS filesystems.
As napp-it creates a zvol with the iSCSI guid as part of the zvol name, a Lun can be accessed without modifications from a Client unde the new server location.

If you have created a zvol manually, you must import via Comstar > Logical Units > Import where you can enter the former guid or create a new guid. Data on the Logical Unit is not modified during an import.

5.6 Backup/ restore Comstar settings

You can use Comstar > Full HA save/restore. This will save/restore all settings without a reboot (Cluster aware, napp-it Pro). A Basic save/restore is a free option and requires a reboot.

5.7 Manuals see Solaris Comstar Administration or https://napp-it.org/manuals/index_en.html

6. Essential filer and maintenance settings

Your filer is up and running, what now

6.1 Settings

Open About > Settings and enter passwords and your emailadress
Select a menu set. Default is „sol" (Solaris) with supported options. Others like en or de can give different options and languages including unsupported community options..

6.1 Get status mail and alerts on problems

Create reports in menu Jobs > Reports (flexible status and alerts, TLS, Port 25, napp-it Pro)
Mail method and reports are job parameters. Reports are user extendable.

or basic (free) status/alerts in menu Jobs > Email
The basic status/alert mail function can use encrypted TLS mail or standart port 25.
The method must be switched manually in menu Jobs. This can be resetted on an update.

If you use Gmail (requires TLS normally):
You can send mails over port 25 for Gmail users over the Gmail relay smtp-relay.gmail.com
https://support.google.com/a/answer/176600?hl=en

6.2 System and data backup

Create a backup job (menu Jobs > Backup) that saves all important napp-it, job, user and Comstar settings to your datapool. If you need to reinstall the OS you can resore them manually or via menu Users > Restore or Comstar Full HA save/restore on napp-it Pro.

Create a replication job to backup your current boot environment to a backup pool.
This is a full disaster backup with the whole current OS with all settings.
After a Crash, resore this BE and boot into.

Create a replication to backup your datapools to a backup pool on same server or to another pool over the network (napp-it Pro)

6.3 Snaps/versioning

Zfs allows to create read-only snaps. They are Ransomware save as even with Admin permissions a Windows client cannot destroy them (unlike Windows shadow copies). Snaps (and ZFS redundancy) is the main way to protect data. External backup is important as a disaster backup ex for a fire.

There is no practical limit in number of snaps. You may create/keep a snap every 15 minutes in current hour, a snap every hour in current day and a snap per day for current months and a snap per month for this and last year. For replications you can create a different snap history on backups.

6.4 other jobs

Create a scrub job to verify data on rpool and datapools
Run it once a month (in a low activity time like weekend)

Create an „other job" to sync date and time via ntpdate (ignore result)

6.5 Disk map

Create a disk map and print it out (or write down WWN and slots) in menu Disks > Map to identify disks on problems

7. Trouble fixing

Your filer has problems!

7.1 Disk problems

Most problems are disk problems. If a disk fails (you want an email alert) and you have added a hot spare disk to your pool, the bad disk is replaced automatically. If you have several spare disks, use one as hot spare and the others as cold spare to avoid a multiple auto replace on a flaky system.

To manually replace a bad disk, simple hot-insert a new disk and start a Disks > Replace bad/new.

If a checksum error happens, a pool state may remain in degraded state.
Delete/ restore the bad files then and start a scrub.

A disk replace mirrors a bad disk with a new one. If the bad disk remain offline after a successful replace, do a Disks > Remove of the bad disk.

If you move disks to a new controller and they are notz properly detected, do a Pool > Export + Pools > Import. This will detect all disks properly.

If a Pool is in state Degraded while all Disks are now repaired,
clear the error via menu Pools > Clear error

If you have performance problems (or iSCSI disconnect/timeouts):

Check for weak disks in menu Pools (iostat soft/hard/transfer errors), Smart (menu Disks > Smart) or check iosts for all disks. Look especially for bad busy% or wait% values. All disks in a pool should perform similar.

Check System > Logs and System > Faults.

Replace weak disks and check them with a full disk test ex via WD Data Lifeguard on Windows.
(WD data lifeguard is on a Hirens USB bootstick, https://www.hirensbootcd.org/)

If a pool import fails or the OS hangs after bootup, remove all datadisks.
When the OS boots up, insert disk by disk and check if it is recognized correctly without a hanging OS.

7.2 Network problems

check cables
deaktivate Jumbo or port aggregation when enabled
re-set ip and gateway

7.3 Unknown problems

Have you modified anything recently that may be the reason?

Start from an absolute minimal system/ minimal RAM and bootup (optionally with a new bootdisk) and add part by part until the problem appears

or remove part by part (disks, HBA, RAM) until the problem disappears.

Many unknown problems are RAM problems. Remove half the RAM, test, test with the other half.

If possible, use a second system to do cross-checks (switch parts over)

8. Tuning

Your filer is too slow!

8.1 network tuning

- increase tcp or vmxnet3 buffers
- use Jumbo frames (9000 + tcp header 216B =9216)

8.2 SMB Tuning

- disable SMB encrypt

8.2 iSCSI Tuning

- enable writeback
- if you need secure sync write, add an slog on HD pools or set sync=always

8.2 NFS Tuning

- increase NFS servers and buffers
- if you need secure sync write, add an slog on HD pools or set sync=always

8.3 Filesystem Tuning

Readcache: ZFS caches read last/ read most ZFS datablocks in Arc/L2Arc, not whole files
default Arcsize is around 70% RAM.

With many volatile files, many users and a slow pool, a large readcache is the key for performance
- add more RAM (ARC readcache), the more user the more files the more RAM (check arcstat)
- add L2Arc, max 10x RAM, not as fast as ARC readcache but persistent with the option to enable read ahead
- add an Slog for hd based pools, with NVMe use those with powerloss protection and enable sync without Slog
- adjust filesystem recsize according to use cases es 16-64K for databases/VMs and up to 1M for a media filer
- enable LZ4 compress to reduce amount of io
- enable dedup (use only with dedup rates >>5 and a lot of RAM or a dedup special vdev mirror)

Writecache: ZFS caches writes in RAM (default 10% RAM, max 4GB)

Main goal is to collect data to avoid very small writes below 64K. Calculate size upon number of active users. When cache fills up 50%, it is written so a thumb-rule is: Concurrent users x 2 x 64K=suggested writecache. With many active users, a filer should have at least 32GB RAM to provide enough RAM for read/write caching.

8.4. Hybrid Pools

Hybrid pools are build from cheap and large diskbased vdevs with additional faster and smaller flash based vdevs. You can add an L2Arc but this does only improve reads on small io and metadata, not whole files or writes. The Slog is not a cache but a protection of the rambased writecache for secure sync writes. With a diskbased pool and sync enabled, an Slog can help a lot. Without sync an Slog is not used at all.

9. Special vdev

If you want to improve read and write performance, you can add a special vdev. This is not a cache but a tiering method where you place most data on cheaper Tier-2 areas of a pool and some critical data like small io, metadate or whole single filesystems on a faster flashbased Tier-1 area. As a special vdevis not a cache but the place where data is stored, a special vdev lost is a whole pool lost so you need redundancy, either a 2way or 3way mirror for critical data. Raid-Z is currently not supported.

If you add a special vdev to a pool, you must decide what data you want on the special vdev.
All datablocks with a blocksize <= small blocksize are stored on the special vdev, other data on other vdevs.

Option 1: Small io and metadata

When ZFS writes data, it splits a file into blocks. Maximum size of datablocks can be set with the recsize setting of a filesystem. Files smaller recsize is stored with a dynymically reduced blocksize (beside draid with a fixed blocksize), ex a 16K file with a default 128K recsize uses only a single 16K block while a 200K file produces two 128K blocks.

The size of datablocks is the key to decide which data land on the slower or faster pool area.
If you want only the extrem performance sensitive small io files with their snaps and metadata on a fast special vdev mirror and all other data on the slower vdevs, you can set the small block size of a filesystem to say 8k. As most data is on the regular pool, size of the special vdev can be 5-20% of poolsize.

Option 2: Small io, metadata and average files like office docs

If you use a larger small blocksize value ex 128K, all files with a blocksize up to 128K are stored on the special vdev. If you have set recsize=128K all files of this filesystem are stored on the special vdev. With a larger recsize ex 256K-1M, only files and their snaps with a size < 128K land on the special vdev, larger files ex mediafiles land on the other vdevs. This requires a larger special vdev capacity of say 20-60% of poolsize.

Option 3: Manual tiering of files between Tier-1 and Tier-2 storage

As needed capacity for files and snaps on a special vdev increases over time, you may want to move files manually between Tiers ex with all new or edited data on Tier-1 and moved to Tier 2 after some time. This is currently not supported in ZFS but there is a workaround. If you switch recsize below or above the small blocksize, rename a file ex to file.tier and copy (not move) back to file, it is stored at the Tier that is set for the new blocksize. You can check blocksizes of files via zdb filesystem ex zdb tank/data. As long as there is enough capacity on a special vdev, the dblk value gives a hint about the file location with its inode number. A command like find /tank/data -type f -exec stat -c "%i %n %z" {} \; gives a list of files with inode number and last edit time.

Keep in mind
Snaps are readonly and remain on the Tier. With Tiering storage reduce number or time to keep snaps. If you backup to Tier-2 via replication, you can preserve there more snaps via the keep and hold job settings.

About special vdevs

If a special vdev is full all further writes go to the other vdevs. You can add additional special vdevs to increase size. A very special type of a special vdev is a dedup special vdev to store dedup tables onto instead of expensive RAM that you mostly want for caching performance not for dedup tables.

more, see https://www.napp-it.org/manuals/index_en.html