

History of the Finger Protocol

by Rajiv Shah

Draft 0.1 - June 2, 2000

[I. Specific Issues of Interest](#)

[II. Introduction](#)

[III. History](#)

[IV. The Finger Protocol](#)

[A. Standards](#)

[1. Pre-RFC 742](#)

[2. RFC 742](#)

[3. RFC 1194, 1996, and 1288](#)

[B. Running Code – Actual Implementations of Finger](#)

[1. BSD Unix](#)

[2. Solaris Unix](#)

[3. GNU Finger – October 1992 replaces Berkeley 4.3 Finger code \(Brittenson and Fox 1992\)](#)

[4. PFinger](#)

[5. Configurable Finger Daemon \(CFingerd\)](#)

[6. FFingerD](#)

[7. Troll-Ftpd](#)

[8. PsFingerD](#)

[9. InfoD](#)

[V. Alternative uses for Finger](#)

[VI. CMU Finger Controversy](#)

[A. Privacy Bits](#)

[B. Renaming Finger](#)

[VII. Privacy issues](#)

[A. Information Revealed and its Consequences](#)

[B. What control do users have](#)

[C. Control by the System Administrators](#)

[VIII. Security Issues](#)

[IX. Miscellaneous](#)

I. Specific Issues of Interest

1. General History of Finger
2. Use of Finger – how essential was it, what were the useful fields, how could it have been modified
3. Different Implementations of Finger – who created them, why, how they different from other daemon's, privacy issues, security issues, how popular were they, what were the defaults, how difficult is to write a Finger daemon, why was the BSD version so prevalent, where there other

versions widely used/available

4. RFCs – why they were written, who participated in the writing, relationship between the RFCs and the various implementations of Finger,
5. Privacy Issues – interested in what information was available, what the defaults were, what users could control, and what system administrators could control
6. Security Issues
7. Renaming of Finger
8. Other interesting anecdotes

II. Introduction

Finger was one of the first computer network applications.^[1] It enabled people to see who else was using the computer system as well as find basic information on that user. (Zimmerman 1991) To find information about a specific user, it was necessary to know that person's email address.^[2] For example, in response to the command `Finger atstarr@unix.amherst.edu` a computer running the Finger program would respond with the following information:

```
Login name: atstarr           In real life: Andrew Starr
Office: Kansas City         Home phone: 555-5555
Last login Mon Nov 8 13:22 on ttyre from sdn-ar-001mokcit
Plan:
To come so far one must be brave.
```

```
ATStarr@Amherst.Edu
http://www.amherst.edu/~atstarr/menu.html
```

Typical information provided by Finger would be a person's real name, their office location and phone number, and they last time they logged in. Users also could modify the plan field to add whatever text they wished. In this example, Andrew added a quotation, his email address, and the URL for his web page.

III. History

The Finger command was created in the early 1970s by Les Earnest at the Stanford Artificial Intelligence Lab (SAIL).^[3] (Zimmerman 1991) Brian Harvey at SAIL implemented the original Finger protocol. (Zimmerman 1991)

The Finger program inspired a copycat program, Name, written by Earl Killian for the ITS system at MIT.^[4] (Zimmerman 1991) Greg Hinchliffe brought up the Finger server for SRI-KA and SRI-KL. (Harrenstien 1977)

Les Earnest on the origins of Finger:

Finger was named for the act of pointing. I recall that sometime after it became popular I received a message from a system administrator who thought that it should be renamed so that users would not have to use a "dirty" word. I gave his request all the consideration that it deserved.

I created Finger around 1971 to meet a local need at the Stanford Artificial [sic] Intelligence Lab. People generally worked long hours there, often with unpredictable schedules. When you wanted to meet with some group, it was important to know who was there and when the others would likely reappear. It also was important to be able to locate potential volleyball players when you wanted to play, Chinese food freaks when you wanted to eat, and antisocial computer users when it appeared that something strange was happening on the system.

The only tool then available for seeing who was running on our DEC-10 computer was a WHO program that showed IDs and terminal line numbers for people who were logged in. There was no information available on people who were not logged in. I frequently saw people running their Fingers down the WHO display saying things like "There's Don and that's Pattie but I don't know when Tom was last seen." or "Who in hell is VVK and where does line 63 go?"

I wrote Finger and developed the supporting database to provide this information in traditional human terms -- real names and places. Because I preferred to talk face to face rather than through the computer or telephone, I put in the feature that tells how long the terminal had been idle, so that I could assess the likelihood that I would find them there if I walked down the hall.

The program was an instant hit. Some people asked for the Plan file feature so that they could explain their absence or how they could be reached at odd times, so I added it. I found it interesting that this feature evolved into a forum for social commentary and amusing observations.

Finger was picked up by a number of other groups with DEC-10 computers that were connected to Arpanet -- software flowed in all directions around the net in those days. It later migrated to Un*x, probably via U.C. Berkeley. Somewhere along the line the idea arose to provide a network Finger service. I don't remember who suggested that but it seemed like a good idea at the time so I stuck it in. Some other anxious people wanted to be able to verify that their mail was delivered to specific addressees, so the Mail feature was also added.

While I was somewhat surprised by the popularity of Finger, it has not been as successful as an earlier program that I invented -- the spelling checker. It too was created to fill a personal need that many others apparently share. We didn't think about commercial development and software protection in those days, but if we had we probably could have made something out of it. On the other hand, I enjoyed the comradery of those gentler times and have no regrets. (Earnest 1990)[\[5\]](#)

In 1977 K. Harrenstien wrote the first RFC, RFC 742, on the Name/Finger protocol. (Harrenstien 1977) He noted that at the time only SAIL, SRI, and ITS supported this protocol.

However, by publishing this standard, it would be possible for other computers to implement Finger.

Jeff Allen discussing the original Finger protocol:

"This simple protocol was easy to design, easy to implement, and most importantly, solved the problem at hand nicely: it allowed the researchers on a handful of machines to find out who was logged into a handful of other machines on the net. From there, it evolved into a quick and easy way for people to distribute information about themselves to others. It remains one of the primary ways PGP keys are exchanged." (Allen 1995)

In 1979 there was the CMU Finger controversy. (Discussed below)

In 1988 the Morris Worm exploited the Finger command. (Discussed in Security section)

Between 1990 to 1991, D. Zimmerman developed an "IAB standards track protocol" for Finger. The document was RFC 1288 "The Finger User Information Protocol" and obsoleted RFCs 1196, 1194, and 742. This is still the "official" standard for the Finger protocol. (Zimmerman 1991)

IV. The Finger Protocol

A. Standards

1. Pre-RFC 742

The original 1971 SAIL Finger command allowed people to query a database that contained information about the computer systems users. For example, it provided information about how long the terminal had been idle. Earnest notes that he later added the plan feature because people wanted to be able to update and modify the information that was displayed.

Finger then spread to others with DEC-10 computers **Is this the same as PDP-10 Computer??**

Later migrated to Unix via U.C. Berkeley **Need more info**

Later added mail feature **Need more info**

2. RFC 742

Calls it the Name/Finger protocol **After this RFC it would only be called the Finger protocol, What happened to Name?**

The RFC noted that SAIL, SRI, and ITS support this protocol

In comparison to the later RFCs, RFC 742 does not explicitly mention examples of what information should be public. Instead it merely states that the returned information should include the full names, last known terminal, and idle time (time since they last used the terminal). And the RFC suggests if the user is not logged in, the Finger protocol should return the "plan feature".

There is no mention in the RFC that information such as office location and phone number should be returned in response to the Finger command. There is also no mention in the RFC that the

Finger command should tell others if a person has new email and when they last checked their email. Although, the some of the examples show this information being provided.

****Need to learn the history of this RFC****

3. RFC 1194, 1996, and 1288

Between 1990 to 1991, D. Zimmerman developed an "IAB standards track protocol" for the Finger command. This document, RFC 1288 "The Finger User Information Protocol", is still the "official" standard for the Finger protocol. (Zimmerman 1991). RFC 1288 was preceded by RFC 1194 of November 1990, RFC 1196 of December 1990, and RFC 742. RFC's 1194 and 1196 are basically identical with RFC 1288 with only some minor corrections and clarifications. However, RFC 1288 differs in some aspects significantly from the RFC 742. RFC 1288 tried not to invalidate any of the existing implementations or add any unnecessary restrictions. Thus it attempted to maintain backward compatibility ****Was it truly backward compatible, what were the differences****. (Zimmerman 1991)

The RFC notes that the most prevalent implementation of Finger was that of the BSD Unix version. "Thus, this memo is based around the BSD version's behavior." The RFC then notes the problems with the BSD implementation of Finger. It suffers from not offering enough options to tailor the Finger RUIP for a particular sites's privacy policy or to protect the user from dangerous data. The RFC also emphasizes that there are many potential security problems with the Finger protocol. The RFC notes that the Finger protocol returns information about a system's users, which may be a "sensitive issue at best." (Zimmerman 1991)

The two most common Finger queries are the {C} query and the {U}{C} query.

The {C} query is a request for a list of all online users. The RFC states any users of the protocol must return a list of all the users with their full names or "actively refuse" the query. Additionally the standard recommends that the system administrator also include other useful information such as the terminal location, office location, office phone number, job name, and idle time. (Zimmerman 1991)

The {U}{C} query is request for more specific information about just one user. The RFC notes that "if you really want to refuse this service, you probably don't want to be running Finger in the first place." The {U}{C} query must return the persons full name and any information that would be revealed by the {C} query. Additionally, the standard recommends that additional information such as office location, office phone number, home phone number, status of login, and a user information file. The user information file (or a plan file) is a file where a user may leave a short message to be included in response to a Finger request. (Zimmerman 1991)

RFC 1288 also differs from RFC 742 in extensively discussing security concerns with the Finger protocol. The RFC notes that Finger is one of ways an intruder may attempt to hack into a computer system. The emphasis here is not a surprise. In 1988, the Morris worm exploited security holes in the Finger protocol. (Zimmerman 1991)

The RFC explicitly discusses security concerns about the information disclosed by Finger:

Warning!! Finger discloses information about users; moreover, such information may be considered sensitive. Security administrators should make explicit decisions about whether to run Finger and what information should be provided in responses. One existing implementation provides the time the user last logged in, the time he last read mail, whether unread mail was waiting for him, and who the most recent unread mail was from! This makes it possible to track conversations in progress and see where

someone's attention was focused. Sites that are information-security conscious should not run Finger without an explicit understanding of how much information it is giving away. (Zimmerman 1991)

The RFC also recommended that implementations of Finger should allow administrators to tailor the information returned such as whether to return office location, office phone number, and logged out time. (Zimmerman 1991)

The examples at the end of the RFC show that much more information than in previous RFC. For example, it information such as office location, directory, shell, and home phone were provided in the examples. (Zimmerman 1991)

B. Running Code – Actual Implementations of Finger

To use Finger, it is necessary for the host computer to run the Finger daemon (a program running in the background) which will answer Finger requests.

1. BSD Unix

Finger appeared in version 3.0 of the Berkeley Software Distribution (BSD) Unix.^[6] According to the system documentation the Finger command usually "displays the user's login name, real name, terminal name and write status (as a ``*' before the terminal name if write permission is denied), idle time, login time, office location and office phone number." By using the `-l` option, the following information would be displayed: "the user's home directory, home phone number, login shell, and the contents of the files `".forward"`, `".plan"` and `".project"` from the user's home directory."

2. Solaris Unix

In Sun's Solaris the default for the Finger command is to display the following information: user name, user's full name, terminal name (prepended with a ``*' (asterisk)` if write-permission is denied), idle time, login time, and host name if logged in remotely.^[7] If queried for a specific user than the following is provided: the user name and the user's full name, the user's home directory and login shell, time the user logged in if currently logged in, or the time the user last logged in; and the terminal or host from which the user logged in, last time the user received mail, and the last time the user read mail, the first line of the `$HOME/.project` file, if it exists, the contents of the `$HOME/.plan` file, if it exists

3. GNU Finger

The Finger command was designed to provide information about users on a computer system. This worked well in the 1970s when there were many people connected to one computer. However, by the 1990s, the networking environment had changed. In the 1990s there were many computers mostly with a single user. To Finger someone in the 1990s, it was necessary to Finger each individual computer. The GNU Finger implementation solved this problem by creating a central database listing all of the users in the site. This database was derived by continuously querying all the different

computers at a "site".^[8] GNU Finger was developed in October 1992 and replaced Berkeley 4.3 Finger code. (Brittenson and Fox 1992)

The GNU Finger displayed the the full name, home directory, shell, mail forwarding, Whether the user has any unread mail, and if so, when it was last read, the last login time and remote host (if known)and ``.plan'` and/or ``.project'` file. It also allowed users to disable Finger individually by linking `~user/.Fingerrc'` to `/bin/true'`.

4. PFinger

Another implementation of Finger, it is more recent. It claims that it has several security advantages over conventional implementations of the Finger command. For example, it allows .pgpfiles, no printing of users shell, home directory, and last login time. It also allows users to turn off Finger information by creating a .noFinger file or a user can update and store their own information.^[9]

5. Configurable Finger Daemon (CFingerd)

CFingerd is considered a hacked Finger daemon which provides extra security functions. It is considered an excellent replacement for standard Finger daemons. (1998) It was written by Ken Hollis with security issues in mind. According to its creators, cFingerd was created because many sites were turning off Finger for outside users. The system administrators did not want outsiders obtaining information about the users on the system. This program was created to provide these sites with a secure alternative. ^[10]

6. FFingerD

FFingerd was created as secure Finger service in response to system administrators disabling the Finger service because Finger advertised too much information about the system and the security flaws in standard Finger daemons. Some key features: "It has been verified to compile on a wide variety of Unix variants. It does not run with superuser privileges. It can display PGP public keys, too. It even has a fascist logging option for paranoid administrators. Users can exclude their account from Fingering. You can't see if an account has been excluded or is not there."^[11]

7. Troll-Ftpd

Alternative to wu-ftp and BSD Finger daemon which have security flaws. It is designed for Linux.^[12]

8. PsFingerD

In response to standard Finger daemon's that provide home directory, shell, and last login information which is valuably to hackers, PsFingerd was created. Some of the features of psFingerd are: Disallowing indirect Fingers and empty Fingers, Support for pgp public key, .noFinger option, and the ability for users to hide their real name.^[13]

9. InfoD

For Windows

<http://www.marketrends.net/infod/>

infod -- Windows Finger Daemon

V. Alternative uses for Finger

Information Retrieval

“Using this dandy little tool, among other things, I've found earthquake updates; a directory that lists which sodas are available in certain soda machines at Columbia University and Carnegie Mellon University, and National Football League standings.”^[14]

An interesting use of the Finger command was done at Carnegie-Mellon University in the 1970s. It involved wiring up a Coke machine to sense how many bottles were present of various flavors inside the machine. Next a program was written that allowed the status of the Coke machine to be determined by the use of the Finger command, Finger coke@cmua.^[15]

Plan files were the old home of ASCII art. Plans contain varied individualized information. For example, favorite quotes, ASCII art, and marketing information could be in a plan file. It was a precursor to the world wide web home page.^[16]

VI. CMU Finger Controversy

A. *Privacy Bits*

A short time ago, the CMU Finger program was endowed with the ability to reveal when a user last logged in and when that user last read his/her mail with our RDMail program. To respect the privacy of the individual I arranged for two user profile bits to be added to our existing profile facility (which determines whether a user automatically sees a bulletin board, or gets a message when mail arrives etc.) The two new bits determine whether Finger may reveal the date/time a user last logged in and the date/time that the MAIL.MSG file was last changed. The default setting for the profile bits inhibits Finger from revealing this information. – Email from Ivor Durham

To recapitulate, Ivor Durham added some privacy bits to allow a user to turn off information about their behavior. According to a report by Mike Schwartz, this information was (1) whether the user is currently logged on, (2) when the user had logged off, (3) whether there was any mail in the mailbox, (4) when the user has last read mail, and (5) if there is mail, the most recent sender. The privacy bits were an option that allowed people to decide if they want this information revealed. Moreover, the privacy bits had a default setting to prevent this information from being released. Thus, to enable others to find out when you last logged on, a person had to proactively turn their

privacy bit "on" to reveal this information. At CMU the other information revealed in the Finger command such as your office location and office number were left to the discretion of the user. Thus with the addition of the privacy bits, a user could now ensure that no information about them was revealed if someone "Fingered" them.

Ivor Durham was a graduate student who helped maintain the computer system responsible for running the Finger command. It was Durham who ensured that the Finger program was modified with the privacy bits. The decision for the implementation of the privacy bits was not made by Durham, but by the operations manager who insisted that people should be able to decide on whether to keep their information private.

****I am still in the process of determining all of the reasoning and decisions that lead up to the implementation of the privacy bits.****

The privacy bits were likely stored in the mail preferences of the rmail program. (Lamb 1999)

At the time of this Finger controversy, the CS department's computers were connected to the ARPAnet. This allowed them to communicate with other computers on the ARPAnet. The existing campus computers were not connected to the computers in the CS department. In fact, there was no general campus network for the CMU campus.

Access to the computers and the Finger command was generally open only to the faculty, graduate students, and the computer science departmental staff. The departmental staff varied from the technical people who maintained the computer systems to secretaries who would use the network for email and word processing. A few graduate students would volunteer to help maintain the computer systems and would receive extra computing privileges in return for their help.

B. Renaming Finger

About the time the Carnegie-Mellon University computer-center staff was ordered by the CMU administration to change the name of the "Finger" command (despite it being an ARPAnet standard). They changed "Finger" to "where" and also took it upon themselves to change Paul's name to "Paul Hilwhere" (initially intending it to be temporary). Paul actually approved of the change (as a kind of gentle protest), and it remained that way for some time.[\[17\]](#)

At some point, someone in Warner Hall, Warner Hall was the administration building, the upper floors were people like the president, vice-president, provost, one of them people higher up thought Finger implied some sort of grotesque image and thought Finger should be renamed. I don't know how their minds worked, but they thought it had obscene connotations so they insisted that it should be renamed, and the computer services people did rename it, I am not sure whether they forced us to rename our version, because they never used our machines. (Lamb 1999)

"the Tops-20 systems deployed for undergraduate use at the university had their ``finger" commands renamed to ``where" because someone in the administration thought that the ``finger" command verb might be interpreted in a rude manner." (Everhart 1999)

****NEED TO FIND MORE INFO ON THIS****

According to Emailman, "In prior days, when more people used shell accounts, "Finger Me!" was heard at campuses around the world!"[\[18\]](#)

VII. Privacy issues

A. Information Revealed and its Consequences

The amount of information available depends on the implementation of Finger. The following are typical fields that are returned:

Real Name, Terminal Name, Write Status ****What is this****, Idle Time, Login Time, Office Location, Office Phone Number, Home Directory, Login Shell, Home Phone Number, .forward, .plan, .project, Last Received Mail, Last Read Mail.

The more controversial fields are when and where you last logged off, when you last received email, and when you last checked your email.

Different implementations of Finger may show different information. For example, Fingering a person at the Internet Chess Club will only provide a person's rating and win loss record. If the person is logged on, ICC will also provide how long they have been logged on, how long they have been idle, whether playing or not, whether observing or not, etc., and any "notes" a person has written.[\[19\]](#) In *Maximum Security*, the author notes that at universities you can typically get the name, telephone number, dorm room number, and major of students. (1998)

Because many people are unaware of the information provided by Finger, some institutions have turned Finger off or set the default for the Finger command to off. In December 03, 1998, the Academic Information Systems (AcIS) at Columbia University in New York decided to modify the Finger command so that people must "turn it on" if they want to be Fingered. According to Jeffrey Eldredge the manager for computer support services, "The reason AcIS decided to select a default of "Fingeroff" rather than "Fingeron" was that many people are not aware that this command exists and how it provides [the entire world] access to personal information." According to Eldredge "Today's world of computing and communications is not such a kind and gentle place; thus, users are demanding better mechanisms to protect their privacy." Eldredge noted that reasoning was based on a number of incidents. (Horan 1998)

When the University of Minnesota Duluth shut off the ability for remote users to "Finger" users they noted issues of performance and privacy. "We have had users harassed on-line by otherwise unknown folks on the Internet. The harasser's used Finger service to find out information about who was logged on."[\[20\]](#)

******(I remember hearing stories where students would get harassed because people could find their telephone number and address easily through the Finger command (and the phonebook) I am sure the university is constantly turning peoples information off)******

Finger triggers Privacy Alarms

Many college and university computer system administrators are responding to rising concerns over misuse of the Finger tool with modifications that restrict the information users can glean, and some have eliminated it altogether. Critics note the tool violates privacy -- it provides information about where people are logging on and when they're

doing it -- and security -- crackers can use it to obtain information that can help them break into computer accounts. "A telephone directory is a great thing, until you realize that people who don't have your best interests in mind can use the information in it to do terrible things to you," says one university computer system administrator. (Chronicle of Higher Education 7/13/94 A15)

B. What control do users have

"The best managed systems allow users to make their own decisions whether information about their email reading habits and last login time will be displayed." (Notess 1995) This was a central issue in the CMU controversy.

There is little documentation and information for how users can change or stop information the Finger command reveals. Sometimes users can change information such as the address or phone number. But other times, all of this information including the last login time and mail information is not modifiable by the user. According to MS, usually users are allowed to change their information, however most users don't know they can request or do this. (MS)

To change information, users would use the chfn (change Finger) UNIX command. However, sometimes systems have this command disabled for security reasons.[\[21\]](#)

C. Control by the System Administrators

System administrators have a great deal of control over the Finger daemon. First, they could decide whether to run Finger. Second, they could control what information would be available on the users on the system. For example, system administrators could decide whether to place fields for office location, home phone number, and a .plan file. They could also remove about certain users who did not wish to be Fingered. Moreover, depending upon a system administrator's acumen, they could modify the defaults of the Finger daemon, for example, the controversy at CMU or the actions at Columbia University.

VIII. Security Issues

On November 2, 1988, the Internet was infected by a worm program. The Morris worm infected thousands of machines and disrupted normal Internet operations for several days. The worm was only able to successfully attack Sun workstations and VAXes running Berkeley Unix code. The worm program relied on several known access loopholes in sendmail and Fingerd. The worm was able to create a memory overflow and then execute a small program. Only 4.3 BSD VAX machines suffered from this attack.[\[22\]](#) (Reynolds 1989; Spafford 1988)

The Finger program suffered from two major security flaws. The first was that the Finger program provided information for hackers. The second was that some implementations of the Finger daemon were not secure as the Morris worm highlighted. As a result of the security and privacy problems many sites began not allowing Finger requests from remote users. (1998) Or sites just eliminated Finger.

"This excess information could be used as clues for guessing user passwords or exploiting other system problems."^[23] "The Finger service is the most common method of acquiring the necessary hints for cracking user passwords and compromising a user's account."^[24] It was also used by spammers.^[25]

"Some Finger daemons release information about the user's shell, home directory and group membership. This information may be used by hackers to attack the system. Some of the information can also be used to compromise the user account. For example, information such as the last time a user logged into the system could be used to build a table of usage patterns. Another example is that by knowing a user's home directory and exploiting a vulnerability in the mail system, a hacker could create an entrance into the system."^[26]

By attacking the Finger service it is possible to disrupt an NIS based network.^[27]

IX. Miscellaneous

Scalability of Finger

"With regard to solving what we now understand as the very complicated problem of Directory Services, Finger is a complete failure. In its time, it was a nice little application of the evolving network.

Why doesn't Finger fit the bill for a network-wide Directory Service? The biggest problem is that there is no cross-indexing in the system of servers. There are literally millions of servers out there, each holding a little bit of useful information. The problem is getting the right server, and retrieving the information of interest. Because the results of a Finger query can't be reliably parsed by a computer program, the arduous task of searching the global Finger database can't even be automated. It has to be done by hand by an experienced network user, one who knows how to find the information they are after." (Allen 1995)

Finger is not designed to log requests. So finding out who is Fingering you is complicated. You can use MasterPlan to identify who is trying to Finger you. It will also see if someone is trying to "clock" you. Clocking is the use of network utilities to monitor another user. (MS)

Finger through a Web browser, See <http://www-bprc.mps.ohio-state.edu/cgi-bin/Finger.pl>

Video Finger, <http://www.media.mit.edu/people/wad/mas961/vidFinger.html>

Discussion group on the Finger User Information Protocol, <http://listserv.spc.edu/archives/info-Finger.html>

Maximum Security : A Hacker's Guide to Protecting Your Internet Site and Network: Sams, 1998.

Allen, Jeff R. "Finding a Needle in a Virtual Haystack." A paper delivered at the Ninth System Administration Conference, Monterey, CA, Sep 18-22 1995.

Brittenson, Jan, and Brian Fox. *GNU Finger*. October 1992. Accessed May 16 2000. URL. Available from http://www.gnu.org/manual/finger-1.37/html_mono/finger.html.

Earnest, Les. *Origins of the finger command*. Feb 9 1990. Accessed August 8 1999. URL. Available from <http://www.web.us.uu.net/staff/djm/lore/finger-origin>.

Everhart, Craig. "Email Communication." , 1999.

Hafner, Katie, and Matthew Lyon. *Where Wizards Stay Up Late: The Origins of the Internet*. New York, NY: Touchstone, 1996.

Harrenstien, K. *RFC: 742 NAME/FINGER*. Dec 30 1977. Accessed May 20 1999. URL. Available from <ftp://ftp.isi.edu/in-notes/rfc742.txt>.

Horan, Brian. "Columbia U.'s AcIS modifies command for "fingering"." *Columbia University Daily Spectator*, December 3 1998.

Lamb, David Alex. "Personal Interview." , 1999.

Notess, Greg R. "On The Nets: Finding and Creating Finger Information." *Online*, May 1995.

Reynolds, J. *RFC 1135: The Helminthiasis of the Internet*. Dec 1989. Accessed May 15 2000. URL. Available from <http://www.cis.ohio-state.edu/htbin/rfc/rfc1135.html>.

Spafford, Eugene H. *The Internet Worm Program: An Analysis*. West Lafayette, IN: Purdue University, 1988, CSD-TR-823.

Zimmerman, D. *The Finger User Information Protocol*. Network Working Group, December 1991. Accessed June 3 1999. URL. Available from <http://info.internet.isi.edu:80/in-notes/rfc/files/rfc1288.txt>.

[1] For background on Finger see (1998; Notess 1995), http://www.faqs.org/faqs/signature_Finger_faq/, and <http://webopedia.internet.com/TERM/f/Finger.html>.

[2] Technically it is the user name and the host computer they are accessing. In fact, at some institutions such as the University of Illinois, the email address will not work because it represents several different hosts.

[3] See <http://www.loria.fr/services/tex/historique/SAIL-byebye.txt> for a history of SAIL

[4] Mark Crispin MSGGROUP#1726 Finger "was the inspiration for the NAME program at ITS."

[5] Part of this is excerpted in (Hafner and Lyon 1996, 216)

[6] <http://www.stopspam.org/usenet/mmf/man/Finger.html>

[7] <http://hoth.stsci.edu/man/man1/Finger.html>

[8] <http://www.media.mit.edu/people/wad/mas961/gnuFinger.html>

[9] <http://www.xelia.ch/unix/pFinger/about>

- [10] <http://www.infodrom.north.de/cFingerd/>
- [11] <http://www.math.fu-berlin.de/~leitner/fFingerd/>
- [12] <http://www.math.fu-berlin.de/~leitner/troll-ftp/>
- [13] <http://www.progsoc.uts.edu.au/local/Fingerd/> Modified fFingerd by Felix von Leitner –
- [14] 9/6/94 Newsday B29, COMPUTERS IN THE 90s LIFE IN CYBERSPACE Let Your 'Finger' Do the Cybering by Joshua Quittner. See <ftp://ftp.csd.uwm.edu/pub/Fingerinfo> or http://ils.unc.edu/emailpro/public_html/More_Fing.html for a list of sites that include topics such as weather reports, sport scores, and news.
- [15] See <http://www.abc.se/~jp/articles/computer/goodies/coke.txt>
- [16] <http://www.onlineinc.com/onlinemag/OL1995/Notess94/notess8.html>
- [17] Posted to Risk by Jim Horning, <http://catless.ncl.ac.uk/Risks/18.08.html>
- [18] <http://www.emailman.com/Finger/>
- [19] <http://www.chessclub.com/help/Finger>
- [20] <http://www.d.umn.edu/itss/support/HD/Finger.html>, see also http://www-cse.ucsd.edu/Computing_Facilities/Announcements/Finger_1997sep03.html
- [21] <http://kb.indiana.edu/data/adzw.html?cust=11652>
- [22] <http://www.netice.com/Advice/Phauna/Worm/Morris/>
- [23] http://www.wwdsi.com/demo/saint_tutorials/excessive_Finger_info.html
- [24] http://www.wwdsi.com/demo/saint_tutorials/excessive_Finger_info.html
- [25] <http://boardwatch.internet.com/mag/98/jan/bwm39.html>
- [26] http://www.wwdsi.com/demo/saint_tutorials/excessive_Finger_info.html
- [27] http://packetstorm.securify.com/advisories/iss/iss.98-06-29.nis_dos