NFSv4.1 Server Implementation

Server support for minorversion 1 can be controlled using the /proc/fs/nfsd/versions control file. The string output returned by reading this file will contain either "+4.1" or "-4.1" correspondingly.

Currently, server support for minorversion 1 is enabled by default. It can be disabled at run time by writing the string "-4.1" to the /proc/fs/nfsd/versions control file. Note that to write this control file, the nfsd service must be taken down. You can use rpc.nfsd for this; see rpc.nfsd(8).

(Warning: older servers will interpret "+4.1" and "-4.1" as "+4" and "-4", respectively. Therefore, code meant to work on both new and old kernels must turn 4.1 on or off *before* turning support for version 4 on or off; rpc.nfsd does this correctly.)

The NFSv4 minorversion 1 (NFSv4.1) implementation in nfsd is based on RFC 5661.

From the many new features in NFSv4.1 the current implementation focuses on the mandatory-to-implement NFSv4.1 Sessions, providing "exactly once" semantics and better control and throttling of the resources allocated for each client.

The table below, taken from the NFSv4.1 document, lists the operations that are mandatory to implement (REQ), optional (OPT), and NFSv4.0 operations that are required not to implement (MNI) in minor version 1. The first column indicates the operations that are not supported yet by the linux server implementation.

The OPTIONAL features identified and their abbreviations are as follows:

- **pNFS** Parallel NFS
- **FDELG** File Delegations
- **DDELG** Directory Delegations

The following abbreviations indicate the linux server implementation status.

- **I** Implemented NFSv4.1 operations.
- **NS** Not Supported.
- **NS*** Unimplemented optional feature.

Operations

Implementation status	Operation	REQ,REC, OPT or NMI	Feature (REQ, REC or OPT)	Definition
	ACCESS	REQ		Section 18.1
I	BACKCHANNEL_CTL	REQ		Section 18.33
I	BIND_CONN_TO_SESSION	REQ		Section 18.34
	CLOSE	REQ		Section 18.2
	COMMIT	REQ		Section 18.3
	CREATE	REQ		Section 18.4
I	CREATE_SESSION	REQ		Section 18.36
NS*	DELEGPURGE	OPT	FDELG (REQ)	Section 18.5
	DELEGRETURN	OPT	FDELG,	Section 18.6
			DDELG, pNFS	
			(REQ)	
I	DESTROY_CLIENTID	REQ		Section 18.50
I	DESTROY_SESSION	REQ		Section 18.37
I	EXCHANGE_ID	REQ		Section 18.35
I	FREE_STATEID	REQ		Section 18.38
	GETATTR	REQ		Section 18.7
I	GETDEVICEINFO	OPT	pNFS (REQ)	Section 18.40
NS*	GETDEVICELIST	OPT	pNFS (OPT)	Section 18.41
	GETFH	REQ		Section 18.8
NS*	GET_DIR_DELEGATION	OPT	DDELG (REQ)	Section 18.39
I	LAYOUTCOMMIT	OPT	pNFS (REQ)	Section 18.42
I	LAYOUTGET	OPT	pNFS (REQ)	Section 18.43
I	LAYOUTRETURN	OPT	pNFS (REQ)	Section 18.44
	LINK	OPT		Section 18.9
	LOCK	REQ		Section 18.10
	LOCKT	REQ		Section 18.11
	LOCKU	REQ		Section 18.12
	LOOKUP	REQ		Section 18.13
	LOOKUPP	REQ		Section 18.14

Implementation status	Operation	REQ,REC, OPT or NMI	Feature (REQ, REC or OPT)	Definition
	NVERIFY	REQ		Section 18.15
	OPEN	REQ		Section 18.16
NS*	OPENATTR	OPT		Section 18.17
	OPEN_CONFIRM	MNI		N/A
	OPEN_DOWNGRADE	REQ		Section 18.18
	PUTFH	REQ		Section 18.19
	PUTPUBFH	REQ		Section 18.20
	PUTROOTFH	REQ		Section 18.21
	READ	REQ		Section 18.22
	READDIR	REQ		Section 18.23
	READLINK	OPT		Section 18.24
	RECLAIM_COMPLETE	REQ		Section 18.51
	RELEASE_LOCKOWNER	MNI		N/A
	REMOVE	REQ		Section 18.25
	RENAME	REQ		Section 18.26
	RENEW	MNI		N/A
	RESTOREFH	REQ		Section 18.27
	SAVEFH	REQ		Section 18.28
	SECINFO	REQ		Section 18.29
I	SECINFO_NO_NAME	REC	pNFS files	Section 18.45,
			layout (REQ)	Section 13.12
I	SEQUENCE	REQ		Section 18.46
	SETATTR	REQ		Section 18.30
	SETCLIENTID	MNI		N/A
	SETCLIENTID_CONFIRM	MNI		N/A
NS	SET_SSV	REQ		Section 18.47
I	TEST_STATEID	REQ		Section 18.48
	VERIFY	REQ		Section 18.31
NS*	WANT_DELEGATION	OPT	FDELG (OPT)	Section 18.49
	WRITE	REQ		Section 18.32

Callback Operations

Implementation status	Operation	REQ,REC, OPT or NMI	Feature (REQ, REC or OPT)	Definition
	CB_GETATTR	OPT	FDELG (REQ)	Section 20.1
I	CB_LAYOUTRECALL	OPT	pNFS (REQ)	Section 20.3
NS*	CB_NOTIFY	OPT	DDELG (REQ)	Section 20.4
NS*	CB_NOTIFY_DEVICEID	OPT	pNFS (OPT)	Section 20.12
NS*	CB_NOTIFY_LOCK	OPT		Section 20.11
NS*	CB_PUSH_DELEG	OPT	FDELG (OPT)	Section 20.5
	CB_RECALL	OPT	FDELG,	Section 20.2
			DDELG, pNFS	
			(REQ)	
NS*	CB_RECALL_ANY	OPT	FDELG,	Section 20.6
			DDELG, pNFS	
			(REQ)	
NS	CB_RECALL_SLOT	REQ		Section 20.8
NS*	CB_RECALLABLE_OBJ_AVAIL	OPT	DDELG, pNFS	Section 20.7
			(REQ)	
I	CB_SEQUENCE	OPT	FDELG,	Section 20.9
			DDELG, pNFS	
			(REQ)	
NS*	CB_WANTS_CANCELLED	OPT	FDELG,	Section 20.10
			DDELG, pNFS	
			(REQ)	

Implementation notes:

SSV:
: The spec claims this is mandatory, but we don't actually know of any implementations, so we're ignoring it for now. The server returns NFS4ERR_ENCR_ALG_UNSUPP on EXCHANGE_ID, which should be future-proof.

GSS on the backchannel:
: Again, theoretically required but not widely implemented (in particular, the current Linux client doesn't request it). We return NFS4ERR_ENCR_ALG_UNSUPP on CREATE_SESSION.

DELEGPURGE:
: mandatory only for servers that support CLAIM_DELEGATE_PREV and/or CLAIM_DELEG_PREV_FH (which allows clients to keep delegations that persist across client reboots). Thus we need not implement this for now.

EXCHANGE_ID:
: implementation ids are ignored

CREATE_SESSION:
: backchannel attributes are ignored

SEQUENCE:
: no support for dynamic slot table renegotiation (optional)

Nonstandard compound limitations:
: No support for a sessions fore channel RPC compound that requires both a ca_maxrequestsize request and a ca_maxresponsesize reply, so we may fail to live up to the promise we made in CREATE_SESSION fore channel negotiation.

See also http://wiki.linux-nfs.org/wiki/index.php/Server_4.0_and_4.1_issues.