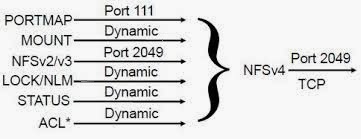
***Difference between NFS3 and NFS4***

Here I have listed some major differences between NFS3 and NFS4.Security has been major improvement taken into account for developing NFS4 version.Version NFS 4.1 is in development stage includes pNFS mechanism.

|  |  |  |
| --- | --- | --- |
|  | **NFSv3** | **NFSv4** |
| **Personality-Behavior** | Stateless protocol | Stateful protocol |
| **Semantics-support** | Support UNIX only | Support UNIX and Windows |
| **Authentication type** | Weak -(AUTH\_SYS) | Strong -(Kerberos) |
| **Identification type** | 32-bit UID/GID | String based (xyz@blogspot.com) |
| **Permissions type** | UNIX based access | Windows like access |
| **Transport protocol** | UDP and TCP | TCP only |
| **Caching data** | Ad-hoc | Delegations |

[](http://4.bp.blogspot.com/-ue0rBwYOtoo/UxNax2DMtKI/AAAAAAAAAiQ/c4Ymmo6cAwE/s1600/nfs4.jpg)

# A brief look at the difference between NFSv3 and NFSv4

There are a few interesting differences between NFSv3 and NFSv4. Comparison of  NFSv3 and NFSv4 is quite hard to obtain and the information is referenced from [NFS Version 4 Open Source Project.](http://www.citi.umich.edu/projects/nfsv4/OLS2001/tsld001.htm)

From a File System perspective, there are

**Export Management**

1. In NFSv3, client must rely on auxiliary protocol, the mount protocol to request a list of server’s exports and obtain root filehandle of a given export. It is fed into the NFS protocol proper once the root filehandle is obtained.
2. In NFSv4 uses the virtual file system to present the server’s export and associated root filehandles to the client.
3. NFSv4 defines a special operation to retrieve the Root filehandle and the NFS Server presents the appearance to the client that each export is just a directory in the pseudofs
4. NFSv4 Pseudo File System is supposed to provide maximum flexibility. Exports Pathname on servers can be changed transparently to clients.

**State**

1. NFSv3 is stateless. In other words if the server reboots, the clients can pick up where it left off. No state has been lost.
2. NFSv3 is typically used with NLM, an auxiliary protocol for file locking. NLM is stateful that the server LOCKD keeps track of locks.
3. In NFSv4, locking operations are part of the protocol
4. NFSv4 servers keep track of open files and delegations

**Blocking Locks**

1. NFSv3 rely on NLM. Basically, Client process is put to “sleep”. When a callback is received from the server, client process is granted the lock.
2. For NFSv4, the client to put to sleep, but will poll the server periodically for the lock.
3. The benefits of the mechanism is that there is one-way reachability from client to server. But it may be less efficient.