

SMB Server on OmniOS or Solaris

First steps

Kernel based/ OS included Solarish SMB server

1. About
 - 1.1 Basics
 - 1.2 enable SMB shares
 - 1.3 advanced SMB settings

2. Solarish ACL
 - 2.1 Solaris ACL vs Windows ntfs ACL
 - 2.2 ACL inheritance
 - 2.3 Share ACL

3. Setup ACL
 - 3.1 ACL for toplevel folder
 - 3.2 ACL for folders below
 - 3.3 ACL and classic Unix permissions
 - 3.4 ACL and Macs
 - 3.5 NFS v3 and SMB

4. OmniOS server for Apple clients

5. more manuals and infos

1.3 Advanced SMB server settings

If you need some advanced server options like locking, signing, maxprotocol or encrypt, goto menu Services > SMB > properties . Some properties like global inheritance options are ZFS filesystem properties. Click on ACL on files in menu ZFS filesystems to modify. Windows alike is aclinherit=pass-through. You can also restrict chmod actions ex via NFS via the aclmode property.

Be careful on nested ZFS filesystems below a share as lower filesystem can have different ZFS properties like casesensitivity, character sets or nbmand locking options what can cause problems. OmniOS allows nested shares, Solaris not to avoid problems.

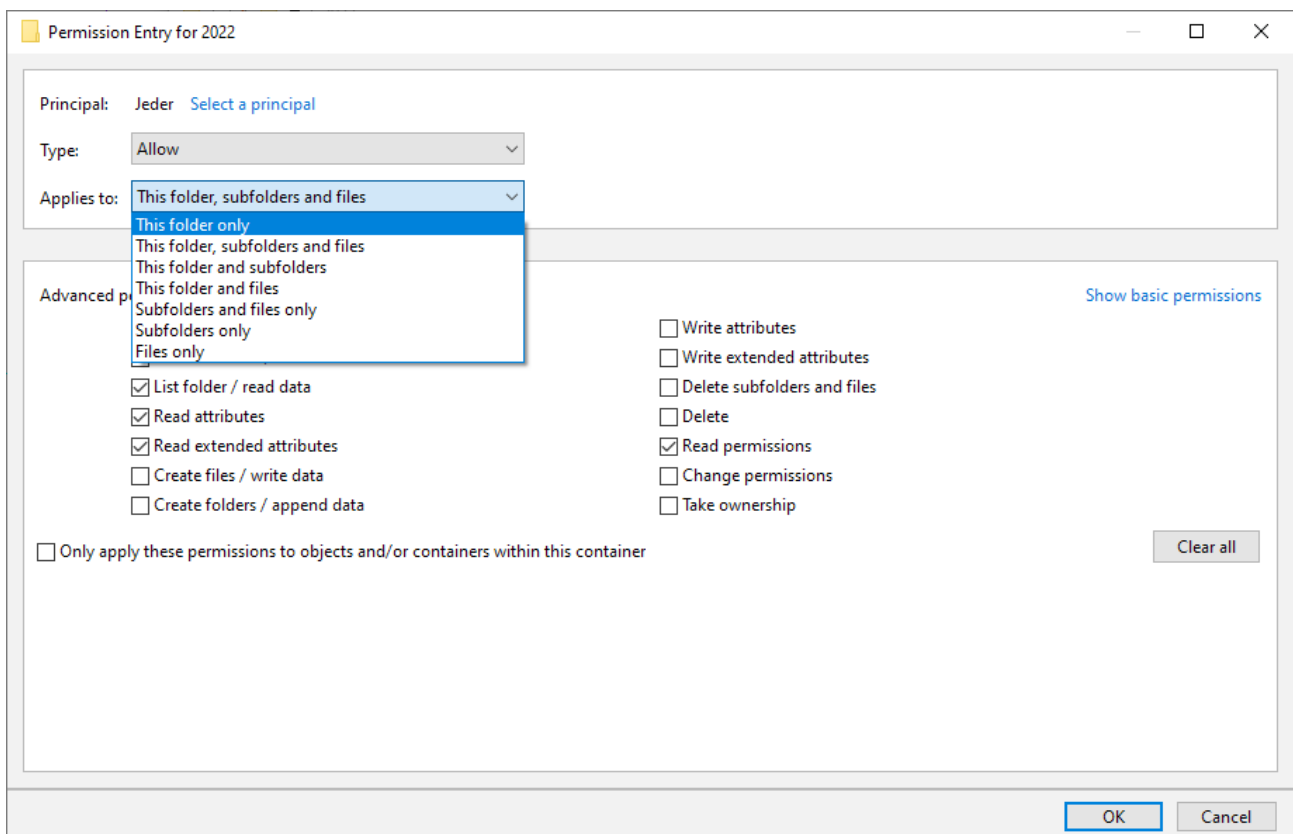
If you use Apple systems you can enable Bonjour/mdns in menu Services > Bonjour and autostart, allow Time-Machine backups and display OmniOS via a nice Xserve icon.

2. Solarish ACL

2.1 Solarish nfsv4 ACL vs Windows ntfs ACL

https://docs.oracle.com/cd/E23824_01/html/821-1448/gbacb.html

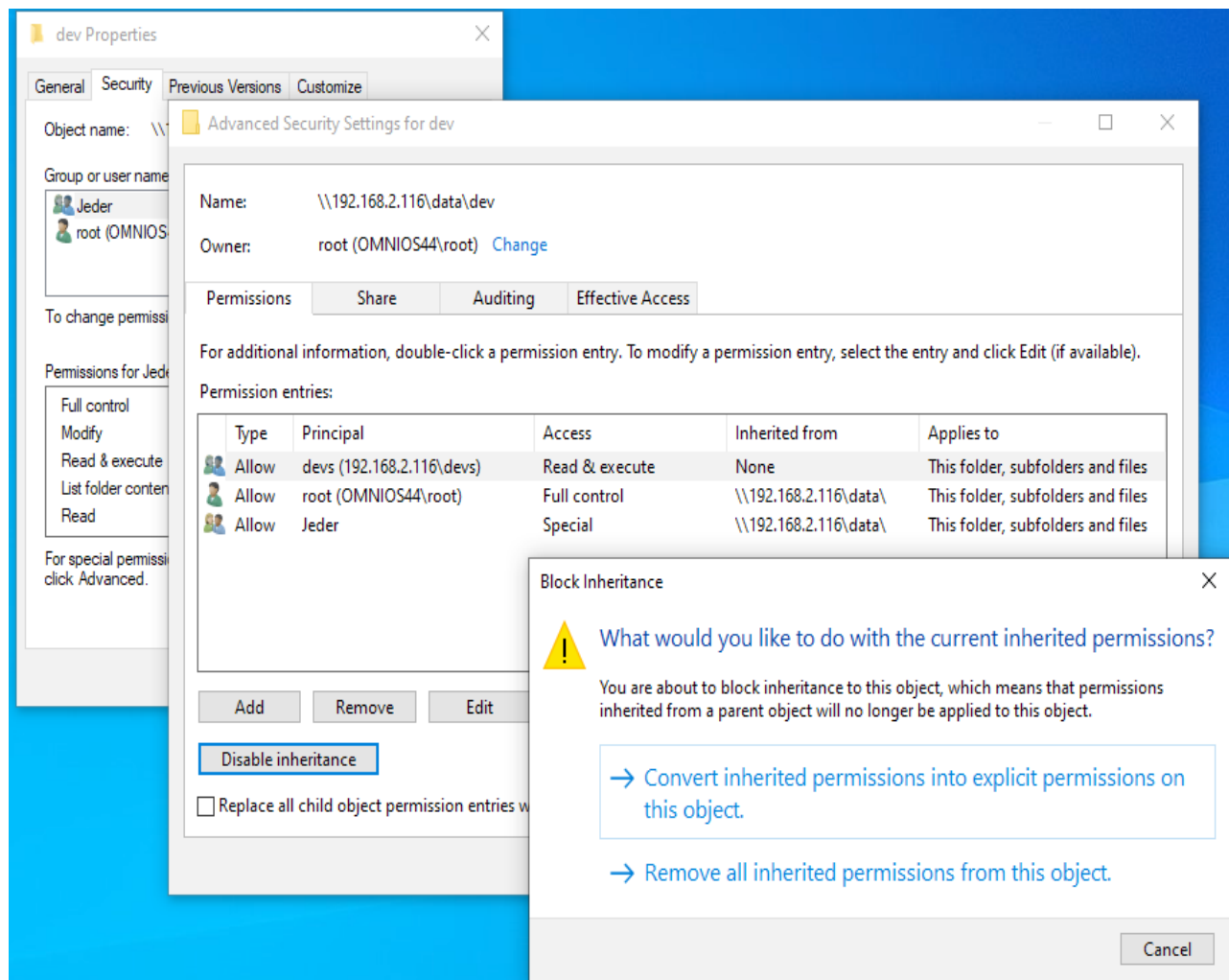
Basically it is ok to say they are quite identical regarding options. Both offer fine granular permission options with inheritance. Main difference is that Windows processes first deny then allow rules while Solarish respects order of rules where the first match defines access. Without deny rules you can mostly ignore differences.



ACL and inheritance options in Windows for a folder named 2022. If you select acl properties for a file instead a folder inheritance and folder options are missing. Compared to the classic Unix permissions „read, write, execute“ fine granular ntfs/nfs4 ACL are far superiour especially when combined with inheritance. I cannot think about complex permission restrictions in a larger organisation without ACL and especially inheritance.

2.2 The inheritance mystery

ACL inheritance is the real killer feature of a modern storage server. Without inheritance permission settings in a filesystem with many subfolders and different access options for different user/ groups can be a real mess. With inheritance you start at the toplevel folder and assign permissions to that folder. You can decide if the rule applies only to this folder or also to deeper objects like folders or files. The same applies to subfolders but there you can decide to disable further inheritance from upper folders with the option to copy or remove inherited permissions.



In this screen you see settings for folder /data/devs where the first ACL is a explicit entry for this folder and the other two are inherited from toplevel folder. If you click on „Disable inheritance“ you get the option to convert inherited settings to explicit settings for this folder or to remove them.

To restore inheritance settings for this sub folder, goto toplevel folder security settings and select „Replace all child object permission entries with inheritable entries from this object“.

2.3 Share ACL

Usually you assign permissions individually to files and folders. For additional global or temporary restrictions you can use share ACL. These are ACL to a share control file /pool/filesstem/.zfs/shares/filesystem. This file is created when you enable a share and deleted when you disable a share. This is why share ACL are normally not persistent. Napp-it store share ACL as additional ZFS properties to allow a restore when you re-enable a share.

Set share ACL in napp-it menu ZFS filesystems > ACL on shares

3. Setup basic ACLs

3.1 Set ACL for the filesystem itself (toplevel of a share) via Windows Properties > Security (SMB connect as root)

Start with a setting that allows root full access:
add root with full access and inheritance to files and folders.

If you want to cleanup a filesystem with ACLs, use „Replace all child object permission entries with inheritable entries from this object“. You can also set ACL recursively in napp-it menu ZFS filesystems > folder ACL > reset ACL ex root=full. Now only root has full access to the share and all folders and files.

3.2 Set ACL for regular folders in a share

Next step is to allow everyone@ at least access to the toplevel of the share. Add read permissions for everyone@ (Solarish reads set) but restrict to this folder only. Now everyone can access the share and nothing else.

If there are already folders ex staff for whom you want restricted access for staff members, set ACL for this folder, add SMB group staff=modify with inheritance to files and folders. Now staff can work with this folder but cannot modify ACL. If you want additionally read access for SMB group „students“ add this group with read access and inheritance to files and folders. If there is a subfolder below staff ex research where you do not want staff, disable inheritance for this folder (delete entries) then add allow SMB group research with modify permissions and inheritance to files and folders to allow access to folders below research.

A special case is if you want allow creation of new files and folders (only). Add the ACL that allows file/ folder creation. The owner is the creator of the file. On Unix the owner and root always have full access. To make this visible in Windows you can add a permission that allows owner full access.

3.3 ACL and classic Unix permissions like 755

If you do a chmod like 755, you set an according explicit ACL for this folder and delete all inheritance settings as they are only available on modern ACLs. This is mostly unwanted so I would say, never use classic Unix permissions on Solaris, always use ACL.

3.4 ACL/SMB and Macs

Macs cannot see or modify ntfs/NFSv4 ACL but respect settings. You must use Windows or napp-it to set ACL. SMB problems on Macs are common, google ex „OSX problems smb“ and avoid some releases like 10.14. and try the offered workarounds in other releases. Up from OmniOS 151032 there are „Apple extensions“ for the kernelbased SMB server that you want to disable in some cases, see

<https://illumos.topicbox.com/groups/omnios-discuss/T889d4225f9982c49-M196e9bc4f2f1275b783f0949/problem-with-smb-mount-on-macos-smb2-aapl>

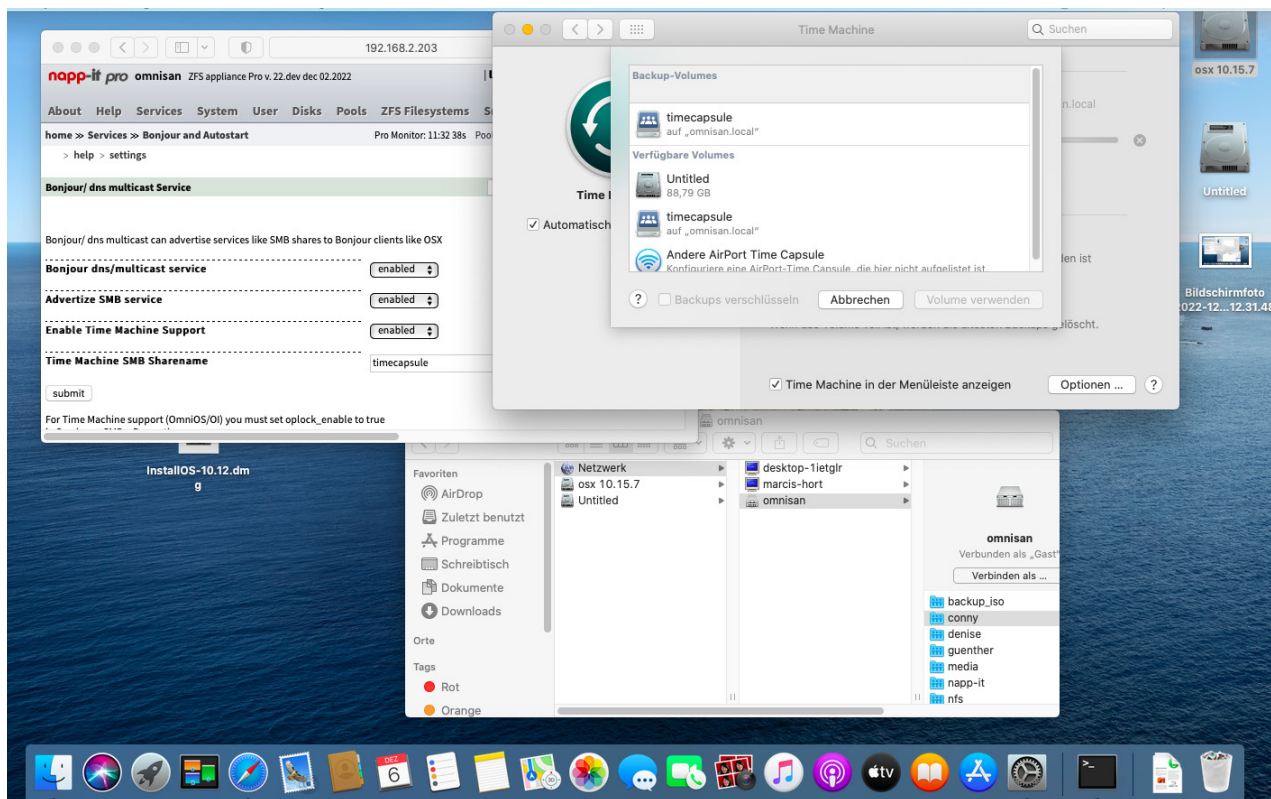
3.5 NFS and SMB

You often use NFS as a network filesystem ex for ESXi. You can share the same filesystem via SMB to allow ZFS snap access and restore via Windows previous version. Only problem is that NFS (v3) has no authentication or authorisation. NFS can only restrict access based on a client uid or a client ip, both can be faked. Depending on clients files are created with user nobody or the client uid. NFS should be only used in secure networks. To allow NFS access you should set an acl everyone@=full or modify with inheritance to files and folders. On NFS write problems reset recursively to this setting. For concurrent access with SMB set nbmand blocking setting to on (ZFS properties). To hinder NFS to modify ACL via chmod set aclmode to restricted (napp-it menu ZFS filesystems > Folder ACL)

4. OmniOS server for Apple clients

OS X and the Solaris based (OmniOS) multithreaded kernelbased SMB server

OmniOS is a very good and fast SMB server for Macs. It supports SMB from v.1 to v.3.1.1 with Apple extensions, Bonjour/mdns discovery in the local LAN segment and Time Machine via SMB. The OmniOS server is displayed with a nice Xserve icon.



OSX.png

If you use Macs with non Apple servers, you often find annoying incompatibilities as Apple introduces new features or remove others without caring for already available products. While Linux and Windows work well with a wide range of SMB services, OS X often does not. This is why OmniOS added Apple extensions to allow OS X features for Macs like Time Machine. On remaining problems, first google for solutions ex „OSX smb problem“. As you will find thousands of hits, reduce optionally for ex to „OS X 10.15 smb problem“ (or NFS).

In general it is strongly recommended to use a current OmniOS stable/LTS and an OSX releases under support. If you are on an older OSX, update to newest ex 10.15.7 Catalina (or newer if possible as 10.15 support ended October 2022, often possible for Macs from 2012 or newer <https://support.apple.com/en-us/HT210222>), as newer OSX versions often have less problems and it may be easier to find help. If your Mac is basically supported but About this Mac > Softwareupdate does not offer an update, download OS X 10.15 installer and start update manually ex from <https://support.apple.com/en-us/HT211683>

The usual way to SMB connect OmniOS is

Finder: Goto > Connect to server: smb://serverip (this connects SMB2/3)

If Bonjour/mdns is enabled, you find the shares also under network

Common problems and solutions.

On problems with smb://, try cifs

Finder: Goto > Connect to server: cifs://serverip (this connects SMB1)

Other fixes for SMB (NFS) problems on current Macs

Google „OSX SMB problem“ and restrict findings to last year.

Check/ask for Apple or OSX at <https://illumos.topicbox.com/latest> or <https://www.illumos.org/issues>

Apple SMB extensions

Mostly you are happy about them. On some (locking) problems you can disable

Put the following on a new file inside `/etc/system.d/` and reboot

```
set smb2_aapl_use_file_ids=1
```

If you ever want to disable completely the smb aapl mode, you can use:

```
set smb2_aapl_extensions=0
```

Other sharing options:

AFP

Finder: Goto > Connect to server: `afp://serverip` (netatalk is available via pkgin but obsolete now, end of live)

iSCSI

You need an initiator software on your Mac (not included in OS). An initiator uses a ZFS volume like a local disk

Google for `osx+initiator`

NFS

Finder: Goto > Connect to server: `nfs://serverip/pool/filesystem` (ex `nfs://192.168.1.1/tank/data`)

NFS problems

NFS (v3) does not offer authentication or authorization like SMB. Use it only on secure networks and prefer SMB. Access restrictions are possible based on fakeable client ip or on client uid (or user nobody). Normally you enable a NFS share with „on“ in the sharing dialog. If you instead enter „`root=@192.168.1.0/24`“ you have access from those clients with another uid. This is similar to Unix `no_root_squash` setting that maps root to nobody on NFS..

Set `nbmand` (ZFS property) to on when using a ZFS filesystem over SMB and NFS concurrently. Prefer an ACL like `everyone@=modify` for all files. For SMB you can additionally restrict access with Share ACL

Timemachine

On OmniOS:

- create a filesystem ex `timecapsule` with `smbshare=on`

Menu `napp-it Services > Bonjour and Autostart`

- enable `Bonjour/mdns`

- enable support for smb and Time Machine for this share `timecapsule`

OSX (Timemachine settings)

- select `timecapsule` as backup target

5. more manuals and infos

5.1 Solaris

https://docs.oracle.com/cd/E23824_01/html/821-1449/smbenvironmentoverview.html

<https://docs.oracle.com/en/operating-systems/solaris/oracle-solaris/11.4/manage-smb/managing-smb-file-sharing-and-windows-interoperability-oracle-solaris-11.4.pdf>

History: Sun Solaris CIFS Server

<http://nfsv4bat.org/Documents/ConnectAThon/2008/cifs-server.pdf>

5.2 Illumos/ OpenIndiana/ OmniOS/ older Solaris

https://docs.oracle.com/cd/E36784_01/html/E36832/smbservertasks.html

5.3 more manuals

<https://omnios.org/> (click on documentation, upper left)

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

Topicbox forum, newest features and questions (topicbox search ex for SMB or Apple)

look at illumos-dev, illumos-discuss and omnios-discuss

Illumos issues (bugtracker)

5.4 Forums

<https://gitter.im/omniosorg/Lobby> (OmniOS Lobby)

<https://forums.servethehome.com/index.php?forums/solaris-nexenta-openindiana-and-napp-it.26/>

<https://hardforum.com/threads/opensolaris-derived-zfs-nas-san-omnios-openindiana-solaris-and-napp-it.1573272/>

<https://www.hardwareluxx.de/community/threads/zfs-stammtisch.570052/>

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