

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
--

UNITED STATES DISTRICT COURT
NORTHERN DISTRICT OF CALIFORNIA

CLOSED CORPORATION, a California
Corporation

Plaintiff,

v.

OPEN SESAME USERS' GROUP, DOES
1-1000, and SCAPE GOAT,

Defendants.

CASE NO. CT-0001-DFO

**EXPERT REPORT OF
EDWARD W. FELTEN**

QUALIFICATIONS AND MATERIALS USED

1
2 1. My name is Edward W. Felten. I am Associate Professor of Computer Science at
3 Princeton University, where I have taught for six years. I received my B.S. from the California
4 Institute of Technology in 1985 and my Ph.D. in Computer Science and Engineering from the
5 University of Washington in 1993.

6 2. I have been involved in computing research since 1984. I have published more than fifty
7 articles in the computing research literature, and two books. I have won awards for my research
8 including a National Young Investigator award from the National Science Foundation, and an
9 Alfred P. Sloan Fellowship. I received an Outstanding Paper Award at the most recent
10 Symposium on Operating Systems Principles, the most prestigious operating systems research
11 conference. A copy of my *curriculum vitae* is attached as Appendix A.

12 3. My fields of specialization within computer science include operating systems, Internet
13 software, and Internet security.

14 4. I have been using the Internet personally since 1984. At Princeton I have taught several
15 courses related to the Internet, including (three times) a senior-level Distributed Computing and
16 Networking course that focuses on how the Internet works and how Internet-enabled operating
17 systems are constructed. I have led several research projects, supported by the National Science
18 Foundation, other government agencies, and companies such as Microsoft, Sun Microsystems,
19 and AT&T, related to Internet technology.

20 5. I have been asked to provide expert opinions on the nature and structure of the Internet,
21 the nature and structure of Usenet groups, open source software development methodology as
22 practiced by the Open Sesame group and other comparable groups, the extent and nature of the
23 participation of commercial enterprises in open source software projects, and related topics.

24 6. In addition to the documents and Web sites cited in this report, my analysis relied on the
25 Web sites of all companies mentioned in this report, the Internet standards documents relating to
26 Usenet, and the contents of the comp.os.linux.development.system newsgroup.

NATURE OF THE INTERNET

7. The Internet is a global network of computers constructed by patching together many local area networks that use widely varying communication media such as telephone lines, dedicated data cables, and wireless links. The Internet is characterized by its global scope and by the use of certain standard data formats and protocols such as the Transmission Control Protocol and the Internet Protocol (together, "TCP/IP") that ensure that any two computers on the Internet can exchange information with each other.

8. The Internet serves as the basic communication infrastructure for a wide range of electronic information services and activities, including electronic mail, electronic discussion groups, teleconferencing, and remote access by traveling workers to institutional databases.

9. FTP is an Internet protocol that allows one computer to make data files available so that any other computer on the Internet can get copies of those files. A computer that makes files available in this way is called an FTP server.

10. The Internet began in California, and it continues to have a very strong presence in California. A series of Georgia Tech surveys of the demographics of Internet users showed that in April 1998, 13.6% of all Internet users were located in California; in October 1998 this percentage had increased to 14.4%. (See http://www.gvu.gatech.edu/gvu/user_surveys/.)

NATURE OF USENET

11. "Usenet is a collection of online discussion groups [also called newsgroups] that are accessible from a large number (at least tens of thousands) of Internet sites." (<http://usenet-addresses.mit.edu/usenet.html>) An online discussion group is a repository of electronic messages collected under a designated name such as comp.os.linux.development.system.

12. Physically, a site is a computer connected to the Internet and running special message propagation software to make these messages available through the Internet to interested subscribers. Such computers are referred to as newsgroup servers.

13. Users create and read messages in a Usenet group by using a software application called a newsreader.

14. A user subscribes to a Usenet group by configuring his or her newsreader to retrieve from a newsgroup server all electronic messages for that discussion group.

1 15. Newsreaders typically present the messages in each discussion group in the same way as
2 standard e-mail mailboxes, so that the user can view, save, and forward the messages, and even
3 be notified when new messages for a particular discussion group arrive.

4 16. Besides viewing, a subscriber can use his newsreader to send or “post” a message to a
5 newsgroup by creating a new message and asking the newsreader to send it to that group.
6 Whether or not the message is added immediately to that discussion group’s repositories depends
7 on whether the newsgroup is moderated or unmoderated, as discussed below.

8 17. Subscribers may choose whether or not to reveal their real identities when posting to a
9 newsgroup.

10 18. Newsgroups are given names which usually indicate a topic of discussion. For instance,
11 the comp.os.linux.development.system discussion group contains messages related to the
12 development of the Linux operating system.

13 19. A newsgroup name is a series of text strings, separated by periods. The names form a
14 hierarchy. For example, newsgroups whose names start with “comp” have to do with computing,
15 newsgroups whose names start with “comp.os” have to do with computer operating systems, and
16 so on.

17 20. Newsgroups can be used as repositories of information (in the form of messages). This
18 information may contain Web addresses (URLs) or other references to material available online
19 by FTP or other means.

20 21. Newsgroups are made available to millions of individuals via the process of message
21 propagation. The process works as follows: each newsgroup server has a list of newsgroups it
22 “carries.” This list is simply a set of newsgroups that the administrators of that server choose to
23 keep track of. A newsgroup server may also know the Internet addresses of other newsgroup
24 servers. If two servers know each other’s Internet addresses and have at least one discussion
25 group they both carry, then they negotiate a message exchange link such that any message posted
26 to the common discussion groups on either one are also forwarded to the other. For example, if
27 two newsgroup servers A and B have a newsgroup message exchange link set up and both carry
28 comp.os.linux.development.system, then if a subscriber to A posts a message to
-- comp.os.linux.development.system this message will be added to the repository on A and

1 automatically forwarded to B to be added to its repository (This process works in a slightly
2 different way for moderated groups, but I omit the details since none of the groups discussed in
3 this report are moderated.)

4 22. A message posted to a newsgroup on one newsgroup server will spread from server to
5 server across thousands of individual message links, until it has reached every newsgroup server
6 in the world. Thus a message posted anywhere in the world is available worldwide.

7 23. There are thousands of newsgroups on the Internet. Anyone with a computer that is
8 connected to the Internet and has newsgroup server software installed on it can create new
9 newsgroups and can decide which other groups he or she chooses to carry.

10 24. The mere fact that someone has created a new newsgroup does not mean that the new
11 newsgroup will automatically be carried by all the other newsgroup servers in the world.

12 25. However, there is a common set of newsgroups that are carried by the majority of
13 newsgroup servers in the world. This set is sometimes referred to as the “standard” Usenet
14 newsgroup hierarchy. These are the newsgroups whose names begin with comp, humanities,
15 misc, news, rec, sci, soc and talk. (See the Guidelines for Usenet Group Creation.)

16 26. Because the newsgroups in the standard hierarchy have such a wide distribution, there is a
17 formalized process for adding a new newsgroup to that hierarchy. This is a multi-step process
18 that involves voting by members of the Usenet community, as discussed below.

19 27. Individuals who wish to create a newsgroup without going through this formalized
20 process may create an “alternative group,” whose name usually begins with “alt.” Any single
21 individual may create an alternative group without requiring approval or assistance from
22 anybody. Alternative newsgroups typically have a limited scope of distribution because of the
23 practical difficulty in convincing other newsgroup administrators to carry them.

24 28. If a small group of people wants to communicate via Usenet, the most convenient way to
25 do this is to set up an alternative newsgroup and then configure each of their newsgroup servers
26 to carry that newsgroup. The effort required to set up a newsgroup in the main hierarchy is only
27 worthwhile if the group is expected to be large and geographically dispersed.

28 29. A newsgroup can be moderated or unmoderated. For unmoderated newsgroups, any
-- message posted to the newsgroup is immediately propagated to all the servers that carry the

1 newsgroup. On the other hand, each moderated newsgroup has a member, designated as the
2 moderator, who must approve any message before it is propagated to the group.

3 30. The methodology for creating a newsgroup in the standard hierarchy is defined in a
4 document entitled "How to Format and Submit a New Group Proposal" document and is as
5 follows: a party interested in forming a new newsgroup in the standard hierarchy must:

- 6 1. Post a Request For Discussion (RFD) proposing the creation of the newsgroup to
7 news.announce.newgroups, news.groups, and any other newsgroups or mailing lists at
8 all related to the proposed topic. news.announce.newgroups is a moderated
9 newsgroup. The proposal must follow a designated format and must include at least
10 (a) a proposed newsgroup name, (b) a charter that describes what topics should be
11 discussed in the newsgroup (c) the name and email address of the proposed
12 moderator(s), or a statement that the newsgroup will be unmoderated, and (d)
13 proposal sponsor information that includes the names and e-mail addresses of all the
14 proponents of the proposal.
- 15 2. Allow 30 days for discussions of the particulars of the proposed newsgroup.
- 16 3. Post a Call For Votes on news.announce.newgroups and all the other newsgroups and
17 mailing lists on which the RFD was posted.
- 18 4. Arrange a neutral vote taker to collect and count the votes.

19 The voting period lasts at least 21 days and no more than 31 days. The vote taker posts
20 the results on news.announce.newgroups. If at least two-thirds of the votes are in favor of
21 creation and if the number of YES votes exceeds the number of NO votes by at least 100,
22 then the proposed newsgroup is added to the standard hierarchy. Proposals that fail
23 cannot be brought to a vote again until six months have elapsed since the previous vote.

24 OPEN SOURCE SOFTWARE DEVELOPMENT

25 1. An Open Source Software Product is a software product that is made available free of
26 charge, in source code form (and perhaps in other forms), to all parties who assent to a simple
27 license agreement. There are many Open Source Software Products.

- 28 2. Open Sesame is an Open Source Software Product.

1 3. Other examples of Open Source Software Products include the Linux and FreeBSD
2 operating systems, the KDE and GNOME graphical user interfaces, and the Apache web server.

3 4. Open Source Software Products are typically developed through collaboration among a
4 physically dispersed group of software developers.

5 5. The Internet plays an important role in the development of Open Source Software
6 Products by facilitating collaboration among large, worldwide groups of software developers.

7 6. Because every user of an Open Source Software Product has the source code for the
8 Product, users can modify the Product (or hire someone to modify it) in order to improve it or fix
9 flaws in it.

10 7. Users who have improved an Open Source Software Product are typically encouraged to
11 contribute their improvements to the Product so that others may benefit from them.

12 8. Typically, a small group of volunteers provides quality control for an Open Source
13 Software Product by examining the modifications contributed by members, and designating
14 which of these modifications are to be included in the next version of the Product.

15 SIMILARITIES BETWEEN OPEN SESAME AND LINUX

16 9. The organization of the Open Sesame group follows the typical pattern described above.
17 In the case of Open Sesame, modifications are contributed, and official versions are released, by
18 sending messages to the comp.os.open-sesame Usenet group.

19 10. Open Sesame is also made available on an FTP server that is accessible from anywhere in
20 the world. When a new version of Open Sesame is made available on the FTP server, this fact is
21 announced via a message to the comp.os.open-sesame newsgroup.

22 11. Like Open Sesame, the group developing the Linux operating system follows the typical
23 organization for Open Source Software Product developers, as described above.

24 12. The main difference in organization between the Open Sesame and Linux groups is that
25 the members of the Open Sesame group go to unusual lengths to maintain their anonymity.

26 13. Most Usenet messages, with the possible exception of messages on controversial or
27 fringe groups, reveal the identity (i.e. the real name) and the sponsoring organization of the
28 person who wrote them. For example, if we examine the 434 messages most recently sent (as of
--

1 October 14, 1999) to the comp.os.linux.development Usenet newsgroup (the closest analog to
2 comp.os.open-sesame), we find that 93.3% of the messages reveal the identity of their author.

3 14. By contrast, members of the Open Sesame newsgroup always hide their real identity and
4 sponsoring organization.

5 MOTIVATION FOR OPEN SOURCE OPERATING SYSTEMS

6 15. Open Source Software Products are attractive to commercial customers for several
7 reasons. First, they can be freely modified by each customer to meet that customer's needs, with
8 no need to wait for the product's vendor to do anything. Second, customers can quickly benefit
9 from improvements made by other customers.

10 16. Open Source Software Products provide many profit-making opportunities to software
11 developers and others in the computer hardware and software businesses.

12 17. For example, Linux has a very significant commercial impact. There is a large and
13 growing commercial sector devoted to Linux.

14 18. Computer manufacturers make profits from Linux by selling computers with Linux and
15 related software pre-installed. Examples include companies that specialize in Linux-based
16 computers, such as VA Linux Systems and Telenet Systems, and larger computer makers such as
17 Hewlett-Packard.

18 19. Several companies make profits from Linux by selling a bundle containing Linux, other
19 software, documentation, and a specialized installation facility, and by providing support to
20 customers in the process of installing Linux. Examples of such companies include Red Hat, Inc.
21 and Caldera Systems.

22 20. Many companies make profits from Linux by selling application programs designed to
23 run on Linux. For example, most of the major vendors of business database software, including
24 Oracle, Informix, and Sybase, offer versions of their products that run on Linux.

25 21. Companies like Cygnus Solutions make profits from Linux by selling technical support
26 services to users of Linux.

27 22. All of the companies listed above as making profits from Linux do business in the
28 Northern District of California. All but two (Red Hat and Caldera) are headquartered in the
-- Northern District of California.

1 23. In the near future companies will be making profits from Open Sesame in the same ways
2 the companies listed above are making profits from Linux.

3 24. Companies that make profits from an Open Source Software Product have an economic
4 incentive to contribute to that Product. It is common for commercial enterprises (acting through
5 their employees) to participate in the development of an Open Source Software Product.

6 25. Since the members of the Open Sesame group hide their identities, there is no basis for
7 any claim that the members are noncommercial entities or are not motivated by profit. Indeed,
8 given the size and apparent diversity of the group, the participation of commercial entities seems
9 likely.

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
--

Dated: October 14, 1999

Edward W. Felten

Appendix A: Curriculum Vitae

Edward W. Felten

Dept. of Computer Science

Princeton University

35 Olden Street

Princeton, NJ 08544

(609) 258-5906

fax: (609) 258-1771

felten@cs.princeton.edu

Education

- Ph.D. in Computer Science and Engineering, University of Washington, 1993.
- M.S. in Computer Science and Engineering, University of Washington, 1991.
- B.S. in Physics, with Honors, California Institute of Technology, 1985.

Employment

- Associate Professor of Computer Science, Princeton University, 1999-present.
- Assistant Professor of Computer Science, Princeton University, 1993-1999.
- Senior Computing Analyst, Caltech Concurrent Computing Project, California Institute of Technology, 1986-1989.

Honors and Awards

- Alfred P. Sloan Fellowship, 1997.

- 1 • Emerson Electric, E. Lawrence Keyes Faculty Advancement Award, Princeton University
2 School of Engineering, 1996.
- 3 • National Young Investigator award, 1994.

4 5 **Research Interests**

6
7 Operating Systems. Internet software. Computer security, especially relating to the World Wide
8 Web. Security of mobile code and mechanisms for distributing software over the Internet.
9 Interaction of security with programming languages and operating systems. Distributed
10 computing. Parallel computing architecture and software.

11 12 **Selected Publications**

13
14 Securing Java: Getting Down to Business with Mobile Code. Gary McGraw and Edward W.
15 Felten. John Wiley and Sons, New York, 1999.

16
17 An Empirical Study of the SHRIMP System. Matthias A. Blumrich, Richard D. Alpert, Yuqun
18 Chen, Douglas W. Clark, Stefanos N. Damianakis, Cezary Dubnicki, Edward W. Felten, Liviu
19 Iftode, Margaret Martonosi, Robert A. Shillner, and Kai Li. Proc. Of 25th International
20 Symposium on Computer Architecture, June 1998.

21
22 Performance Measurements for Multithreaded Programs. Minwen Ji, Edward W. Felten, and Kai
23 Li. Proc. of 1998 SIGMETRICS Conference, June 1998.

24
25 Understanding Java Stack Inspection. Dan S. Wallach and Edward W. Felten. Proc. of 1998
26 IEEE Symposium on Security and Privacy, May 1998.

1 Java Security: Web Browsers and Beyond. Drew Dean, Edward W. Felten, Dan S. Wallach, and
2 Dirk Balfanz. In "Internet Besieged: Countering Cyberspace Scofflaws," Dorothy E. Denning and
3 Peter J. Denning, eds. ACM Press, New York, 1997.

4
5 Extensible Security Architectures for Java. Dan S. Wallach, Dirk Balfanz, Drew Dean, and
6 Edward W. Felten. Proc. of 16th ACM Symposium on Operating Systems Principles, Oct. 1997.
7 Outstanding Paper Award.

8
9 Web Spoofing: An Internet Con Game. Edward W. Felten, Dirk Balfanz, Drew Dean, and Dan S.
10 Wallach. Proc. of 20th National Information Systems Security Conference, Oct. 1997.

11
12 A Java Filter. Dirk Balfanz and Edward W. Felten. Technical Report 567-97, Dept. of Computer
13 Science, Princeton University, October 1997.

14
15 Inside RISKS: Webware Security. Edward W. Felten. Communications of the ACM, 40(4):130,
16 1997.

17
18 Reducing Waiting Costs in User-Level Communication. Stefanos N. Damianakis, Yuqun Chen,
19 and Edward W. Felten. Proc. of 11th Intl. Parallel Processing Symposium, April 1997.

20
21 Stream Sockets on SHRIMP. Stefanos N. Damianakis, Cezary Dubnicki, and Edward W. Felten.
22 Proc. of 1st Intl. Workshop on Communication and Architectural Support for Network-Based
23 Parallel Computing, February 1997. (Proceedings available as Lecture Notes in Computer
24 Science #1199.)

25
26 Client-Server Computing on the SHRIMP Multicomputer. Stefanos N. Damianakis, Angelos
27 Bilas, Cezary Dubnicki, and Edward W. Felten. IEEE Micro 17(1):8-18, February 1997.

1 Fast RPC on the SHRIMP Virtual Memory Mapped Network Interface. Angelos Bilas and
2 Edward W. Felten. IEEE Transactions on Parallel and Distributed Computing, February 1997.

3
4 Java Security: Hostile Applets, Holes and Antidotes. Gary McGraw and Edward Felten. John
5 Wiley and Sons, New York, 1996.

6
7 Implementation and Performance of Integrated Application-Controlled File Caching, Prefetching
8 and Disk Scheduling. Pei Cao, Edward W. Felten, Anna R. Karlin, and Kai Li. ACM
9 Transactions on Computer Systems, Nov 1996.

10
11 Early Experience with Message-Passing on the SHRIMP Multicomputer. Richard D. Alpert,
12 Angelos Bilas, Matthias A. Blumrich, Douglas W. Clark, Stefanos Damianakis, Cezary
13 Dubnicki, Edward W. Felten, Liviu Iftode, and Kai Li. Proc. of 23rd Intl. Symposium on
14 Computer Architecture, 1996.

15
16 A Trace-Driven Comparison of Algorithms for Parallel Prefetching and Caching. Tracy Kimbrel,
17 Andrew Tomkins, R. Hugo Patterson, Brian N. Bershad, Pei Cao, Edward W. Felten, Garth A.
18 Gibson, Anna R. Karlin, and Kai Li. Proc. of 1996 Symposium on Operating Systems Design and
19 Implementation.

20
21 Simplifying Distributed File Systems Using a Shared Logical Disk. Robert A. Shillner and
22 Edward W. Felten. Princeton University technical report TR-524-96.

23
24 Java Security: From HotJava to Netscape and Beyond. Drew Dean, Edward W. Felten, and Dan
25 S. Wallach. Proc. of 1996 IEEE Symposium on Security and Privacy.

26
27 Integrated Parallel Prefetching and Caching. Tracy Kimbrel, Pei Cao, Edward W. Felten, Anna
28 R. Karlin, and Kai Li. Proc. of 1996 SIGMETRICS Conference.

1 Contention and Queueing in an Experimental Multicomputer: Analytical and Simulation-based
2 Results. Wenjia Fang, Edward W. Felten, and Margaret Martonosi. Princeton University
3 technical report TR-508-96.

4
5 Software Support for Virtual Memory-Mapped Communication. Cezary Dubnicki, Liviu Iftode,
6 Edward W. Felten, and Kai Li. Proc. of Intl. Parallel Processing Symposium, April 1996.

7
8 Protected, User-Level DMA for the SHRIMP Network Interface. Matthias A. Blumrich, Cezary
9 Dubnicki, Edward W. Felten, and Kai Li. Proc. of 2nd Intl. Symposium on High-Performance
10 Computer Architecture, Feb. 1996.

11
12 Improving Release-Consistent Shared Virtual Memory using Automatic Update . Liviu Iftode,
13 Cezary Dubnicki, Edward W. Felten, and Kai Li. Proc. of 2nd Intl. Symposium on High-
14 Performance Computer Architecture, Feb. 1996

15
16 Design and Implementation of NX Message Passing Using SHRIMP Virtual Memory Mapped
17 Communication. Richard D. Alpert, Cezary Dubnicki, Edward W. Felten, and Kai Li. Princeton
18 University technical report TR-507-96.

19
20 Synchronization for a Multi-Port Frame Buffer on a Mesh-Connected Multicomputer. Bin Wei,
21 Gordon Stoll, Douglas W. Clark, Edward W. Felten, and Kai Li. Parallel Rendering Symposium,
22 Oct. 1995.

23
24 A Study of Integrated Prefetching and Caching Strategies. Pei Cao, Edward W. Felten, Anna R.
25 Karlin, and Kai Li. Proc. of 1995 ACM SIGMETRICS Conference. Best Paper award.

26
27 Evaluating Multi-Port Frame Buffer Designs for a Mesh-Connected Multicomputer. Gordon
28 Stoll, Bin Wei, Douglas W. Clark, Edward W. Felten, Kai Li, and Patrick Hanrahan. Proc. of
-- 22nd Intl. Symposium on Computer Architecture.

1
2 Virtual Memory Mapped Network Interface Designs. Matthias A. Blumrich, Cezary Dubnicki,
3 Edward W. Felten, Kai Li, and Malena Mesarina. IEEE Micro, 15(1):21-28, February 1995.

4
5 Dynamic Tree Searching. Steve W. Otto and Edward W. Felten. In "High Performance
6 Computing", Gary W. Sabot, ed., Addison Wesley, 1995.

7
8 Implementation and Performance of Application-Controlled File Caching Pei Cao, Edward W.
9 Felten, and Kai Li. Proc. of 1st Symposium on Operating Systems Design and Implementation,
10 pages 165-178, November y1994.

11
12 Application-Controlled File Caching Policies. Pei Cao, Edward W. Felten, and Kai Li. Proc. of
13 USENIX Summer 1994 Technical Conference, pages 171-182, 1994.

14
15 Virtual Memory Mapped Network Interface for the SHRIMP Multicomputer. Matthias A.
16 Blumrich, Kai Li, Richard D. Alpert, Cezary Dubnicki, Edward W. Felten, and Jonathan S.
17 Sandberg. ISCA '94.

18
19 Protocol Compilation: High-Performance Communication for Parallel Programs. Edward W.
20 Felten. Ph.D. dissertation, Dept. of Computer Science and Engineering, University of
21 Washington, August 1993.

22
23 Building Counting Networks from Larger Balancers. Edward W. Felten, Anthony LaMarca, and
24 Richard Ladner. Univ. of Washington technical report UW-CSE-93-04-09.

25
26 Performance Issues in Non-Blocking Synchronization on Shared-Memory Multiprocessors. Juan
27 Alemany and Edward W. Felten. Proceedings of Symposium on Principles of Distributed
28 Computing, 1992.

1 Improving the Performance of Message-Passing Applications by Multithreading. Edward W.
2 Felten and Dylan McNamee. Proceedings of Scalable High-Performance Computing Conference
3 (SHPCC), 1992.

4
5 The Case for Application-Specific Communication Protocols. Edward W. Felten. Univ. of
6 Washington technical report TR-92-03-11.

7
8 A Centralized Token-Based Algorithm for Distributed Mutual Exclusion. Edward W. Felten and
9 Michael Rabinovich. Univ. of Washington technical report TR-92-02-02.

10
11 Issues in the Implementation of a Remote Memory Paging System. Edward W. Felten and John
12 Zahorjan. Univ. of Washington technical report TR-91-03-09.

13
14 A Highly Parallel Chess Program. Edward W. Felten and Steve W. Otto. 1988 Conference on
15 Fifth Generation Computer Systems.