

Microsoft Networks
SMB FILE SHARING PROTOCOL

Document Version 6.0p

January 1, 1996
Microsoft Corporation

This document is an early release of the final specification. It is meant to specify and accompany software that is still in development. Some of the information in this documentation may be inaccurate or may not be an accurate representation of the functionality of the final specification or software. Microsoft assumes no responsibility for any damages that might occur either directly or indirectly from these inaccuracies. Microsoft may have trademarks, copyrights, patents or pending patent applications, or other intellectual property rights covering subject matter in this document. The furnishing of this document does not give you a license to these trademarks, copyrights, patents, or other intellectual property rights

LG Electronics, Inc. et al.

Introduction	5
Resource Sharing Connections	5
Message Format	7
Sample Messge Flow	8
SMB Protocol Dialects	8
Message Transport	9
Reliable NetBIOS Transports	9
Connectionless IPX Transport	10
Naming On Ipx	12
Opportunistic Locks	12
Exclusive Oplocks	13
Batch Oplocks	14
Level II Oplocks	15
NAMED PIPES	15
Named Pipe Features	16
SMB MESSAGES AND FORMATS	16
SMB Header	16
Flags field	17
Flags2 Field	17
Tid Field	18
Pid Field	18
Mid Field	19
Status Field	19
Timeouts	19
Data Buffer (<i>Buffer</i>) and String Formats	19
Time And Date Encoding	20
Access Mode Encoding	20
File Attribute Encoding	21
“ANDX” SMB Messages	21
SMB MESSAGES	22
Valid SMB Messages by Negotiated Dialect	22
NEGOTIATE: Negotiate Protocol	23
SESSION_SETUP_ANDX: Session Setup And X	26
LOGOFF_ANDX: User Logoff And X	29
TREE_CONNECT: Tree Connect	29
TREE_CONNECT_ANDX: Tree Connect And X	30
TREE_DISCONNECT: Tree Disconnect	31
CREATE_DIRECTORY: Create Directory	32
DELETE_DIRECTORY: Delete Directory	32

CHECK_DIRECTORY: Check Directory	32
OPEN: Open File	33
CREATE: Create File	34
CLOSE: Close File	35
FLUSH: Flush File	35
DELETE: Delete File	35
RENAME: Rename File	36
QUERY_INFORMATION: Get File Attributes	37
SET_INFORMATION: Set File Attributes	37
READ: Read File	37
WRITE: Write Bytes	38
LOCK_BYTE_RANGE: Lock Bytes	39
UNLOCK_BYTE_RANGE: Unlock Bytes	40
CREATE_TEMPORARY: Create Temporary File	40
CREATE_NEW: Create File	40
PROCESS_EXIT: Process Exit	41
SEEK: Seek in File	41
SMB_QUERY_INFORMATION_DISK: Get Disk Attributes	42
SEARCH: Search Directory	43
OPEN_PRINT_FILE: Create Print Spool file	44
WRITE_PRINT_FILE: Write to Print File	45
CLOSE_PRINT_FILE: Close and Spool Print Job	45
GET_PRINT_QUEUE: Get Printer Queue Entries	46
LOCK_AND_READ: Lock and Read Bytes	47
WRITE_AND_UNLOCK: Write Bytes and Unlock Range	47
READ_RAW: Read Raw	48
READ_MPX: Read Block Multiplex	50
WRITE_RAW: Write Raw Bytes	51
WRITE_MPX: Write Block Multiplex	54
SET_INFORMATION2: Set File Information	55
QUERY_INFORMATION2: Get File Information	56
LOCKING_ANDX: Lock or UnLock Bytes	56
MOVE: Rename File	59
COPY: Copy File	60
ECHO: Ping the Server	61
WRITE_AND_CLOSE: Write Bytes and Close File	61
OPEN_ANDX: Open File And X	63
NT_CREATE_ANDX: Create File	65
READ_ANDX: Read Data	66
WRITE_ANDX: Write Bytes to file or resource	67
TRANSACTIONS	69
SMB_COM_TRANSACTION and SMB_COM_TRANSACTION2 Formats	69
SMB_COM_NT_TRANSACTION Formats	71
Functional Description	72
SMB_COM_TRANSACTION Operations	75
Mail Slot Transaction Protocol	75
Named Pipe Transaction Protocol	75
CallNamedPipe	76
WaitNamedPipe	76
PeekNamedPipe	76
GetNamedPipeHandleState	77
SetNamedPipeHandleState	77
GetNamedPipeInfo	78
TransactNamedPipe	78
RawReadNamedPipe	79

RawWriteNamedPipe	79
SMB_COM_TRANSACTION2 Operations	79
TRANS2_OPEN2	80
TRANS2_FIND_FIRST2	82
SMB_INFO_STANDARD	84
SMB_INFO_QUERY_EA_SIZE	84
SMB_INFO_QUERY_EAS_FROM_LIST	84
SMB_FIND_FILE_DIRECTORY_INFO	85
SMB_FIND_FILE_FULL_DIRECTORY_INFO	85
SMB_FIND_FILE_BOTH_DIRECTORY_INFO	85
SMB_FIND_FILE_NAMES_INFO	85
TRANS2_FIND_NEXT2	86
TRANS2_QUERY_FS_INFORMATION	86
SMB_INFO_ALLOCATION	88
SMB_INFO_VOLUME	88
TRANS2_QUERY_PATH_INFORMATION	88
SMB_INFO_STANDARD & SMB_INFO_QUERY_EA_SIZE	88
SMB_INFO_QUERY_EAS_FROM_LIST & SMB_INFO_QUERY_ALL_EAS	89
SMB_INFO_IS_NAME_VALID	89
TRANS2_SET_PATH_INFORMATION	89
SMB_INFO_STANDARD & SMB_INFO_QUERY_EA_SIZE	90
SMB_INFO_QUERY_ALL_EAS	90
TRANS2_QUERY_FILE_INFORMATION	90
TRANS2_SET_FILE_INFORMATION	90
TRANS2_CREATE_DIRECTORY	91
SMB_COM_NT_TRANSACTION Operations	91
NT_TRANSACT_CREATE	92
NT_TRANSACT_IOCTL	92
NT_TRANSACT_SET_SECURITY_DESC	93
NT_TRANSACT_NOTIFY_CHANGE	93
NT_TRANSACT_QUERY_SECURITY_DESC	94
NT_CANCEL: Cancel request	94
FIND_CLOSE2: Close Search	94

SMB COMMAND CODES **95**

ERROR CODES AND CLASSES **96**

Introduction

This document describes the Lan Manager Server Message Block (SMB) file sharing protocol. Client systems use this protocol to request file, print, and communications service from server systems over a network.

There are several different versions and sub-versions of this protocol, a particular version is referred to as a *dialect*. When two machines first come into network contact they negotiate the dialect to be used. For example, two NT systems would agree to use the NT-specific protocol dialect, while a Windows For Workgroups client communicating with an NT server might negotiate a Windows For Workgroups dialect. Different dialects can include both new messages as well as changes to the fields and semantics of existing messages in other dialects.

Resource Sharing Connections

Each server makes a set of resources available to clients on the network. A resource being shared may be a directory tree, named pipe, printer, etc. So far as clients are concerned, the server has no storage or service dependencies on any other servers; a client considers the server to be the sole provider of the file (or other resource) being accessed.

The SMB protocol requires server authentication of users before file accesses are allowed, and each server authenticates its own users. A client system must send authentication information to the server before the server will allow access to its resources.

The SMB protocol defines two methods which can be selected by the server for security: *share level* and *user level*:

- A *share level* server makes some directory on a disk device (or other resource) available. An optional password may be required to gain access. Thus any user on the network who knows the name of the server, the name of the resource and the password has access to the resource. Share level security servers may use different passwords for the same shared resource with different passwords allowing different levels of access. Windows for Workgroups and Windows 95 servers, for instance, implement the share level security model.
- A *user level* server makes some directory on a disk device (or other resource) available but in addition requires the client to provide a user name and corresponding user password to gain access. NT servers and LM/U servers implement this security model and do not support the *share level* model. User level servers are preferred over share level servers for any new server implementation, since corporations generally find *user level* servers easier to administer as employees come and go.

When a *user level* server validates the account name and password presented by the client, an identifier representing that authenticated instance of the user is returned to the client in the *Uid* field of the response SMB. This *Uid* must be included in all further requests made on behalf of the user from that client. A *share level* server returns no useful information in the *Uid* field.

The user level security model was added after the original dialect of the SMB protocol was issued, and subsequently some clients may not be capable of sending account name and passwords to the server. A server in user level security mode communicating with one of these clients will allow a client to connect to resources even if the client has not sent account name and password information:

1. If the client's computer name is identical to an account-name known on the server, and if the password supplied to connect to the shared resource matches that account's password, an implicit "user logon" will be performed using those values.

If the above fails, the server may fail the request or assign a default account name of its choice.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.