Introduction: CIFS from Eight Miles High

> Fillet of a fenny snake,
> In the cauldron boil and bake;
> Eye of newt, and toe of frog,
> Wool of bat, and tongue of dog.
>
> — *Macbeth*, Act IV, Scene i,
> William Shakespeare

0.1 First Impressions

First impressions are important. The handshake, the smile, here's our brochure, would you like a cup of tea?

Microsoft's Windows family of operating systems makes good first impressions. There's a pleasant sound at start-up, all of the basics are represented by simple icons, and everything else is available through a neatly categorized menu.

As the relationship progresses, however, it becomes clear that there is a lot going on beneath the candy-coated surface. This is particularly true of the CIFS protocol suite. The Network Neighborhood icon that appears on the Windows desktop hides a great deal of gear-churning and behind-the-scenes fussing.

The large installed base of Microsoft's Windows products has granted *de facto* standard status to CIFS. Unfortunately, implementation documentation and detailed protocol specifications are scarce, incomplete, and inconsistent. This is a problem for network administrators, third-party CIFS implementors, and anyone else who wants to know more about the ingredients than you can read on the bottom of the box.

Despite the dearth of good under-the-hood documentation, there are several non-Windows CIFS products. Some of these are based on older versions of Microsoft's own software, but the majority were created by studying the few available references and reverse-engineering to fill in the gaps.

0.2 What is CIFS?

CIFS is a network filesystem plus a set of auxiliary services supported by a bunch of underlying protocols. Any and all of these various bits have been called CIFS, which leaves us with a somewhat muddy definition. To make things easier, we'll start by saying that CIFS is "Microsoft's way of doing network file sharing", and work out the details as we go on.

The name "CIFS," of course, is an acronym. It stands for **C**ommon **I**nternet **F**ile **S**ystem, a title which deserves a bit of dissection.

Common

The term has a variety of connotations, but we will assume that Microsoft was thinking of *common* in the sense of *commonly available* or *commonly used*. All Microsoft operating systems have had some form of CIFS networking available or built in, and there are implementations of CIFS for most major non-MS operating systems as well.

Unfortunately, there is not yet a specification for CIFS that is complete, correct, authoritative, and freely available. Microsoft defines CIFS by their implementations and, as we shall see, their attempts at documenting the complete suite have been somewhat random. This has an adverse impact on the *commonality* of the system.

Internet

At the time that the "CIFS" name was coined many people felt that Microsoft was late to the table regarding the exploitation of the Internet. As will be described further on, the naming scheme they used back then (based on a piece of older LAN technology known as NetBIOS) doesn't scale to large networks — certainly not the Internet. The idea that CIFS would become an Internet standard probably came out of the work that was being done to redesign Microsoft's networking products for Windows NT5 (now known as Windows 2000 or W2K). Under W2K, CIFS can use the Domain Name System (DNS) for name resolution.

File System

CIFS allows you to share directories, files, printers, and other cool computer stuff across a network. That's the filesystem part. To make use of these shared resources you need to be able to find and identify them, and you also need to control access so that unauthorized folk won't fiddle where they shouldn't. This means that there is a hefty amount of administrivia to be managed, so CIFS file sharing comes surrounded by an entourage. There are separate, but intertwined protocols for service announcement, naming, authentication, and authorization. Some are based on published standards, others are not, and most have changed over the years.

0.2.1 *A Recipe for Protocol Soup*

The filesharing protocol at the heart of CIFS is an updated version of the venerable **S**erver **M**essage **B**lock (SMB) protocol, which dates back to the mid-1980s. The new name first appeared around 1996/97 when Microsoft submitted draft CIFS specifications to the **I**nternet **E**ngineering **T**ask **F**orce (IETF). Those drafts have since expired, and more recent documentation made available by Microsoft comes encumbered with confusing (and pointless) licensing restrictions.

The SMB protocol was originally developed to run over NetBIOS (**Net**work **B**asic **I**nput **O**utput **S**ystem) LANs. This is a nasty little skeleton in the CIFS closet. Until W2K, NetBIOS support was required for SMB transport. The machine and service names visible in the Windows "Network Neighborhood" are, basically, NetBIOS addresses.

With Windows 3.11 (Windows for Workgroups), Microsoft introduced a service announcement and location system called the Browse Service. This service maintains the list of available file and print services that is presented via the Network Neighborhood (named "My Network Places" in newer Windows products). Also with Windows 3.11 Microsoft introduced the "workgroup" concept. Workgroups simplified network management by organizing servers and services into administrative groups. Microsoft expanded upon the workgroup concept under Windows NT to create NT Domains.[1]

[1] The terms "NT Domain" and "W2K Domain" will be used to distinguish Microsoft's authentication/authorization domains from Domain Name System (DNS) domains.

As if that were not enough, there are also several SMB "dialects." These roughly correspond to major OS product releases or updates from Microsoft, and each adds extensions to the core SMB protocol. In their IETF CIFS draft, Microsoft presented an SMB dialect that was independent of NetBIOS, and W2K does include such a beast. As part of the split with NetBIOS, W2K also offers new name resolution, service announcement, authentication, and authorization mechanisms — all based, more or less, upon Internet standards.

Don't worry. Like most complex problems, this can all be understood by breaking it down into little pieces and studying each one in turn. The whole is not so terrible once you understand the parts.

0.3 The CIFS Community

Microsoft's implementations are the *de facto* CIFS standards. This is no surprise, as the SMB protocol was originally developed by IBM, Microsoft, Intel, and 3Com specifically for MS-DOS and PC-DOS. It is Microsoft's current massive dominance in the desktop world, however, that makes the CIFS marketplace worthwhile. Several companies earn their money by selling CIFS client and server software, or fileserver hardware with CIFS support. Without complete documentation, these third-party vendors might be forced to rely only on their own reverse-engineering or on licensed derivations of Microsoft's own implementations. This would reduce the "commonality" of CIFS and, given Microsoft's dominant market share, could have a negative impact on competitors' ability to compete.

Fortunately, there is a lot of communication within the CIFS community. There is also a renegade band of coders known as the Samba Team. Since 1991, they have been gathering information and implementing their own CIFS server, called Samba. (Note how the letters "s", "m", and "b" appear in sequence in the Samba name. Cool, eh?) Samba is published as Open Source under the terms of the GNU General Public License. Samba Team members typically share what they learn, and have even been known to write a little documentation now and again. Samba is included with most distributions of Linux and several commercial Unix flavors as well.

Samba has generated a few related projects, including SMB client filesystems for Linux, AmigaOS, and other platforms. There is also Richard Sharpe's `libsmbclient`, the Samba-TNG project, the jCIFS project, and this book.

0.3.1 *Visiting the Network Neighborhood*

On most days, members of the CIFS community can be found hanging out on Microsoft's CIFS mailing list, the Samba-Technical mailing list, or the jCIFS mailing list. In addition to these virtual geek cafés there is the mostly-annual CIFS conference. In the past it has been sponsored by such luminary organizations as EMC, Microsoft, Network Appliance, SCO, and the Storage Networking Industry Association. The conference provides an opportunity for CIFS developers to meet each other face-to-face, swap stories, whine, and (best of all) test their products with & against everyone else's. If you are serious about implementing CIFS, we'll see you there.

Service Network GmbH is the primary sponsor of yet another conference of interest. The first Samba eXPerience (aka sambaXP) was held in Göttingen, Germany, in April of 2002. It was very successful, and has become an annual event. While it is specific to Samba and related Open Source implementations (Samba-TNG, jCIFS, etc.), the information exchanged is valuable to anyone interested in CIFS networking.

0.3.2 *Community Collaborations*

It should also be noted that an effort, organized at one of the CIFS conferences and lead by the **S**torage **N**etwork **I**ndustry **A**ssociation (SNIA), has been underway within the CIFS community to draft an "open" CIFS reference with input from many interested parties. Version 1.0 of the SNIA CIFS Technical Reference has been released and is available on the SNIA web site. For more information, poke around the SNIA CIFS Working Group web pages.

0.4 Audience

This book is aimed at developers who want to add CIFS compatibility to their products. It will also be very helpful to network and system administrators who need to understand the curious things that CIFS does on the wire, in the server, and at the desktop. In addition, there is empirical evidence which suggests that the Internet security community (both the light and the dark sides) is keenly interested in the (mis)behavior of the CIFS suite. This is a technical book, and knowledge of programming and TCP/IP networking is assumed.

The protocol descriptions, however, start with the basics and build up, so very little previous knowledge of CIFS is expected.

For the programmer, there are several code examples. They have all been tested under Debian GNU/Linux, but you may need to do a little work to get them to run elsewhere. The code is intended to be illustrative rather than functional. It works, but it is not production-quality. That's okay, since part of the purpose of this book is to help you write your own code — if that's where your interests lie. If you don't care about source code you can safely skip much of it. Those who do like source can find additional examples at `http://ubiqx.org/libcifs/`.

A certain amount of SMB/CIFS protocol information has been available since the early days, but finding the important bits typically involves digging through detailed technical references, protocol specifications, packet dumps, web pages, whitepapers, source code, and mailing list archives. That's a lot of work, and a nuisance, and annoying. As a result, CIFS development has become an arcane art practiced by an elite few... and that's a darned shame.

This book attempts to solve this problem by selectively digging through the muck and presenting the uncovered gems in a coherent form, thus making the CIFS suite more accessible to more people.

0.5 Scope

Our focus is on the inner workings of CIFS filesharing, particularly the client side. Through necessity (and a macabre sense of fascination) we will also cover NetBIOS LAN emulation over TCP/IP, basic SMB authentication, and browsing. We will delicately dance around the NT Domain system and CIFS for W2K. These are much bigger and hairier, and deserve their own books.[2]

The book is separated into three main parts:

I. NBT: NetBIOS over TCP/IP

This part covers the NBT protocol, which is an implementation of the NetBIOS API on top of TCP/IP. NBT is necessary for communicating with older CIFS servers and clients.

[2] ...and if we find any such books, we will list them in the References section.

II. SMB: The Server Message Block Protocol

Part II covers SMB, the filesharing protocol at the core of CIFS. This part also covers authentication.

III. Browsing: Advertising Services

The Browser Service is built on top of NBT and SMB and is used to distribute information about the SMB fileservers available on the network.

Following these three parts are appendices, a glossary, bibliography for further reading, and an index — all the good stuff you would expect in such a book.

0.6 Acknowledgements and Thanks

The investigation of CIFS is a forensic art. This book is an attempt to coalesce the knowledge gathered by the CIFS community and present it in a useful form. My thanks go to the Samba Team, particularly Andrew Tridgell who started the Samba project and suggested that I start the jCIFS project. Thanks are also due to the jCIFS Team for raising — and often answering — so many good questions, and particularly to Michael B. Allen for churning out so much working code.

Acknowledgements also go to the folks on the Samba-Technical mailing list, the Samba-TNG mailing list, Microsoft's CIFS mailing list, and the folks at Microsoft who were able to provide insights into the workings of CIFS.

Writing documentation of this sort is a lengthy and annoying process. Special thanks go to the believers: Rachel, Aled, and Amalia; and also to the four-legs: Neko, Marika, Bran, and Maddie.

Additional notes of praise and recognition (in no particular order) to David Hirsch, Jeanne Dzurenko, Judy Diebel, Paul Nelis, Virginia Norton, Dave Farmer, John Ladwig, Susan Levy Haskell, Tim Howling, Olaf Barthel, Amy Gavel, Stephanie Cohen, Andrew Bartlett, Prairie Barnes, Chris Yerkes, James Carey, and Tom Barron.

The majority of the diagrams in this book were produced using the Dia diagram editor. The document was originally created as 100% hand-crafted and W3C-validated HTML using a simple text editor. CVS was used for document source management.

0.6.1 *The Book*

Thanks to Mark Taub for believing that I could turn my online ramblings into an honest-to-goodness book, and to Jill Harry for being “the boss” and gently but firmly guiding me through the process. Thanks also to Bruce Perens for including my book as part of his series, and to all the folks at Prentice Hall who helped to make this dream a reality.

The book was raked over the coals for technical correctness by Andrew Bartlett and Jerry Carter, both of the Samba Team and both nearly as pedantic as I am. They deserve a lot of credit for the good stuff that is contained herein (the bugs are my fault).

The original HTML source was skillfully converted to publisher-ready form by Alina Kirsanova, and then carefully copy-edited by Dmitry Kirsanov. They did excellent work. Any errors in grammar or formatting which remain are probably the result of my being a prima donna and insisting on having my own way.

0.7 About the Author

Christopher R. Hertel is one of those guys in the bright orange vests who lean up against a shovel in the construction zones along the Information Superhighway. By day, he is a Network Design Engineer at the University of Minnesota. He is also a member of the Samba Team, a founding member of the jCIFS Team, and an inconsistently average foil fencer. Most important of all, he is a full-time dad and husband.

0.7.1 *Quick Story*

A few years back I was interviewing for a job that I really thought I wanted. During the technical interview, I was asked “Is NetBEUI routable?” My head was full of protocol specs and packet headers, and I got a little flustered. I confused NetBEUI with the general idea of encapsulated NetBIOS. Of course I gave the wrong answer, and I did not get the job.

They say success is the sweetest and most honest form of revenge. ☺

0.8 License

Code examples are licensed under the terms of the GNU Lesser General Public License. This allows you to build libraries from the licensed code and use those libraries with your own code, even if your code is proprietary. The library source code, however, must be made available if you distribute your product. See the LGPL for details.