INTERNET-DRAFT
draft-crhertel-smb-url-06.txt
Expires July 8, 2004

Christopher R. Hertel
Samba Team

January 8, 2004

SMB File Sharing URI Scheme

Status of this Memo

Abstract

The Server Message Block (SMB) protocol is one of the most widely used network file system protocols in existence. This document describes a format for an SMB Uniform Resource Indicator (SMB URI). The SMB URI can be used to indicate SMB workgroups, servers, shares, files, inter-process communications pipes, print queues, and devices; the objects in the SMB network file system space.

Table of Contents

1. Introduction

The Server Message Block protocol (SMB) was created in the 1980's by Dr. Barry Feigenbaum at IBM Corporation. It was later extended by various contributors at 3Com, IBM, Intel, and Microsoft. During the mid 1990's SMB was renamed CIFS (for Common Internet File System). Both names are in use today.

SMB was originally carried via a proprietary network transport, the interface to which was called NetBIOS (Network Basic Input Output System). Two Internet RFCs ([RFC1001], [RFC1002]) were published which describe a mechanism for implementing the NetBIOS API on top of TCP and UDP, thus allowing SMB to be carried over IP internetworks. Those RFCs are now known collectively as Internet Standard #19 (STD 19), and the transport protocol that they describe

is commonly called NBT (or, sometimes, NetBT) for NetBIOS over TCP.

In addition to transport via NBT, newer implementations of SMB typically support SMB over TCP/IP without the intervening NetBIOS emulation layer.

Several attempts have been made to document and even standardize the SMB protocol ([XOPENSMB], [ONET], [SNIACIFS], [IMPCIFS]), yet the further development of SMB remains under the control of Microsoft. Despite its proprietary nature, the workings of SMB are sufficiently well known that SMB file sharing has been successfully implemented by several third-party commercial vendors and in Open Source. SMB server and client software is available for a wide variety of operating system platforms. The very large number of systems which support this form of file sharing make an SMB URI scheme both practical and desirable.

1.1. Purpose

This document does not attempt to describe the implementation of SMB, or the workings of NBT. The goal is to present the syntax of the SMB URI, and to describe how it is mapped to the semantics of the SMB protocol suite. The appendices do contain a limited set of implementation guidelines.

2. Working with NBT Transport

The NBT transport maps the semantics of the NetBIOS API onto TCP and UDP. In order to make this work, NBT defines three services:

- The NetBIOS Name Service
- The NetBIOS Datagram Service
- The NetBIOS Session Service

2.1. The NetBIOS Name Service

NetBIOS uses names to identify communications endpoints across the network. NetBIOS names are the addresses used in a "NetBIOS network". The NetBIOS Name Service is responsible for transparently mapping these names to IP addresses so that NetBIOS applications and services can find one another. In essence, the Name Service is used to create virtual "NetBIOS LANs" on top of IP internetworks.

The set of nodes that are connected to the same virtual NetBIOS LAN is known as the "scope" of the LAN. NBT scopes are always given a name, or Scope Identifier, though this normally goes unnoticed because the default Scope ID is the empty string (""). Multiple scopes, each having a different Scope ID, may intersect across the same IP networks without conflict.

2.2. The NetBIOS Datagram Service

The purpose of the Datagram Service is to ensure that NetBIOS datagrams may be sent anywhere within the scope of the virtual NetBIOS LAN. NetBIOS datagrams are mapped to UDP datagrams, and NBT nodes listen for datagrams on port 138 by default.

The SMB URI does not make explicit use of the Datagram Service.

2.3. The NetBIOS Session Service

The Session Service maps NetBIOS Sessions to TCP sessions. Multiple NetBIOS sessions may be carried over a single TCP connection.

SMB file sharing over NBT transport is carried by the Session Service. It is worth noting that the protocol for transport of SMB messages over NBT is nearly identical to transport via native TCP. The differences are:

- The NBT Session Service uses TCP port 139 by default. Native TCP transport uses port 445.

- NBT transport expects a Session Request/Session Response exchange in order to initiate a new NetBIOS session over the Session Service (see [RFC1001], section 16.1.1). Native TCP transport does not expect this exchange (though some implementations allow the Session Request to be optional on both transports).

2.4. Accommodating NBT in the SMB URI

The SMB URI scheme supports both NBT and native TCP transport of SMB. The syntax of the scheme provides for the inclusion of NBT context information so that NetBIOS names can be properly resolved and NetBIOS sessions established.

The scheme provides the ability to specify the following NBT context information:

- Broadcast Address for NBT broadcast name resolution
- NBNS Server Address for point-to-point name resolution
- Name resolution mode
- NBT Scope ID
- NetBIOS CALLED name (destination address)
- NetBIOS CALLING name (source address)

NBT context information is appended to the tail end of an SMB URI string in the form of a query. Context information is specified using key/value pairs. For example, the string:

```
smb://server/share/path/file.txt?NBNS=172.24.19.18
```

says that point-to-point name queries for the server name "server" should be sent to the NetBIOS Name Server at address 172.24.19.18.

Multiple context elements may be specified by separating the key/value pairs with semi-colons:

```
smb://server/share/path/file.txt?NBNS=172.24.19.18;NODETYPE=P
```

Most SMB implementations use configuration files or DHCP to establish an initial NBT context. The starting context is referred to as the "base context" in the remainder of this document. NBT context information given in absolute URIs is applied against the base context to give the "current context". NBT context information given in relative URIs is applied against the current context to update the current context.

The current context is always used to interpret the meaning of a given URI string. A relative URI containing updated NBT context information will cause the resulting URI to be re-evaluated.

3. Accessing NBT Workgroup Information

The "workgroup" system is built on top of NBT and allows SMB file servers to be organized into groups. The goal is to make it easier to locate resources by categorizing them.

A list of member servers is maintained for each workgroup, as well as a list of all known workgroups. The combined list is known as the "browse list". A copy of the browse list may be obtained by sending a specific query via SMB.

The SMB URI provides syntax that indicates requests for subsets of the browse list. In particular, the form:

```
smb://
```

represents a request for the list of all known workgroups, and the form:

```
smb://smb_browse/
```

represents a request for the list of servers that are members of a particular workgroup.

A problem arises, however, because the syntax used for requesting the list of servers in a workgroup is indistinguishable from that of a request for the list of shares offered by an SMB file server. The two requests must be differentiated semantically. Consider the following example:

```
smb://corgi/
```

If the name "corgi" is a NetBIOS name and it resolves to a workgroup name then a user agent would return a list of servers in the CORGI workgroup. Otherwise, the user agent would return a list of file shares offered by the server named CORGI.

It is rare, but possible (in a misconfigured NBT network), that a NetBIOS name will represent both a workgroup and an SMB file server. In this situation, SMB file services take precedence. Some user agents may be capable of returning both the list of servers in the workgroup and the list of shares provided by the SMB file server, and allowing the user to determine which is correct.

4. SMB File Sharing Hierarchy

The SMB URI scheme views the SMB file sharing environment hierarchically. Conceptually, the hierarchy is arranged as follows:

```
smb://                      Known workgroups
smb://smb_browse/             + SMB servers within a workgroup
smb://smb_server/               + Shares offered by an SMB server
smb://smb_server/abs_path         + Directories, files, etc.
```

There is a special case to be considered when moving between a workgroup reference and a reference to a server in the workgroup. Consider a workgroup named "corgis" and a server named "cue" that is a member of that workgroup.

Presented with the URI string

```
smb://corgis/
```

a user agent may return a list of servers that are members of the CORGIS workgroup, including node CUE, and allow the user to select one of those SMB servers. The relative reference from

```
smb://corgis/
```

to

```
smb://cue/
```

would be "../cue/". It may be simpler, however, for the user agent to provide absolute references to the workgroup member servers.

When moving upward in the hierarchy, one might might expect:

"smb://cue/" + ".." ==> "smb://"

but in this example node "cue" is a member of the "corgis" workgroup, so:

"smb://cue/" + ".." ==> "smb://corgis/"

The NBT workgroup membership of an SMB server may be determined either by sending a Node Status Request query to the server (see [RFC1001], section 15.1.4) or by maintaining a local cache of workgroup information, or both. Obviously, the choice is implementation dependent. If the server's workgroup membership is not available via either of these methods, then it is acceptable to move directly to the top of the hierarchy (smb://).

5. SMB URI Definition

The following grammar defines the syntax of the SMB URI. It is based upon the grammar given in Appendix A of [RFC2396], and amended by [RFC2732]. Refer to those RFCs (or later RFCs that supercede them) for token definitions missing from the grammar below.

```
smb_URI        = ( smb_absURI | smb_relURI )
smb_absURI     = scheme "://" smb_service [ "?" [ nbt_context ] ]
smb_relURI     = abs_path | rel_path

scheme         = "smb" | "cifs"
smb_service    = ( smb_browse | smb_net_path )

smb_browse     = [ smb_userinfo "@" ] [ smb_srv_name ]
                   [ ":" port ] [ "/" ]
smb_net_path   = smb_server [ abs_path ]

smb_server     = [ smb_userinfo "@" ] smb_srv_name [ ":" port ]

smb_srv_name   = nbt_name | host
nbt_name       = netbiosname [ "." scope_id ]
netbiosname    = 1*( netbiosnamec ) *( netbiosnamec | "*" )
netbiosnamec   = ( alphanum | escaped  | ":" | "=" | "+" | "$" |
                   "," | "-" | "_" | "!" | "~" | "'" | "(" | ")" )
scope_id       = domainlabel *( "." domainlabel )

smb_userinfo   = [ auth_domain ";" ] userinfo_nosem
auth_domain    = smb_srv_name
userinfo_nosem = *( unreserved | escaped |
                   ":" | "&" | "=" | "+" | "$" | "," )
```

```
      nbt_context   = nbt_param *(";" nbt_param )
```

```
nbt_param     = ( "BROADCAST=" IPv4address [ ":" port ]
                | "CALLED=" netbiosname
                | "CALLING=" netbiosname
                | ( "NBNS=" | "WINS=" ) host [ ":" port ]
                | "NODETYPE=" ("B" | "P" | "M" | "H")
                | ( "SCOPEID=" | "SCOPE=" ) scope_id
                )
```

6. SMB URI Syntax Elements

The SMB URI scheme is more or less comparable to other URI schemes used for remote filesystem access. It differs primarily in its support for the NBT transport and NBT workgroups. This section provides further explanation and description of those syntax elements that are most likely to require it.

6.1. scheme

An SMB URI is identified by one of two scheme names: "smb" or "cifs". These are equivalent. Support for both names is provided because both are in common usage today and because there are existing implementations of the SMB URI scheme that support one or the other or both.

New and updated implementations must support both scheme names in order to be compatible with existing SMB URI references.

As of this writing, the "smb" prefix appears to be the more popular of the two.

6.2. smb_service

The SMB URI can be used to access workgroup information or SMB file server services. There are minor differences in SMB URI syntax depending up on which of these service types is being accessed.

It is possible, for instance, to request workgroup information without specifying a destination server name. In particular, the URI:

```
smb://
```

represents a request for the list of locally available workgroups.

In some situations the workgroup list may not be available to unauthenticated users, so the SMB URI scheme allows inclusion of smb_userinfo information without the need to specify an smb_srv_name (a workgroup name). Thus, the following is permitted:

smb://neko@/

In the above, the username "neko" is being supplied. (The user agent should prompt for a password to prevent the password from being displayed.)

As with the smb_userinfo field, an SMB URI may include a port reference without an smb_srv_name, as in the following example:

smb://:4220/

This is an example of an attempt to retrieve an NBT workgroup list via SMB using destination TCP port 4220.

Another difference between workgroup and SMB file server references is that a workgroup reference may not be followed by a path. The browse list does not offer shares, directories, or files so an SMB URI string such as the following cannot represent a workgroup query:

smb://corgis/puppies/

6.3. auth_domain

The auth_domain field is passed to the underlying SMB layer for interpretation. It is used to specify the SMB authentication authority, (typically a "Domain Controller"). The interpretation of this field is specific to the workings of the SMB protocol and should be handled by the underlying SMB implementation.

6.4. smb_srv_name

The SMB URI supports the use of NetBIOS names and Scope IDs to identify SMB servers and services. When included as part of an SMB URI, the syntax of the NetBIOS name is a superset of the syntax of a DNS domain name label. For example:

smb://jcifs/

Syntactically, the string "jcifs" in the smb_srv_name field of the above string may be seen as either a DNS host name (unqualified), or as a NetBIOS name. The underlying SMB implementation must determine the namespace of the name. (This is a common problem in SMB implementations and is typically solved by first attempting to

resolve the name as a NetBIOS name and then, if that fails, as a DNS

host name.)

Likewise, given:

smb://jcifs.samba.org/

the string "jcifs.samba.org" may be interpreted either as a qualified DNS name, or as a NetBIOS name with appended Scope ID.

A NetBIOS name is simply a string of octets with a maximum length of 15 octets. (The actual maximum length of the NetBIOS name is 16-octets, but the 16th is reserved.) In practice, the only restriction on the syntax of a NetBIOS name is that it may not begin with an ASCII asterisk character (0x2A). Octet values that are permitted by NetBIOS name syntax but excluded by the SMB URI syntax must be escaped. Note, in particular, that the dot character (0x2E) must be escaped if used in a NetBIOS name.

The resolution of NetBIOS names to IP addresses is described in [RFC1001] and [RFC1002].

6.5. port

STD 19 includes a mechanism for retargeting Session Service connections to alternate ports (see [RFC1001], section 16.1.1.) which means that non-standard ports may be used for SMB over NBT transport. There may be other valid reasons for providing SMB services on on-standard ports.

The URI port field may be used to specify an alternate service port for SMB over either NBT or native TCP transport. (The transport type must be detected by the underlying SMB implementation.)

6.6. scope_id

The SMB URI scheme provides two mechanisms for specifying an NBT Scope ID. The first, as shown in the grammar above, is to append the Scope ID to the NetBIOS name as part of the smb_srv_name field, using a dot (".") as a delimiter. This mechanism is included to support existing implementations.

The other mechanism is to include the Scope ID as part of the nbt_context (which will be described shortly). The two examples given below are equivalent:

smb://netbios.scope.id/
smb://netbios/?SCOPE=scope.id

The latter format is less ambiguous and, therefore, preferred. User Agents that rewrite URI strings for display purposes should rewrite SMB URI strings that contain a Scope ID to conform to the preferred format.

6.7. nbt_context

The nbt_context may be used to provide information about the NBT transport layer and related support servers. Information provided in the nbt_context overrides the current NBT context maintained by the user agent. The nbt_context is interpreted locally by the user agent.

The nbt_context is made up of zero or more nbt_params fields, which are specified as key/value pairs. For example:

```
smb://jcifs/?CALLED=VIRTSERV;NBNS=172.24.19.18
```

In the above example, the CALLED parameter is assigned a value of "VIRTSERV", and the NBNS parameter is assigned a value of "172.24.19.18".

The following keywords are defined:

BROADCAST: The IPv4 broadcast address to which to send NBT broadcast name queries. This may, for example, be used on multi-homed hosts to specify a target subnet.

The value assigned to the BROADCAST keyword may optionally include a port number (delimited by a colon). The default port for NBT name resolution is UDP/137. It is rare that a different port will be used for broadcast name resolution.

CALLED: Specifies the NetBIOS name of the SMB server (the NetBIOS destination address.) A CALLED name is required by the NBT Session Request message (see [RFC1002], Section 4.3.2).

If NBT transport is used and the CALLED name is not specified within the URI string, the underlying SMB implementation must deduce the CALLED name from available information. (See Appendix A, below.)

CALLING: Specifies the NetBIOS name of the client (the NetBIOS source address.) This value is only used with NBT transport. It is required by the NBT Session Request message (see [RFC1002], Section 4.3.2).

If NBT transport is used and the CALLING name is not specified in the current NBT context, the underlying SMB implementation must generate a suitable name. (Typically, this will be the system's host name.)

NBNS: Specifies the NetBIOS Name Server (NBNS) to be used for point-to-point NBT Name Resolution. The NBNS may be specified using a DNS name or an IP address. See [RFC1001] for information on the NBNS.

The value assigned to this parameter may, optionally, include a port number (delimited by a colon). The default port for NBT name resolution is UDP/137. Use of a non-standard port for point-to-point NBT name resolution is rare, but less so than it is for broadcast name resolution.

NODETYPE: One of B, P, M, H, or the empty string. These represent the different mechanisms by which a NetBIOS name may be resolved to an IP address on an NBT network. The first three types are defined in STD 19. H mode is the inverse of M mode (in H mode the NBNS is queried before a broadcast query is sent). An empty NODETYPE indicates that NBT name resolution should not be attempted (use DNS name resolution instead). Some examples:

```
smb://smedley/?NBNS=172.24.19.18;NODETYPE=H
smb://corgis/?NODETYPE=B
smb://jcifs.samba.org/?NODETYPE=;CALLED=SMBSERV
```

SCOPE: Specifies the NBT Scope Identifier. Use of the SCOPE keyword is preferred over inclusion of the Scope ID in the nbt_name field. User agents must support both mechanisms, however.

The default Scope ID is the empty string. This can be specified in the SMB URI by assigning an empty value to the SCOPE keyword. For example:

```
smb://bran/SCOPE=
smb://marika/SCOPE=;NODETYPE=B
```

SCOPEID: A synonym for SCOPE.

WINS: A synonym for NBNS.

Although all of the keywords and values are shown in upper case, case is not significant.

The client implementation should provide a means for setting the base

context. The nbt_context is used to override default values or to supply values missing from the local configuration. Most of all, the nbt_context makes it possible for an SMB URI string to maintain a consistent interpretation as it travels from one NBT scope to another.

7. The Relationship Between the SMB URI and the UNC Format

Some operating systems support a format known as Universal Naming Convention (UNC). UNC is a means for identifying network resources. SMB is one of the protocols supported by UNC.

In general, a UNC string specifying a resource available via SMB protocol can be converted into an SMB URI string by simply adding the "smb:" or "cifs:" prefix and reversing the direction of all of the separating slashes. For example:

UNC form	URI form
\\corgis\docs\	smb://corgis/docs/
\\corgis\docs\jolyon\	smb://corgis/docs/jolyon/
\\corgis\docs\jolyon\rabbit.txt	smb://corgis/docs/jolyon/rabbit.txt

8. Authentication and Security Considerations

SMB authentication can be divided into the following categories:

- None
- Share-based
- User-based
- Authentication Server-based (NT Domain and Kerberos)

The authentication mechanism to be used is negotiated during client/server session setup. Client applications, therefore, are aware of the server's authentication requirements and may prompt for appropriate input (password, username, authentication domain). By prompting for authentication information, an application ensures that such information is entered by the user in a controlled manner, and that security measures (if any) such as password encryption or password hash generation are applied by the SMB protocol handler before the data are transmitted.

Some authentication values may also be provided within the SMB URI string. In particular, the following fields may optionally be included in the URI:

auth_domain - The authentication domain (single-signon database server) to use for authorization

userinfo - User account identifier (username)

9. Character Encoding Issues

The only restriction that STD 19 places on the octet values that may be used in a NetBIOS name is that the name may not begin with an asterisk ('*', ASCII value 0x2A). No other values are listed as excluded in the RFCs. For historical reasons, however, some implementations disallow the use of a nul byte (0x00) within a NetBIOS name. NetBIOS names are interpreted as a string of octets, so common mutli-byte character sets may cause problems with older implementations.

Octet values less than 128 (0x80) in a NetBIOS name are interpreted as US-ASCII characters. The interpretation of octet values above 127 are dependent upon host configuration; there is no protocol mechanism to specify which codepage or character set is in use. URI escape sequences should be used to represent characters with Octet values above 127.

NetBIOS names, share names, and the directory paths and filenames offered by an SMB server may all contain characters from outside the 7-bit US-ASCII character set. Applications MUST support the use of the URI escape sequence as described in [RFC2396] to accommodate octet values that represent non-US-ASCII characters.

The SMB protocol has evolved over time to include support for various character encoding schemes. A complete discussion of SMB and NBT character encoding issues is way beyond the scope of this document.

10. Acknowledgments

The creation of this document would not have been possible without the help and guidance of

Michael B. Allen
David Farmer
Roy T. Fielding
Steven French
Larry Masinter
Richard Sharpe

and the aggregate knowledge and wisdom of

The jCIFS Team
The Samba Team
The Samba-TNG Team
The SNIA CIFS Work Group
The samba-technical mailing list participants

The URI-review mailing list participants

11. References

[RFC1001] Karl Auerbach, et. al., "Protocol Standard For a NetBIOS Service on a TCP/UDP Transport: Concepts and Methods", RFC 1001, March 1987.

[RFC1002] Karl Auerbach, et. al., "Protocol Standard For a NetBIOS Service on a TCP/UDP Transport: Detailed Specifications", RFC 1002, March 1987.

[RFC2396] Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.

[RFC2732] R. Hinden, B. Carpenter, L. Masinter, "Format for Literal IPv6 Addresses in URL's", RFC 2732, December 1999.

[XOPENSMB] "Protocols for X/Open PC Interworking: SMB, Version 2", ISBN 1-872630-45-6, The Open Group, October 1992.

[ONET] Microsoft Corporation, Intel Corporation, "Microsoft Networks/OpenNET Filesharing Protocol", Document Version 2, Intel Part No. 138446, November 7, 1988.

[SNIACIFS] Storage Network Industry Association CIFS Documentation Work Group, "Common Internet File System (CIFS) Technical Reference", Version: CIFS-TR 1.0, March 1, 2002.

[IMPCIFS] Hertel, Christopher R., "Implementing CIFS -- the Common Internet File System", ISBN 0-13-047116-X, Prentice Hall PTR, August 2003 (http://ubiqx.org/cifs/)

12. Author's Address

Christopher R. Hertel
University of Minnesota
Networking and Telecommunications
2218 University Avenue SE
Minneapolis, MN 55414-3029, USA

E'mail: crh@samba.org
crh@ubiqx.mn.org

Appendix A. Working with NetBIOS Names (Implementation Notes)

The information presented in this section is intended as a guide for implementors.

Name resolution, particularly with the inclusion of support for RFC1001/1002 NBT naming, may result in ambiguous meaning for some SMB URI strings. This problem is reduced if correct NBT syntax information is included in URI strings, and can be eliminated if all implementations follow the same basic sequence when resolving server names to addresses.

A.1. NetBIOS Names

NetBIOS names are addresses. They represent communication end-points within a NetBIOS LAN. [RFC1001] and [RFC1002] provide a mechanism for creating virtual NetBIOS LANs over TCP and UDP transport. The core of that mechanism is the NetBIOS Name Service, which provides for mapping between NetBIOS names and IP addresses. A given host system may register several NetBIOS names, each representing an application or service that can communicate with other applications or services through the NetBIOS API.

A.2. SMB Sessions via NBT

SMB sessions are established and messages transferred via the NetBIOS session service (see [RFC[1001], section 5.3 and [RFC1002] section 4.3). The system that originates the connection is the "calling" node, and the target node is the "called" node. In order to establish an SMB session, a TCP connection must be established between the calling and called nodes. If a TCP connection already exists, the SMB session may make use of the existing connection.

Before a NetBIOS session can be established, the calling node must discover the IP address of the called node. This is done using the NetBIOS Name Service (see [RFC1001] section 5.2 and [RFC1002] section 4.2). NetBIOS names are always 16 octets, padded with spaces (0x20) if necessary, as specified in the RFCs. By convention, however, the 16th octet is reserved for use as a service type indicator. This field is known as the "suffix".

The suffix is NEVER specified in an SMB URI string, but is appended by the implementation.

A.3. Resolving DNS names and IP addresses to SMB server names

The NetBIOS session service requires that the client provide the NetBIOS names of both the calling and called nodes. When connecting to an SMB server, the calling name is the default NetBIOS name of the client, space padded as described, with a suffix byte value of 0x00. The called name is the NetBIOS name of the server with a suffix byte value of 0x20.

Applications which support the SMB URI must include support for the use of DNS names or IP addresses in addition to NetBIOS names when initiating SMB connections via NetBIOS over TCP/IP transport. This functionality is an extension to the NetBIOS over TCP/IP behavior specified in RFC 1001 and RFC 1002, and is not part of that standard. It is, however, a common extension and must be supported for compatibility reasons, and to provide access to SMB shares in situations in which the NetBIOS name space cannot be guaranteed to be consistent.

As stated above, the Session Request packet requires a called and a calling name, both of which are NetBIOS names. In order to create an NBT Session Request packet, the DNS name or IP address of the server must be reverse-mapped to the server's NetBIOS name. Mechanisms for doing so include:

- Issuing a NetBIOS Adapter Status Query

  A NetBIOS Adapter Status Query is sent to the target IP address. (See [RFC1001] section 15.1.4 and [RFC1002] sections 4.2.17 & 4.2.18.) If a response is received, and if the target node is running an SMB server service, then the response will include a NetBIOS name with a suffix byte value of 0x20. This NetBIOS name may be used as the called name in a Session Request packet.

  It is possible that multiple entries will have a suffix byte of 0x20. If this is the case each name may be tried in turn, or one of the other methods must be used to discover the name of the SMB server service.

- Generic Server Name

  This method is not supported by all SMB server implementations.

  Some SMB servers will accept the generic SMB server name "*SMBSERVER". A client can simply use the name "*SMBSERVER" as the called name in a Session Request packet. As with all SMB server NetBIOS names, the "*SMBSERVER" name must be space padded and terminated with a suffix byte value of 0x20.

  The "*SMBSERVER" begins with an asterisk character, so it is an

illegal NetBIOS name (see [RFC1001], section 5.2) and it is never

registered with the NetBIOS Name Service. It will not be returned in a NetBIOS Adapter Status Response.

If the target does not support the "*SMBSERVER" generic name, or if it is not running SMB services, it will return a CALLED NAME NOT PRESENT error.

Some SMB servers are capable of providing multiple SMB file services, each under a different NetBIOS name. In order to support the generic server name, these servers must designate one service as a default that will answer to "*SMBSERVER".

- Parsing the DNS Name or IP address (guessing)

  This is the least reliable method for discovering an SMB server name.

  Systems which support STD 19 transport will often use the same base host name within the DNS and NetBIOS name spaces. Thus, the first label of the DNS name is a good guess at the NetBIOS name of the target. If the input is an IP address rather than a DNS name, the a reverse lookup against the DNS may be performed to determine the DNS name.

  The first label of the DNS name consists of the initial portion of the DNS name string up to but not including the first dot character ('.'). If the label is greater than 15 bytes in length, it is cannot be a NetBIOS name. The label must be space padded to a total of 15 bytes, with a suffix value 0x20 added. This forms a valid NetBIOS name that may be used as a called name in a Session Request packet.

  If the target returns a CALLED NAME NOT PRESENT error, then the DNS name guess is incorrect.

Any of the above may be tried in any order.

A.4. Determining the Namespace of the smb_srv_name

NetBIOS names, DNS names, and IP addresses can not be easily distinguished syntactically. For example, the string "192.168.101.1" might be an IP address, but it is also a valid NetBIOS name and may even be a partially qualified DNS name. The appropriate mechanism for distinguishing between these server specifier types is the trial-and-error method.

Implementations should begin with the assumption that the specifier is a NetBIOS name. The following process is used to test this assumption:

If the NODETYPE is the empty string then no NetBIOS name resolution mechanism has been selected and the name cannot be resolved as a NetBIOS name. Exit.

If the name string contains dot characters ('.', ASCII 0x2E), then separate the name into NetBIOS name and Scope ID at the first dot. Otherwise use the entire string as the NetBIOS name, and assume an Scope ID of "".

REPEAT

If the resulting NetBIOS name is greater than 15 octets in length, then the name is not a NetBIOS name. Exit.

Issue an STD 19 Name Query using the NetBIOS name and Scope ID. A suffix value of 0x20, 0x1B, and/or 0x1D should be used. (See section A.5., below.)

If a Positive Name Query Response is received, then the name is a NetBIOS name. Exit, indicating success and returning the NetBIOS name and scope ID as parsed.

END

If the server specifier is not a NetBIOS name, then it is either a DNS name or an IP address. These are semantically equivalent.

A.5. Workgroup vs. SMB Server Names

If the URI string is of the form

```
smb://smb_srv_name/
```

then the smb_srv_name may represent either an SMB server name or a workgroup name. The name MUST NOT be interpreted as a workgroup name if:

- There is path information following the trailing slash.

  Workgroups do not make shares or directories available.

- The server field is entered as a DNS name or an IP address.

  Workgroups, conceptually, represent a group of servers rather

than an individual server and the browse list may be retrieved

from one or more browse servers. (A workgroup name is a NetBIOS group name.)

In these cases, the server name is interpreted as a reference to an SMB server only. Thus, workgroups may only be accessed via their NetBIOS names.

When testing the name using the algorithm presented in section A.4, a NetBIOS name suffix value of 0x20 is used to find an SMB server, and a suffix value of 0x1D or 0x1B is used to find a workgroup browse server.

A system operating in B mode will use the 0x1D suffix to search for a Local Master Browser operating on the same subnet. A system operating in P mode must use the 0x1B suffix to query the NBNS for the Domain Master Browser. An M mode system will first send a broadcast query for the 0x1D name and, if that fails, query the NBNS for the 0x1B name. H mode behavior is the opposite of M mode.