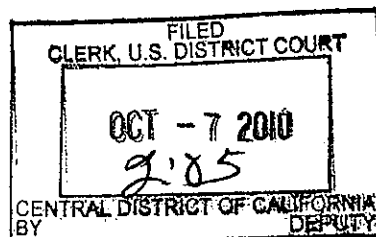


COPY

1 QUINN EMANUEL URQUHART  
& SULLIVAN, LLP  
2 Sean Pak (Bar No. 219032)  
seanpak@quinnemanuel.com  
3 Jennifer A. Kash (Bar No. 203679)  
jenniferkash@quinnemanuel.com  
4 50 California Street, 22<sup>nd</sup> Floor  
San Francisco, California 94111-4788  
5 Telephone: (415) 875-6600  
Facsimile: (415) 875-6700



6 QUINN EMANUEL URQUHART & SULLIVAN, LLP  
7 David M. Grable (Bar No. 237765)  
davegrable@quinnemanuel.com  
8 865 South Figueroa Street, 10<sup>th</sup> Floor  
Los Angeles, California 90017-2543  
9 Telephone: (213) 443-3000  
Facsimile: (213) 443-3100

10 QUINN EMANUEL URQUHART  
& SULLIVAN, LLP  
11 David A. Nelson\*  
12 davenelson@quinnemanuel.com  
500 West Madison St., Ste. 2450  
13 Chicago, Illinois 60661  
Telephone: (312) 705-7400  
14 Facsimile: (312) 705-7401

15 \* *pro hac vice* to be filed

16 Attorneys for Plaintiff Symantec  
Corporation

18 UNITED STATES DISTRICT COURT

19 CENTRAL DISTRICT OF CALIFORNIA, WESTERN DIVISION

20 SYMANTEC CORPORATION,

21 Plaintiff,

22 vs.

23 M86 SECURITY, INC.,

24 Defendant.

CASE NO. SACV10-1513-JST(RNB)

COMPLAINT FOR PATENT  
INFRINGEMENT

DEMAND FOR JURY TRIAL

26 This is a patent infringement action brought before this Court pursuant to  
27 28 U.S.C. §§ 1331 and 1338(a), in which Plaintiff, Symantec Corporation  
28

1 (“Symantec”), for its complaint against Defendant, M86 Security, Inc. (“M86”),  
2 alleges as follows:

### 3 INTRODUCTION

4 1. This is an action brought by Symantec against M86 for M86’s  
5 infringement of Symantec’s patents. In particular, Symantec seeks remedies for  
6 M86’s infringement of Symantec’s U.S. Patents Nos. 5,898,784 (“the ’784 patent”),  
7 5,996,011 (“the ’011 patent”) and 7,366,919 (“the ’919 patent”) (collectively, “the  
8 Asserted Patents”).

### 9 PARTIES

10 2. Symantec, Inc. is a corporation organized and existing under the laws  
11 of the State of Delaware, having a principal place of business at 350 Ellis Street,  
12 Mountain View, California 94043.

13 3. On information and belief, M86 Security, Inc. is a corporation  
14 organized and existing under the laws of the State of Delaware, having a principal  
15 place of business at 828 West Taft Avenue, Orange, California 92865.

### 16 JURISDICTION AND VENUE

17 4. This lawsuit is a civil action for patent infringement arising under the  
18 patent laws of the United States, 35 U.S.C. § 101, *et seq.* Accordingly, this Court  
19 has subject matter jurisdiction pursuant to 28 U.S.C. §§ 1331 and 1338(a).

20 5. This Court has personal jurisdiction over M86. On information and  
21 belief, M86 is a corporation having its principal place of business within this  
22 District. Furthermore, on information and belief, M86 has purposefully established  
23 substantial, systematic and continuous contacts with this District and expects or  
24 should reasonably expect to be haled into court here. Thus, this Court’s exercise of  
25 jurisdiction over M86 will not offend traditional notions of fair play and substantial  
26 justice.

6. Venue is proper in this District pursuant to 28 U.S.C. §§ 1391(b)–(c) and 1400(b) because M86 resides in this District and because a substantial part of the events or omissions giving rise to this claim occurred in this District.

**COUNT I: INFRINGEMENT OF U.S. PATENT NO. 5,898,784**

7. Symantec incorporates by reference the preceding averments set forth in paragraphs 1–6.

8. The '784 patent, entitled "Transferring Encrypted Packets over a Public Network," was duly and lawfully issued on April 27, 1999. A true and correct copy of the '784 patent is attached to this Complaint as Exhibit 1.

9. Symantec is the owner of all rights, title, and interest in the '784 patent, including the right to bring this suit for injunctive relief and damages.

10. On information and belief, M86 has infringed and continues to infringe, has contributed to and continues to contribute to acts of infringement, and/or has actively and knowingly induced and continues to actively and knowingly induce the infringement of the '784 patent by making, using, offering for sale and selling in the United States, and by importing into the United States without authority, and/or by causing others to make, use, offer for sale and sell in the United States, and import into the United States without authority, products and services, including but not limited to the M86 Secure Web Gateway and the M86 Secure Web Service Hybrid products and related services.

11. On information and belief, M86's infringement, contributory infringement and/or inducement of infringement is literal infringement or, in the alternative, infringement under the doctrine of equivalents.

12. M86's infringing activities have caused and will continue to cause Symantec irreparable harm, for which it has no adequate remedy at law, unless M86's infringing activities are enjoined by this Court in accordance with 35 U.S.C. § 283.

1        13. Symantec has been and continues to be damaged by M86's  
2 infringement of the '784 patent in an amount to be determined at trial.

3        14. On information and belief, M86's infringement of the '784 patent is  
4 willful and deliberate, and justifies an increase in damages of up to three times in  
5 accordance with 35 U.S.C. § 284.

6        15. On information and belief, M86's infringement of the '784 patent is  
7 exceptional and entitles Symantec to attorneys' fees and costs incurred in  
8 prosecuting this action under 35 U.S.C. § 285.

9                    **COUNT II: INFRINGEMENT OF U.S. PATENT NO. 5,996,011**

10        16. Symantec incorporates by reference the preceding averments set forth  
11 in paragraphs 1-6.

12        17. The '011 patent, entitled "System and Method for Filtering Data  
13 Received by a Computer System," was duly and lawfully issued on November 30,  
14 1999. A true and correct copy of the '011 patent is attached to this Complaint as  
15 Exhibit 2.

16        18. Symantec is the owner of all rights, title, and interest in the '011 patent,  
17 including the right to bring this suit for injunctive relief and damages.

18        19. On information and belief, M86 has infringed and continues to infringe,  
19 has contributed to and continues to contribute to acts of infringement, and/or has  
20 actively and knowingly induced and continues to actively and knowingly induce the  
21 infringement of the '011 patent by making, using, offering for sale and selling in the  
22 United States, and by importing into the United States without authority, and/or by  
23 causing others to make, use, offer for sale and sell in the United States, and import  
24 into the United States without authority, products and services, including but not  
25 limited to the M86 WebMarshal and the M86 MailMarshal products and related  
26 services.

20. On information and belief, M86's infringement, contributory infringement and/or inducement of infringement is literal infringement or, in the alternative, infringement under the doctrine of equivalents.

21. M86's infringing activities have caused and will continue to cause Symantec irreparable harm, for which it has no adequate remedy at law, unless M86's infringing activities are enjoined by this Court in accordance with 35 U.S.C. § 283.

22. Symantec has been and continues to be damaged by M86's infringement of the '011 patent in an amount to be determined at trial.

23. On information and belief, M86's infringement of the '011 patent is willful and deliberate, and justifies an increase in damages of up to three times in accordance with 35 U.S.C. § 284.

24. On information and belief, M86's infringement of the '011 patent is exceptional and entitles Symantec to attorneys' fees and costs incurred in prosecuting this action under 35 U.S.C. § 285.

**COUNT III: INFRINGEMENT OF U.S. PATENT NO. 7,366,919**

25. Symantec incorporates by reference the preceding averments set forth in paragraphs 1-6.

26. The '919 patent, entitled "Use of Geo-Location Data for Spam Detection," duly and lawfully issued on January 20, 1998. A true and correct copy of the '919 patent is attached to this Complaint as Exhibit 3.

27. Symantec is the owner of all rights, title, and interest in the '919 patent, including the right to bring this suit for injunctive relief and damages.

28. On information and belief, M86 has infringed and continues to infringe, has contributed to and continues to contribute to acts of infringement, and/or has actively and knowingly induced and continues to actively and knowingly induce the infringement of the '919 patent by making, using, offering for sale and selling in the United States, and by importing into the United States without authority, and/or by

1 causing others to make, use, offer for sale and sell in the United States, and import  
2 into the United States without authority, products and services, including but not  
3 limited to M86's MailMarshal products and related services.

4 29. On information and belief, M86's infringement, contributory  
5 infringement and/or inducement of infringement is literal infringement or, in the  
6 alternative, infringement under the doctrine of equivalents.

7 30. M86's infringing activities have caused and will continue to cause  
8 Symantec irreparable harm, for which it has no adequate remedy at law, unless  
9 M86's infringing activities are enjoined by this Court in accordance with 35 U.S.C.  
10 § 283.

11 31. Symantec has been and continues to be damaged by M86's  
12 infringement of the '919 patent in an amount to be determined at trial.

13 32. On information and belief, M86's infringement of the '919 patent is  
14 willful and deliberate, and justifies an increase in damages of up to three times in  
15 accordance with 35 U.S.C. § 284.

16 33. On information and belief, M86's infringement of the '919 patent is  
17 exceptional and entitles Symantec to attorneys' fees and costs incurred in  
18 prosecuting this action under 35 U.S.C. § 285.

**REQUEST FOR RELIEF**

WHEREFORE, Symantec respectfully requests that:

(a) Judgment be entered that M86 has infringed one or more claims of each of the Asserted Patents;

(b) Judgment be entered permanently enjoining M86, its directors, officers, agents, servants and employees, and those acting in privity or in concert with them, and their subsidiaries, divisions, successors and assigns, from further acts of infringement, contributory infringement, or inducement of infringement of the Asserted Patents;

(c) Judgment be entered awarding Symantec all damages adequate to compensate it for M86's infringement of the Asserted Patents including all pre-judgment and post-judgment interest at the maximum rate permitted by law;

(d) Judgment be entered that M86's infringement of each of the Asserted Patents is willful and deliberate, and therefore, that Symantec is entitled to treble damages as provided by 35 U.S.C. § 284;

(e) Judgment be entered that M86's infringement of the Asserted Patents is willful and deliberate, and, therefore, that this is an exceptional case entitling Symantec to an award of its attorneys' fees for bringing and prosecuting this action, together with interest, and costs of the action, pursuant to 35 U.S.C. § 285; and

(f) Judgment be entered awarding Symantec such other and further relief as this Court may deem just and proper.



1 DATED: October 7, 2010

Respectfully submitted,

QUINN EMANUEL URQUHART &  
SULLIVAN, LLP

By 

Jennifer A. Kash

Attorneys for Plaintiff Symantec  
Corporation



QUINN EMANUEL URQUHART  
& SULLIVAN, LLP  
Sean Pak (Bar No. 219032)  
seanpak@quinnemanuel.com  
Jennifer A. Kash (Bar No. 203679)  
jenniferkash@quinnemanuel.com  
50 California Street, 22<sup>nd</sup> Floor  
San Francisco, California 94111-4788  
Telephone: (415) 875-6600  
Facsimile: (415) 875-6700

QUINN EMANUEL URQUHART & SULLIVAN, LLP  
David M. Grable (Bar No. 237765)  
davegrable@quinnemanuel.com  
865 South Figueroa Street, 10<sup>th</sup> Floor  
Los Angeles, California 90017-2543  
Telephone: (213) 443-3000  
Facsimile: (213) 443-3100

QUINN EMANUEL URQUHART  
& SULLIVAN, LLP  
David A. Nelson\*  
davenelson@quinnemanuel.com  
500 West Madison St., Ste. 2450  
Chicago, Illinois 60661  
Telephone: (312) 705-7400  
Facsimile: (312) 705-7401  
\* *pro hac vice* to be filed

Attorneys for Plaintiff Symantec  
Corporation

UNITED STATES DISTRICT COURT

CENTRAL DISTRICT OF CALIFORNIA, WESTERN DIVISION

SYMANTEC CORPORATION,

Plaintiff,

vs.

M86 SECURITY, INC.,

Defendant.

CASE NO.

DEMAND FOR JURY TRIAL

**TO EACH PARTY AND TO THE COUNSEL OF RECORD FOR  
EACH PARTY:**

Plaintiff Symantec Corporation hereby demands a jury trial in the above-titled  
action pursuant to Rule 38(b) of the Federal Rules of Civil Procedure.

1 DATED: October 7, 2010

Respectfully submitted,

QUINN EMANUEL URQUHART &  
SULLIVAN, LLP

By 

Jennifer A. Kash

Attorneys for Plaintiff Symantec  
Corporation

# **Exhibit 1**



US005898784A

**United States Patent** [19]

Kirby et al.

[11] **Patent Number:** **5,898,784**[45] **Date of Patent:** **Apr. 27, 1999**[54] **TRANSFERRING ENCRYPTED PACKETS  
OVER A PUBLIC NETWORK**[75] Inventors: **Alan J. Kirby**, Hollis, N.H.; **Jeffrey A. Kraemer**, Northboro; **Ashok P. Nadkarni**, Shrewsbury, both of Mass.[73] Assignee: **Raptor Systems, Inc.**, Waltham, Mass.[21] Appl. No.: **08/963,512**[22] Filed: **Nov. 3, 1997**

swIpe IP Security Protocol, Columbia University and AT&amp;T, John Ioannidis and Matt Blaze, Dec. 1993.

Security Architecture for the Internet Protocol, R. Atkinson, Naval Research Laboratory, pp. 1-44, Aug. 1995.

IP Authentication Header, R. Atkinson, Naval Research Laboratory, pp. 1-26, Aug. 1995.

IP Encapsulating Security Payload (ESP), R. Atkinson, Naval Research Laboratory, pp. 1-24, Aug. 1995.

The Photuris Session Key Management Protocol, P. Karn, Qualcomm, W.A. Simpson, DayDreamer, pp. 1-106, Nov. 1995.

**Related U.S. Application Data**

[63] Continuation of application No. 08/586,230, Jan. 16, 1996, abandoned.

[51] Int. Cl.<sup>6</sup> ..... **H04L 9/00**[52] U.S. Cl. .... **380/49**[58] Field of Search ..... **380/23, 30, 49**

(List continued on next page.)

*Primary Examiner*—Salvatore Cangialosi*Attorney, Agent, or Firm*—Fish & Richardson P.C.

[57]

**ABSTRACT**

The invention features receiving encrypted network packets sent over a network at a network interface computer, and passing the encrypted network packets to a computer on an internal network.

[56] **References Cited****U.S. PATENT DOCUMENTS**

5,099,517	3/1992	Gupta et al.	380/49
5,161,193	11/1992	Lampson et al.	380/49
5,235,644	8/1993	Gupta et al.	380/49
5,325,362	6/1994	Aziz	370/94.3
5,416,842	5/1995	Aziz	380/49
5,442,708	8/1995	Adams, Jr. et al.	380/49
5,444,782	8/1995	Adams, Jr. et al.	380/49
5,548,646	8/1996	Aziz et al.	380/23
5,550,984	8/1996	Gelb	370/94.1

**OTHER PUBLICATIONS**

"Internet Portal, Version 1.1", pp. 1-16, Digital Equipment Corp.

"Digital Internet Tunnel V1.0", pp. 1-3, Digital Equipment Corp.

"Internet Security: Screening External Access Link (SEAL)", pp. 1-2, Digital Equipment Corp.

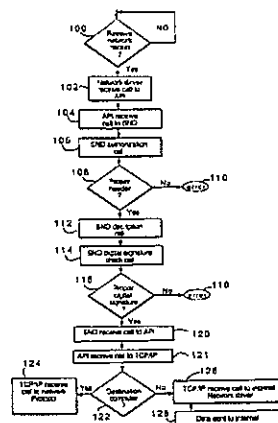
World Wide Web document of Digital Equipment Corporation's tunneling product, Nov. 15, 1995.

Kerberos Network Authentication Service (V5), Digital Equipment Corporation, pp. 1-106, Sep. 1993.

The invention also features receiving encrypted network packets at a first computer over a network from a second computer, examining a field in each network packet to determine which of a plurality of encryption algorithms was used to encrypt the network packet, and decrypting the network packet in accordance with the determined encryption algorithm.

The invention further features receiving network packets sent over a network, determining which virtual tunnel each network packet was sent over, and routing each network packet to a destination computer in accordance with the determined virtual tunnel.

The invention features encrypting network packets at a computer connected to an internal network, passing the encrypted network packet over the internal network to a public network interface computer, and passing the encrypted network packet over a public network connected to the network interface computer.

**26 Claims, 10 Drawing Sheets****Exhibit** 1**Page** //

5,898,784

Page 2

---

OTHER PUBLICATIONS

Simple Key-Management For Internet Protocols (SKIP), Ashar Aziz, et al. Sun Microsystems, Inc., pp. 1-72, Dec. 1995.

Internet Security Association and Key Management Protocol (ISAKMP), Douglas Maughan and Mark Schertler, National Security Agency, pp. 1-117, Nov. 1995.

U.S. application No. 08/561,790, filed Nov. 22, 1994, Kirby et al.—Controlling Passage Of Packets Or Messages.

U.S. application No. 08/585,765, filed Jan. 12, 1995, Kirby et al.—Data Encryption/Decryption For Network Communication.

U.S. application No. 08/586,231, filed Jan. 12, 1995, Levesque et al.—Key Management For Network Communication.

Exhibit 1

Page 12

U.S. Patent

Apr. 27, 1999

Sheet 1 of 10

5,898,784

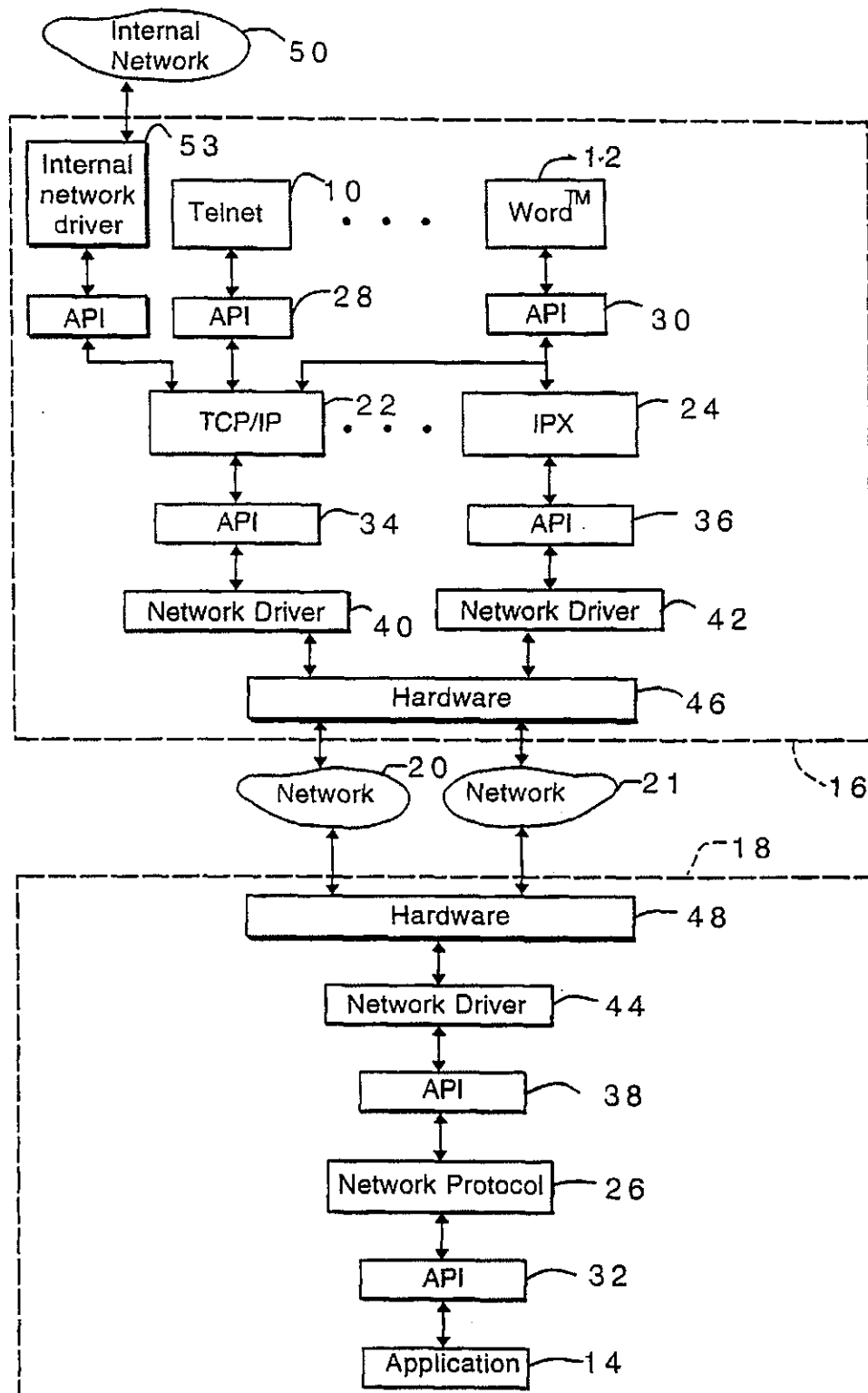


FIG. 1

U.S. Patent

Apr. 27, 1999

Sheet 2 of 10

5,898,784

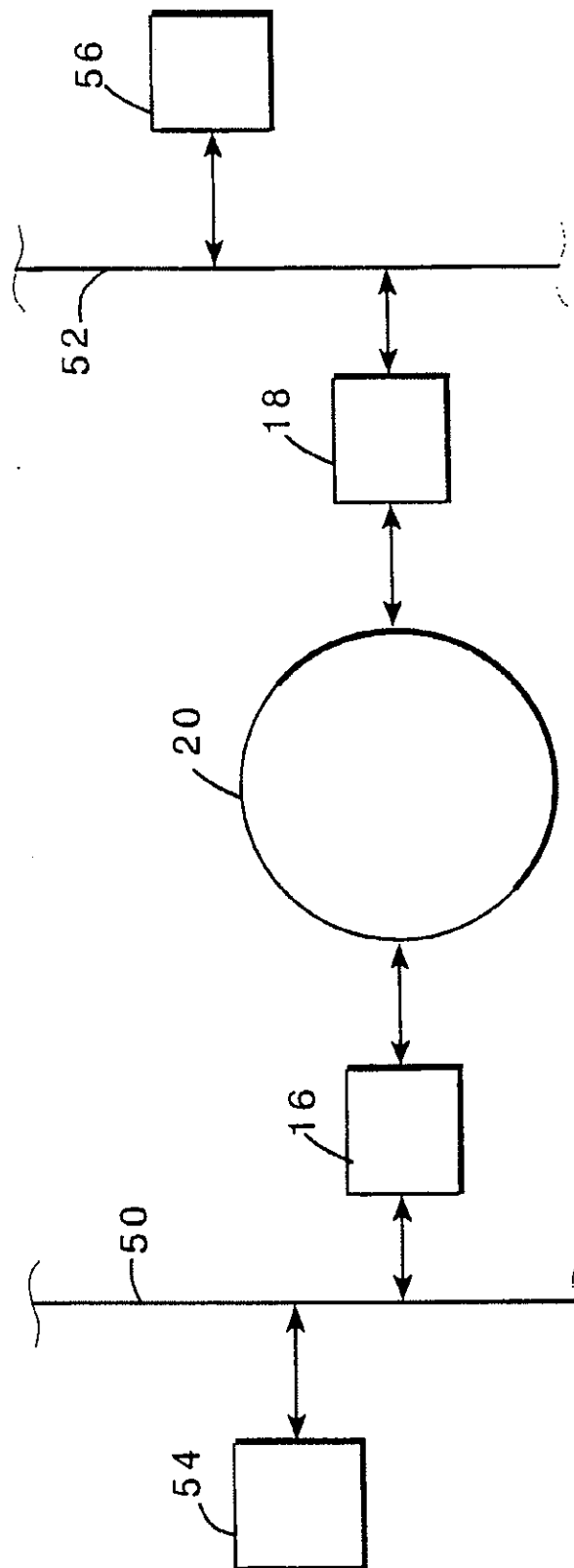


FIG. 2



U.S. Patent

Apr. 27, 1999

Sheet 3 of 10

5,898,784

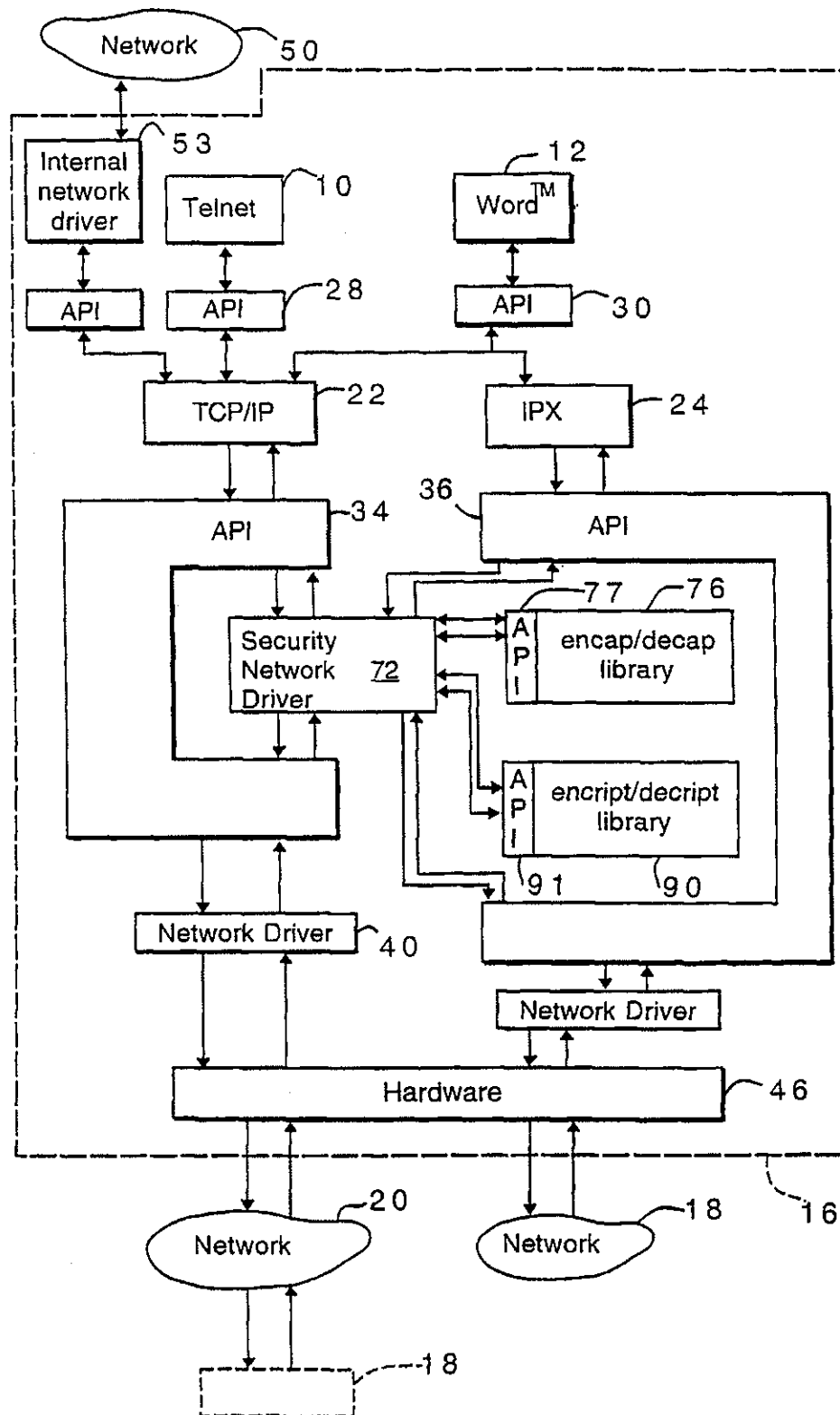


FIG. 3

U.S. Patent

Apr. 27, 1999

Sheet 4 of 10

5,898,784

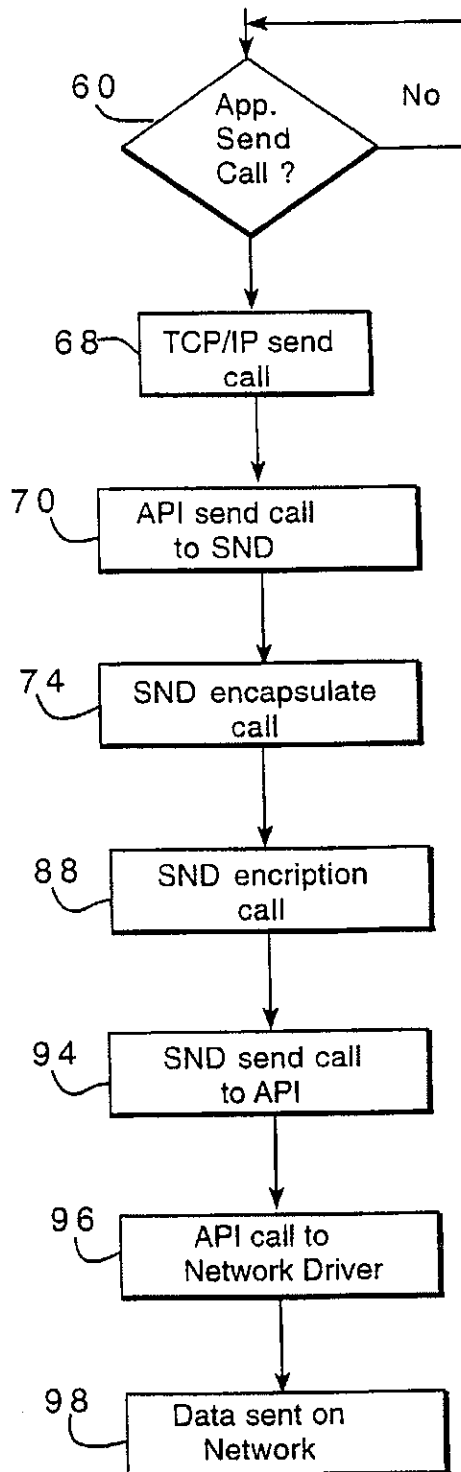


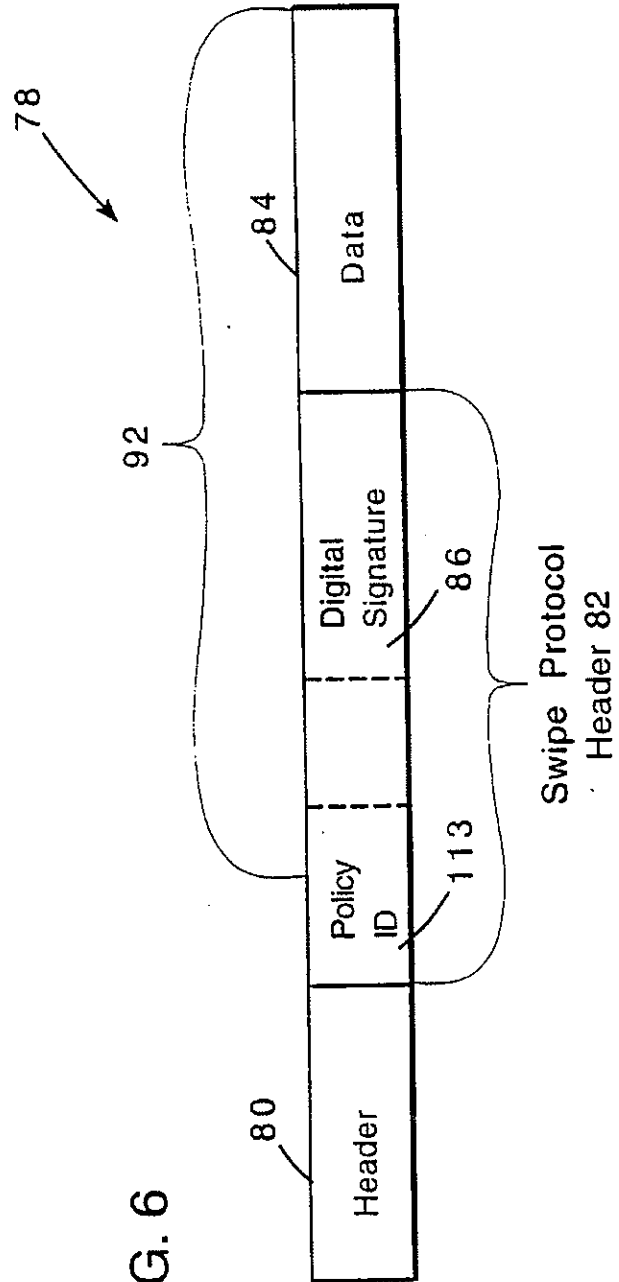
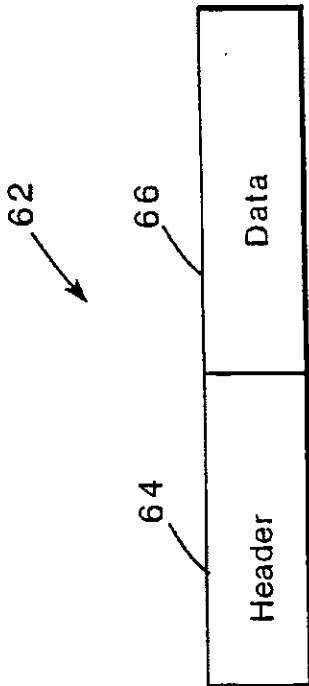
FIG. 4

U.S. Patent

Apr. 27, 1999

Sheet 5 of 10

5,898,784



U.S. Patent

Apr. 27, 1999

Sheet 6 of 10

5,898,784

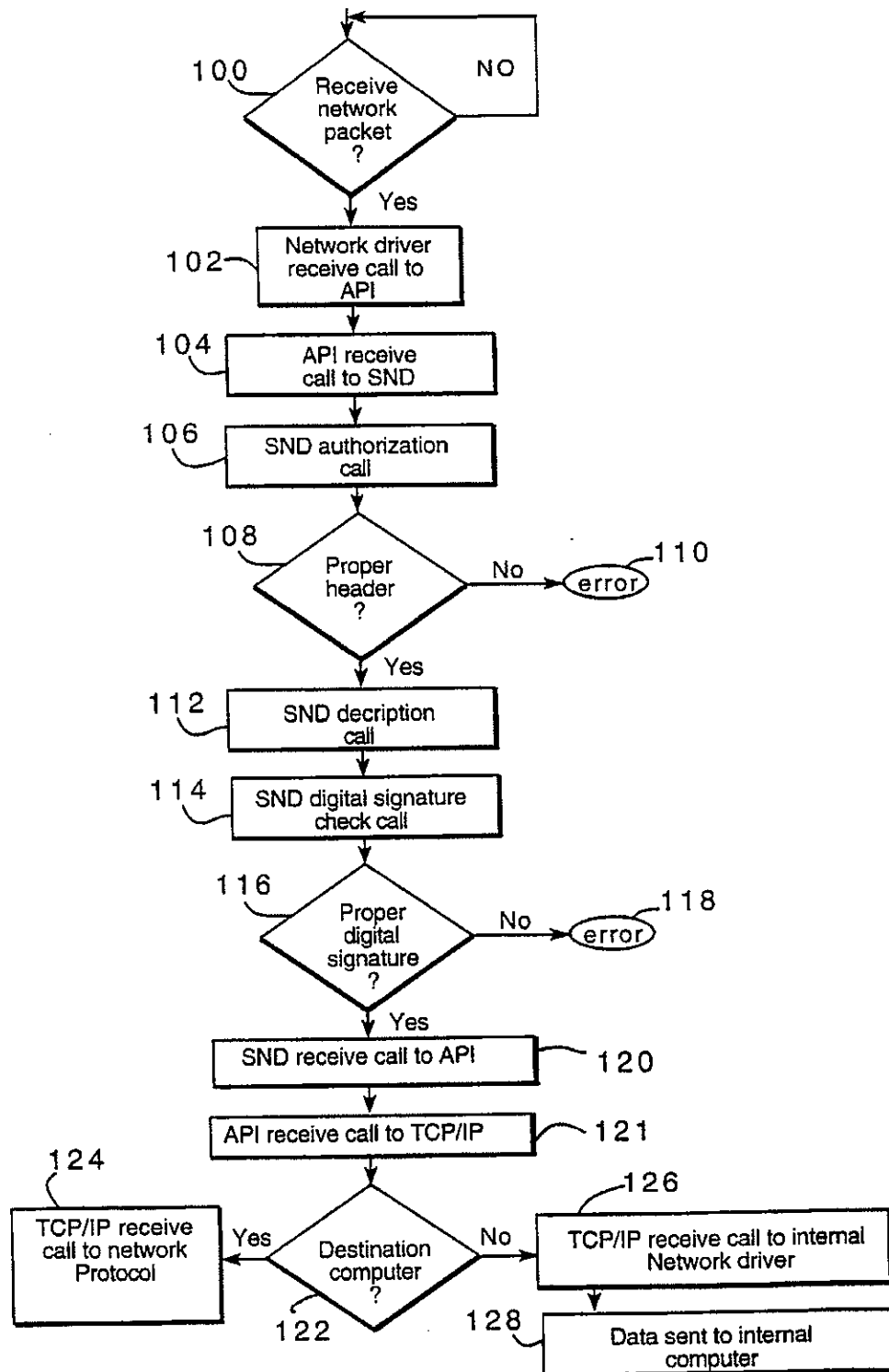


FIG. 7

**U.S. Patent**

**Apr. 27, 1999**

Sheet 7 of 10

**5,898,784**

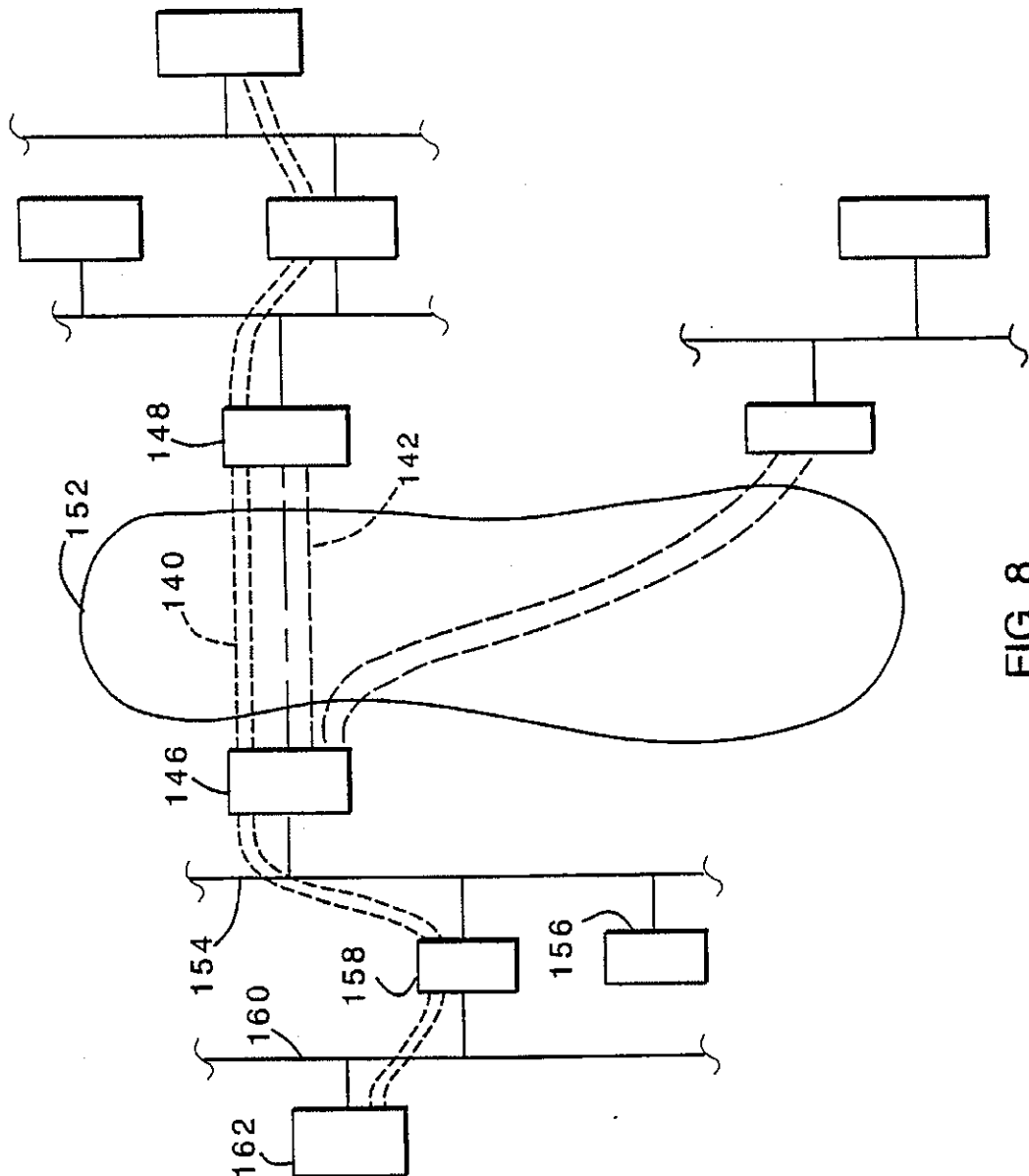


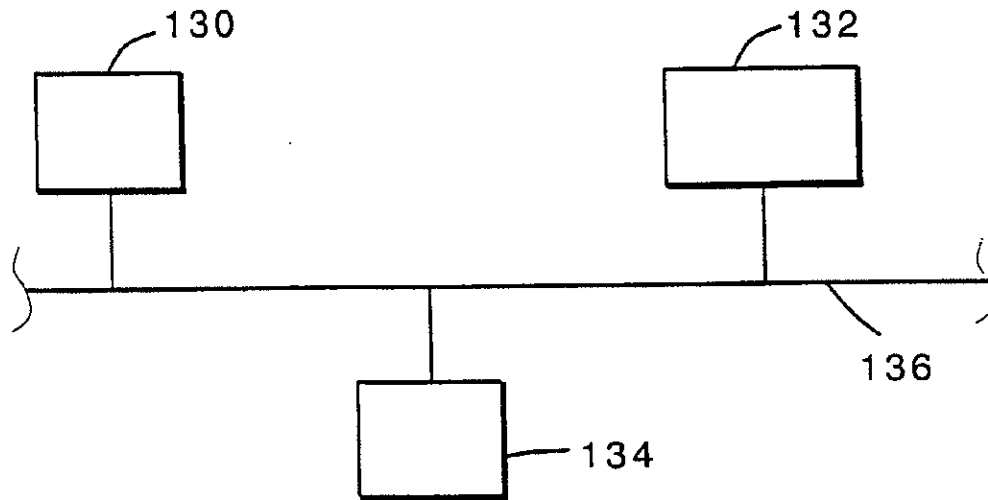
Exhibit 1  
Page 19

**U.S. Patent**

**Apr. 27, 1999**

**Sheet 8 of 10**

**5,898,784**



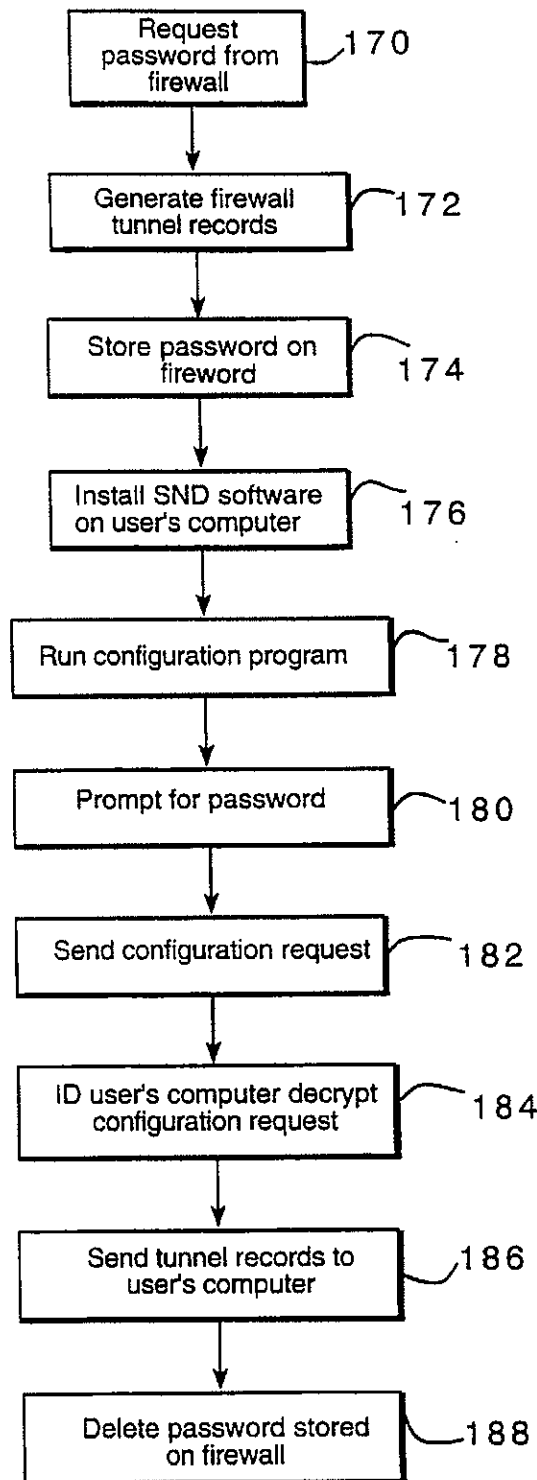
**FIG. 9**

**U.S. Patent**

Apr. 27, 1999

Sheet 9 of 10

**5,898,784**



**FIG.10**

Exhibit 1  
Page 21



U.S. Patent

Apr. 27, 1999

Sheet 10 of 10

5,898,784

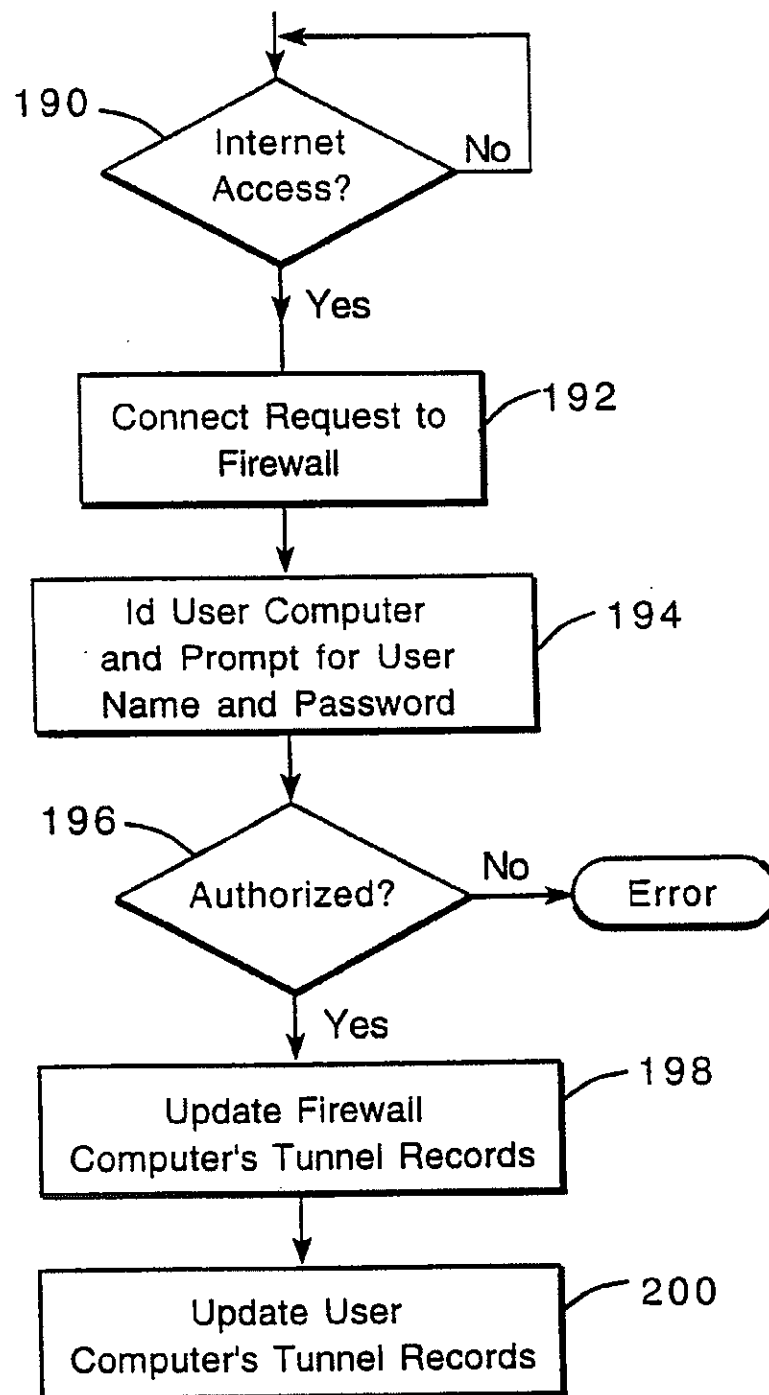


FIG. 11

5,898,784

1

## TRANSFERRING ENCRYPTED PACKETS OVER A PUBLIC NETWORK

This is a continuation of application Ser. No. 08/586,230, filed Jan. 16, 1996, now abandoned.

### BACKGROUND

This invention relates to transferring encrypted packets over a public network.

Referring to FIG. 1, while executing a variety of software applications, 10, 12, 14, for example, Telnet 10 or Microsoft™, Inc. Word™ 12, computers 16 and 18 may exchange data over networks 20, 21, for example, a telephone company network, a private network, or a public network such as the internet or X.25. The applications communicate using network protocols 22, 24, 26, for example, transmission control protocol/internet protocol (TCP/IP) 22 or internet packet exchange (IPX) 24, through application programming interfaces 28, 30, 32. Through application programming interfaces 34, 36, 38, the network protocols communicate with network drivers 40, 42, 44 to direct network interface hardware 46, 48 to transfer data over the networks.

While on a network, data being transmitted, including the addresses of the source and destination computers 16, 18, is accessible to others who may be monitoring the network. For security, the data is often encrypted before being sent on the network.

Referring also to FIG. 2, for additional security, firewall computers 16, 18, which have direct access to a network 20 may be used to prevent unauthorized access to internal/private networks 50, 52. For example, when an internal network driver 53 within firewall computer 16 receives data from an internal computer 54 that is destined for a computer 56 on a public network, it encrypts the data and the addresses of source computer 54 and destination computer 56. Computer 16 then prepends to the encrypted data a new IP header including its own address as well as the address of a destination computer, which may also be a firewall computer, e.g., computer 18.

When a firewall computer receives a network packet from the network, it determines whether the transmission is authorized. If so, the computer examines the header within the packet to determine what encryption algorithm was used to encrypt the packet. Using this algorithm and a secret key, the computer decrypts the data and addresses of the source and destination computers 54, 56 and sends the data to the destination computer. If both the source and destination computers are firewall computers, the only addresses visible (i.e., unencrypted) on the network are those of the firewall computers. The addresses of computers on the internal networks, and, hence, the internal network topology, are hidden. This has been termed "virtual private networking" (VPN).

Encrypting/decrypting data has been performed by complex security software within applications or, to simplify the applications, encrypting/decrypting has been performed within the protocol stack of network protocols.

### SUMMARY

In general, in one aspect, the invention features a method of handling network packets. Encrypted network packets are received from the network at a network interface computer and passed to a computer on an internal network.

Implementations of the invention may include one or more of the following features. Before passing the encrypted

2

network packets to the computer on the internal network, the destination computer for each encrypted network packet is determined. Determining the destination computer may include determining whether a source computer that sent each encrypted network packet is authorized to send encrypted network packets to the destination computer. Determining the destination computer may also include examining a field in a header of the network packet, and the field may correspond to a virtual network tunnel.

An encrypted network packet may be passed to the computer on the internal network if the computer on the internal network is determined to be the destination computer. Instead, the encrypted network packet may be decrypted at the network interface computer when the network interface computer is determined to be the destination computer. Network packets decrypted by the network interface computer may be passed to a computer on an internal network.

The method may also include encrypting network packets and sending the encrypted network packets from the network interface computer to the network. The computer on the internal network may encrypt the network packets, and the method may further include passing the encrypted network packets to the network interface computer. The network interface computer may be a firewall computer, and the network may be a public network.

In general, in another aspect, the invention features receiving encrypted network packets at a first computer over a network from a second computer, and examining a field in each network packet to determine which of a plurality of encryption algorithms was used to encrypt the network packet. The network packet is then decrypted in accordance with the determined encryption algorithm.

Implementations of the invention may include one or more of the following features. The field may be examined to determine a destination computer for each encrypted network packet. A determination may be made as to whether a source computer that sent each encrypted network packet is authorized to send encrypted network packets to the destination computer. Encrypted network packets may be passed to a computer on an internal network when the destination computer is determined to be the computer on the internal network. The network packets may be decrypted when the destination computer is determined to be the first computer, and the decrypted network packets may be passed to a computer on an internal network. The field may correspond to a virtual network tunnel, and the network may be a public network. The first computer may be a firewall computer.

In general, in another aspect, the invention features receiving network packets over a network, and determining which virtual tunnel each network packet was sent over is made. Each network packet is then routed to a destination computer in accordance with the determined virtual tunnel.

Implementations of the invention may include one or more of the following features. Each network packet may be decrypted in accordance with the determined virtual tunnel.

In general, in another aspect, the invention features encrypting network packets at a computer connected to an internal network and passing the network packets over the internal network to a network interface computer. The network interface computer then passes the encrypted network packets over a public network.

In general, in another aspect, the invention features receiving network packets from a network, and determining over which virtual tunnel each network packet was sent. A

5,898,784

3

determination is also made as to whether the source computer that sent each network packet is authorized to send network packets over the determined virtual tunnel.

Implementations of the invention may include one or more of the following features. Each network packet may be routed to a destination computer in accordance with the determined virtual tunnel when the source computer is determined to be authorized.

Advantages of the invention may include one or more of the following. Using the policy id field to create virtual tunnels allows a receiving computer to determine both a packet's encryption algorithm and where the packet should be routed. Multiple tunnels between the same two computers allows packets encrypted with different encryption algorithms to be sent between the same computers. The virtual tunnels permit the encapsulating/decapsulating and encrypting/decrypting of network packets to be spread across multiple computers. Using the tunnel databases, the firewall computers may restrict access to particular tunnels and, in effect, perform packet filtering for each tunnel.

Other advantages and features will become apparent from the following description and from the claims.

#### DESCRIPTION

FIG. 1 is a block diagram of two computers connected together through two networks.

FIG. 2 is a block diagram of two firewall computers and networks.

FIG. 3 is a block diagram of a computer including a security network driver.

FIG. 4 is a flow chart of encapsulation and encryption.

FIGS. 5 and 6 are block diagrams of network packets.

FIG. 7 is a flow chart of decryption and decapsulation.

FIG. 8 is a block diagram of virtual tunnels.

FIG. 9 is a block diagram of a computer network.

FIG. 10 is a flow chart of tunnel record generation.

FIG. 11 is a flow chart of tunnel record updating.

As seen in FIG. 3, security network driver software 72 is inserted between network protocol TCP/IP 22 and corresponding network driver 40. The security network driver encrypts information before it is sent on the network by the network driver and decrypts information received from the network by the network driver before the information is sent to the network protocol. As a result, after choosing a security network driver with the required security features, users may freely choose among available applications and network protocols regardless of the required level of security and regardless of the available encryption/decryption libraries and without having to compromise their security needs. Moreover, the chosen applications and network protocols need not be modified. To change the level of security, the user may simply choose another security network driver or modify the current security network driver.

Generally, a computer's operating system software defines a "road map" indicating which applications may communicate with each other. To insert a security network driver between a network protocol and a network driver, the road map is altered. The vendor of the operating system software may make the road map available or the road map may be determined through observation and testing. Once the road map is altered, functions such as send and receive, between the network protocol and the network driver are diverted to the network security driver to encrypt data before it is sent on the network and to decrypt data when it is received from the network.

4

Referring to FIGS. 3 and 4, as an example, to send data from computer 16 to computer 18 on the internet, Telnet 10 issues (step 60) a send call to TCP/IP 22 through network protocol API 28. The send call includes a network packet 62 (FIG. 5) having a header 64 and data 66. The header includes information such as the addresses of the source and destination computers and the type of application that sent the data. The network protocol then issues (step 68) a send call to the network driver API which, in accordance with the altered road map, issues (step 70) a send call to a security network driver (SND) 72.

The security network driver issues (step 74) an encapsulate call to an encapsulate/decapsulate library 76 through an API 77. In one example, the encapsulate/decapsulate library uses the swiPe IP Security Protocol created by J. Ioannidis of Columbia University and M. Blaze of AT&T<sup>TM</sup>, Inc. which is described in an Internet Draft dated Dec. 3, 1993 and incorporated by reference. Referring also to FIG. 6, the encapsulate call generates a new network packet 78 in accordance with the swiPe protocol. The new packet includes a header 80, a swiPe protocol header 82, and data 84. According to options within the swiPe protocol, header 80 may be the original header 64 (FIG. 5), in which case, data 84 is the original data 66, or header 80 may be a new header including the address of a source firewall computer, e.g., computer 16 (FIG. 2), and a destination computer which may also be a firewall computer, e.g., 18. Where header 80 is a new header, data 84 includes the entire original network packet 62 (FIG. 5).

After encapsulating the network packet, the security network driver issues (step 88, FIG. 4) an encryption call to an encryption/decryption library 90 (FIG. 3) through an API 91. Library 90 encrypts a portion 92 of the encapsulated network packet including data 84 and part of swiPe protocol header 82. Header 80 (FIG. 6) is not encrypted. Thus, if, according to options within the swiPe protocol, header 80 is the original header 64 (FIG. 5), then the addresses of the source and destination computers are visible on the internet. On the other hand, if header 80 is a new header including the addresses of firewall computers, then the addresses of internal source and destination computers are encrypted and not visible on the internet.

Library 90 may be of the type sold by RSA Data Security<sup>TM</sup>, Inc. of Redwood City, California and may encrypt the data according to an RSA algorithm such as RC2 or RC4 or according to a federal information processing standard (FIPS) such as data encryption standard (DES).

The security network driver then issues (step 94) a send call, including the encapsulated/encrypted network packet, to the API, and the API, in accordance with the altered road map, issues (step 96) a send call to a network driver, e.g., network driver 40. The network driver then causes hardware 46 to transmit (step 98) the encapsulated/encrypted network packet on the network.

Referring to FIGS. 3 and 7, the network drivers of each computer 16, 18 (FIGS. 2 and 3) maintain a database of addresses to which they will respond. For example, when network driver 40 receives (step 100) a properly addressed network packet from network 20, the network driver issues (step 102) a receive call to corresponding network protocol API 34. In accordance with the altered road map, the API issues (step 104) a receive call to security network driver (SND) 72 which issues (step 106) an authorization call to encapsulate/decapsulate library 76 through API 77. Library 76 examines the unencrypted portion of swiPe header 82 (FIG. 6) to determine (step 108) whether it is proper. If it is not proper, an error (step 110) is flagged.

5,898,784

5

If the header 82 is not a swIPe header, then the security network driver issues a receive call to the API including the unaltered packet.

If the swIPe header is proper, the security network driver issues (step 112) a decryption call to encryption/decryption library 90 through API 91. A portion of the unencrypted swIPe protocol header includes a policy identification (id) field 113. The policy id field indicates the encryption algorithm used to encrypt the data. Library 90 uses a secret key that was previously exchanged between the computers and the encryption algorithm to decrypt data 84.

After decryption, the security network driver issues (step 114, FIG. 7) a digital signature check call to encapsulate/decapsulate library 76. The swIPe protocol header includes a digital signature 86. The digital signature is a unique number calculated using the data in the network packet, the secret key, and a digital signature algorithm. Library 76 recalculates the digital signature and compares (step 116) it to digital signature 86 in the network packet. If the network packet is tampered with during transmission and any data within the packet is changed, then the digital signature in the packet will not match the digital signature generated by the receiving computer and an error (step 118) will be flagged.

If the signatures match, then the security network driver issues (step 120) a receive call to the API which issues (step 121) a receive call to the TCP/IP network protocol including only the original network packet 62 (FIG. 5, data 66 and addresses of the source and destination computers 64). If (step 122) the network packet is destined for computer 16, then TCP/IP issues (step 124) a receive call to an application 10, 12 and if the network packet is destined for a computer on an internal network, e.g., computer 54 (FIG. 2) on network 50, then TCP/IP issues (step 126) a receive call to internal network driver 53 which then sends (step 128) the data to the internal computer.

Referring to FIG. 8, the policy id field may be used to create virtual tunnels 140, 142 between firewall computers 146, 148 on internet 152. When computer 146 receives a network packet, it checks the policy id to determine which "tunnel" the packet came through. The tunnel indicates the type of encryption algorithm used to encrypt the packet.

Multiple tunnels 140, 142 may connect two computers 146, 148 and each tunnel may use a different encryption algorithm. For example, tunnel 140 may use the RC2 encryption algorithm from RSA Data Security™, Inc. while tunnel 142 uses the FIPS DES encryption algorithm. Because the RC2 encryption algorithm is less secure and requires less computer processing time than the FIPS DES standard, users may send a larger number of network packets requiring less security over tunnel 140 as opposed to tunnel 142. Similarly, predetermined groups of users or computers may be restricted to sending their packets over particular tunnels (effectively attaching a packet filter to each tunnel).

The tunnel may also indicate where the packet is to be sent. Primary firewall computers 16, 18 store information about the internal path of each tunnel in a tunnel database. When computer 146 receives a packet whose policy id indicates that the packet came through a tunnel that ends at computer 146, e.g., tunnel 142, computer 146 decapsulates and decrypts the packet and sends the decrypted packet over internal network 154 to the proper destination computer in accordance with the decrypted destination address. When computer 146 receives a packet whose policy id indicates that it came through a tunnel that does not end with computer 146, e.g., tunnel 140, computer 146 does not decapsulate and decrypt the packet. Instead, computer 146

6

sends the encrypted packet to internal firewall computer 158 in accordance with the tunnel database.

Internal firewall computer 158 also has a tunnel database in which the internal path of any tunnels connected to computer 158 are stored. As a result, when computer 158 receives a packet whose policy id indicates that it came through a tunnel that ends with computer 158, e.g., tunnel 140, it decapsulates and decrypts the packet according to the policy id and sends the decrypted packet over internal network 160 to computer 162 in accordance with the decrypted destination address.

The only addresses visible on the internet and on internal network 154 are the addresses of the firewall computers 146, 148, and 158. The address of internal computer 162 and, hence, the network topology of network 160 are protected on both the internet and internal network 154.

The tunnel databases provide the firewall computers 146, 148, and 158 with information as to the internal path of the tunnels. Thus, if computer 162 was another firewall computer, computer 146 may modify the destination address of packets received on tunnel 140 to be the address of computer 162 to cause computer 158 to send the packet directly to computer 162 without checking the policy id field.

Encapsulating/decapsulating and encrypting/decrypting network packets may require a large portion of a computer's processing power. Creating virtual tunnels using the policy id field allows the encapsulating/decapsulating and encrypting/decrypting of network packets to be spread across several computers. For example, computer 146 may decapsulate and decrypt network packets destined for computers connected to internal network 154 while computer 158 may decapsulate and decrypt network packets destined for computers connected to internal networks 154 and 160. Similarly, computer 146 may encapsulate and encrypt network packets sent from computers connected to internal network 154 while computer 158 may encapsulate and encrypt network packets sent from computers connected to internal networks 154 and 160.

The Kerberos Key Distribution Center components of Kerberos Network Authentication System created under project Athena at Massachusetts Institute of Technology, defines one method of providing computers with secret keys. Referring to FIG. 9, computer 130 is termed the "trusted" computer, and before computers 132 and 134 may transfer encrypted data to each other over network 136, both computers send a request to trusted computer 130 for a secret key. For a more detailed description of the Kerberos Key Distribution Center, see RFC1510 (request for comment) "Kerberos Network Authentication Service" by J. Kohl & B. Neuman, Sept. 10, 1993, which is incorporated by reference.

Referring back to FIG. 2, to transfer secure (i.e., encapsulated and/or encrypted) network packets between two computers, operators of the two computers may verbally exchange a secret key for each tunnel between the computers and then manually initialize the computers to transfer data by generating a tunnel record including a secret key for each tunnel between the two computers. Firewall computers are typically managed by skilled technicians capable of generating tunnel records. Typical users have non-firewall computers and may wish to transfer encapsulated/encrypted data with a firewall computer. To avoid requiring that a typical user generate tunnel records and instead of having a separate trusted computer provide secret keys to two computers, a firewall computer 16, 18 may provide secret keys to other computers.

Exhibit

1

Page

25



5,898,784

7

Referring also to FIG. 10, when a user wishes to transfer packets between his/her computer and a firewall computer, the user requests (step 170) a password (a onetime pad) from the firewall operator. The operator then generates (step 172) tunnel records for each tunnel over which the user's computer and the firewall computer may transfer network packets. The operator also stores (step 174) the password given to the user on the firewall computer. The user installs (step 176) the security network driver (SND) software on his/her computer and runs (step 178) a configuration program. The configuration program prompts (step 180) the user for the password and sends (step 182) a configuration request to the firewall computer.

The firewall computer identifies (step 184) the user's computer as the sender of the request and notifies the user's computer of the available tunnels by sending (step 186) the complete tunnel records, including secret keys, associated with each tunnel to the user's computer. The tunnel records are sent through network packets that are encrypted using the password and the encryption algorithm. Afterwards, the firewall deletes (step 188) the password, and further network packets are transmitted between the two computers through the available tunnels and encrypted according to the secret key associated with each tunnel.

Referring to FIG. 11, generally, each time the user's computer accesses (step 190) the internet, a new internet address is assigned. The firewall computer needs to know the new address in order to update the tunnel records. To notify the firewall computer of the new internet address, each time the user's computer accesses the internet, the configuration software issues (step 192) a connect request to the firewall computer. The firewall computer identifies (step 194) the computer and may prompt the user for a user name and a user password. If the user name and password are authorized (step 196), the firewall updates (step 198) the tunnel records with the internet address sent as part of the connect request. The configuration software also updates (step 200) the non-firewall computer's tunnel records with the computer's new internet address.

Other embodiments are within the scope of the following claims.

For example, instead of encapsulating the network packets using the swiPe protocol header, other internet security algorithms may be used.

Although the security network driver was described with respect to send and receive functions, APIs from different manufacturers, for example, Sun™, Inc. and Microsoft™, Inc., include a variety functions, and the security network driver is designed to respond to each possible function.

The security network driver may also be simultaneously connected to multiple network protocols, e.g., both TCP/IP 22 and IPX 24, as shown in FIG. 3.

What is claimed is:

1. A method of handling network packets, comprising:
  - receiving an encrypted network packet from an external network at a first computer; and
  - determining whether to decrypt the encrypted network packet at the first computer or to pass the encrypted network packet to a computer on a network that is internal with respect to the first computer for decryption.
2. The method of claim 1, further comprising, before passing the encrypted network packet to the computer on the network that is internal with respect to the first computer
  - determining a destination computer for the encrypted network packet.

8

3. The method of claim 2, wherein determining a destination computer further includes:

determining whether a source computer that sent the encrypted network packet is authorized to send encrypted network packets to the destination computer.

4. The method of claim 2, wherein determining a destination computer includes:

examining an index field in a header of the network packet.

5. The method of claim 4, wherein the field corresponds to a virtual network tunnel.

6. The method of claim 2, wherein an encrypted network packet is passed to the computer on the network that is internal with respect to the first computer when the destination computer for the encrypted network packet is determined to be the computer on the network that is internal with respect to the first computer.

7. The method of claim 1, further comprising:

decrypting an encrypted network packet at the first computer when the destination computer for the encrypted network packet is determined to be the first computer.

8. The method of claim 7, further comprising:

passing the decrypted network packet to the computer on the network that is internal with respect to the first computer.

9. The method of claim 1, further comprising:

encrypting network packets; and  
sending encrypted network packets from the first computer to the external network.

10. The method of claim 9, wherein the computer on the network that is internal with respect to the first computer encrypts the network packets, and further comprising:

passing the encrypted network packets to the first computer.

11. The method of claim 1, wherein the first computer comprises a firewall computer.

12. The method of claim 1, wherein the external network comprises a public network.

13. A method of handling network packets, comprising receiving an encrypted network packet from a public network at a firewall computer;

determining the destination computer of the encrypted network packet by examining a virtual tunnel field that corresponds to the method of encryption;

determining whether a source computer that sent the encrypted network packet is authorized to send encrypted network packets to the destination computer; and

determining whether to decrypt the encrypted network packet at the firewall computer or to pass the encrypted network packet to a computer on a network that is internal with respect to the first computer for decryption.

14. A method of handling a network packet, comprising receiving an encrypted network packet at a first computer over a network from a source computer;

examining a field in the network packet to determine which of a plurality of encryption algorithms was used to encrypt the network packet and to determine a destination computer for each encrypted network packet; and

decrypting the network packet at the determined destination computer.

15. The method of claim 14, further comprising:

determining whether a source computer that sent each encrypted network packet is authorized to send encrypted network packets to the destination computer.

Exhibit

1

Page

26

5,898,784

9

16. The method of claim 14, further comprising:  
passing encrypted network packets to a computer on an  
internal network when the destination computer is  
determined to be the computer on the internal network.
17. The method of claim 14, further comprising: 5  
decrypting network packets when the destination com-  
puter is determined to be the first computer.
18. The method of claim 17, further comprising:  
passing the decrypted network packets to a computer on 10  
an internal network.
19. The method of claim 14, wherein the field corresponds  
to a virtual network tunnel.
20. The method of claim 14, wherein the network com-  
prises a public network.
21. The method of claim 14, wherein the first computer 15  
comprises a firewall computer.
22. A method of handling an encrypted network packet,  
comprising:  
receiving the encrypted network packet sent over a net- 20  
work at a first computer;  
determining which virtual tunnel the network packet was  
sent over; and  
routing the network packet to a destination computer that  
is internal with respect to the first computer in accor- 25  
dance with the determined virtual tunnel.
23. The method of claim 22, further comprising:  
decrypting each network packet in accordance with the  
determined virtual tunnel.

10

24. A method of handling a network packet, comprising:  
encrypting network packets at a first computer connected  
to an internal network;  
storing a virtual tunnel identifier in the packet that is used  
to determine routing of the packet;  
passing the encrypted network packet over the internal  
network to a public network interface computer; and  
passing the encrypted network packet over a public net-  
work connected to the public network interface com-  
puter.
25. A method of handling network packets, comprising:  
receiving network packets sent over a network at a first  
computer;  
examining each packet's virtual tunnel field to determine  
which virtual tunnel each network packet was sent over  
and whether a source computer that sent each network  
packet is authorized to send network packets over the  
determined virtual tunnel.
26. The method of claim 25, further comprising:  
routing each network packet to a destination computer in  
accordance with the determined virtual tunnel when the  
source computer is determined to be authorized.

\* \* \* \* \*

Exhibit

1

Page

27

# **Exhibit 2**





US005996011A

**United States Patent** [19][11] **Patent Number:** **5,996,011****Humes**[45] **Date of Patent:** **Nov. 30, 1999**[54] **SYSTEM AND METHOD FOR FILTERING DATA RECEIVED BY A COMPUTER SYSTEM**[75] **Inventor:** Donald Creig Humes, Newport News, Va.[73] **Assignee:** Unified Research Laboratories, Inc., Hampton, Va.[21] **Appl. No.:** 08/823,123[22] **Filed:** Mar. 25, 1997[51] **Int. Cl.<sup>6</sup>** ..... G06F 13/00[52] **U.S. Cl.** ..... 709/225; 709/232; 709/206; 713/201[58] **Field of Search** ..... 395/200.47, 200.48, 395/200.49, 200.54, 200.55, 200.59, 187.01, 188.01; 707/104, 9; 709/217, 218, 219, 224, 225, 229, 232; 713/201, 202[56] **References Cited****U.S. PATENT DOCUMENTS**

4,839,853	6/1989	Deerwester et al. .
4,849,898	7/1989	Adi .
5,056,021	10/1991	Ausborn .
5,128,865	7/1992	Sadler .
5,255,386	10/1993	Prager .
5,297,039	3/1994	Kanaegami et al. .
5,317,507	5/1994	Gallant .
5,331,554	7/1994	Graham .
5,418,948	5/1995	Turtle .
5,471,610	11/1995	Kawaguchi et al. .
5,576,954	11/1996	Driscoll .
5,598,557	1/1997	Doner et al. .
5,706,507	1/1998	Schlöss .
5,784,564	7/1998	Camaisa et al. ....
5,796,948	8/1998	Cohen .
5,825,722	11/1998	Bradshaw et al. ....
5,832,212	11/1998	Cragun et al. ....
5,884,033	3/1999	Duvall et al. ....

707/104

395/200.54

395/200.55

395/200.55

395/200.55

709/225 X

**OTHER PUBLICATIONS**

Charkravarthy et al., "NetSurf: Using Semantic Knowledge to Find Internet Information Archives", SIGIR '95, pp. 4-11.

Voorhees, "Using WordNet to Disambiguate Word Senses for Text Retrieval", SIGIR '93, pp. 171-180.

Jacquemin et al., "Retrieving Terms and their Variants in a Lexicalized Unification-Based Framework", SIGIR '94, pp. 132-141.

"Net Nanny", Trove Investment Corporation—Home Page, pp. 1-4/www.netnanny.com.

Venditto, "Safe Computing", Internet World, Sep. 1996, pp. 49-58.

Liddy et al., "Dr-Link, Document Retrieval Using Linguistic Knowledge Project Description", pp. 39-43.

EchoSearch Homepage, pp. 1-2. <http://www.iconovex.com/Echo/Echos.HTM>.

Antonacci et al., "Capturing the deep meaning of texts through deduction and inference", IBM J. Res. Develop., vol. 36, No. 3, May 1992, pp. 333-344.

"SurfWatch ProServer from Spyglass", pp. 1-4. <http://www.spyglass.com/>.

Syndex: Parent's Guide, "Protecting Your Children Via Content Filtering", pp. 1-2. <http://www.syndex.com:80/>.

Retkwa, "Corporate Censors", Internet World, Sep. 1996, pp. 60-64.

*Primary Examiner*—Zarni Maung  
*Assistant Examiner*—Patrice Winder  
*Attorney, Agent, or Firm*—Larson & Taylor

[57] **ABSTRACT**

The present invention provides a system and method for restricting access to data received by a computer over a network by filtering certain data from the data received. In a preferred embodiment, the present invention provides a computer based method for filtering objectionable or target text data from World Wide Web pages which are received by a computer system connected to the Internet. According to the method, if the web page requested by the user contains only a minimum of objectionable or target data, the user may receive a portion of the filtered web page for downloading and viewing on his or her computer. If the web page requested contains a large amount of objectionable or target data, the invention will cause a "forbidden" page to be displayed on the user's computer monitor.

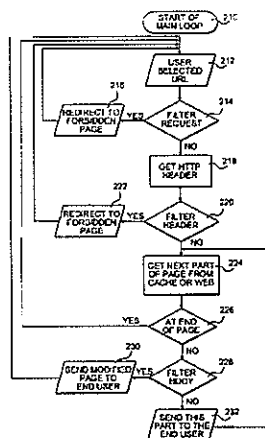
**26 Claims, 5 Drawing Sheets**

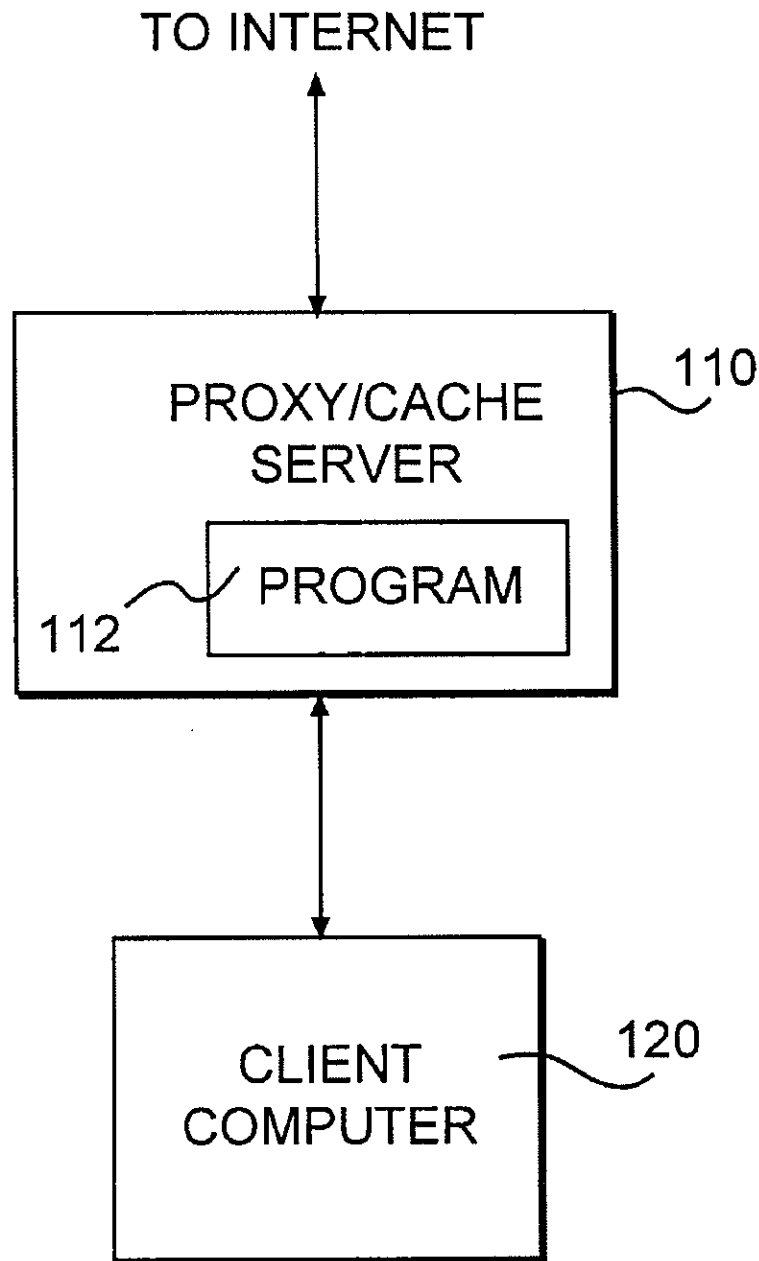
Exhibit 2  
 Page 28

U.S. Patent

Nov. 30, 1999

Sheet 1 of 5

5,996,011



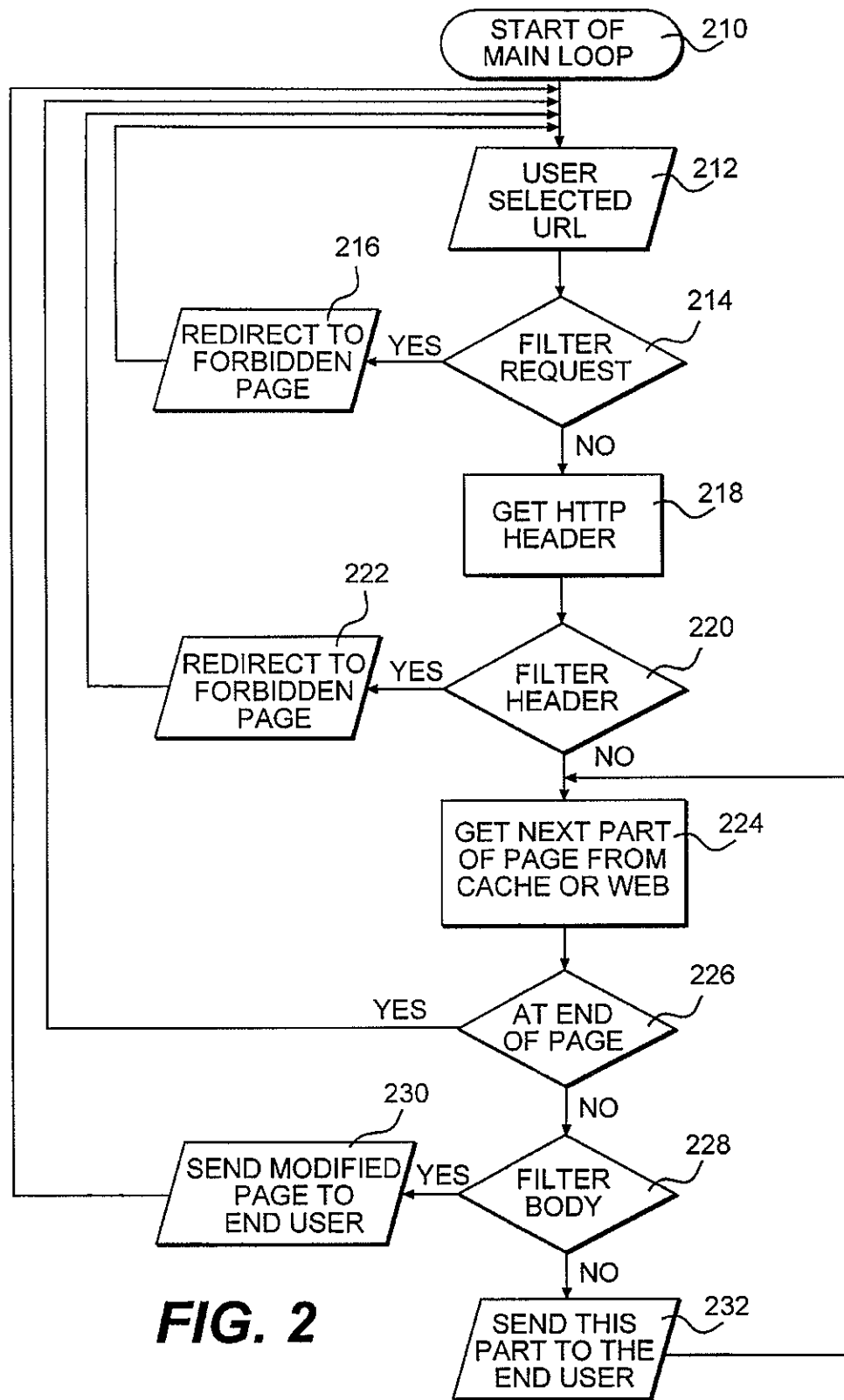
**FIG. 1**

U.S. Patent

Nov. 30, 1999

Sheet 2 of 5

5,996,011

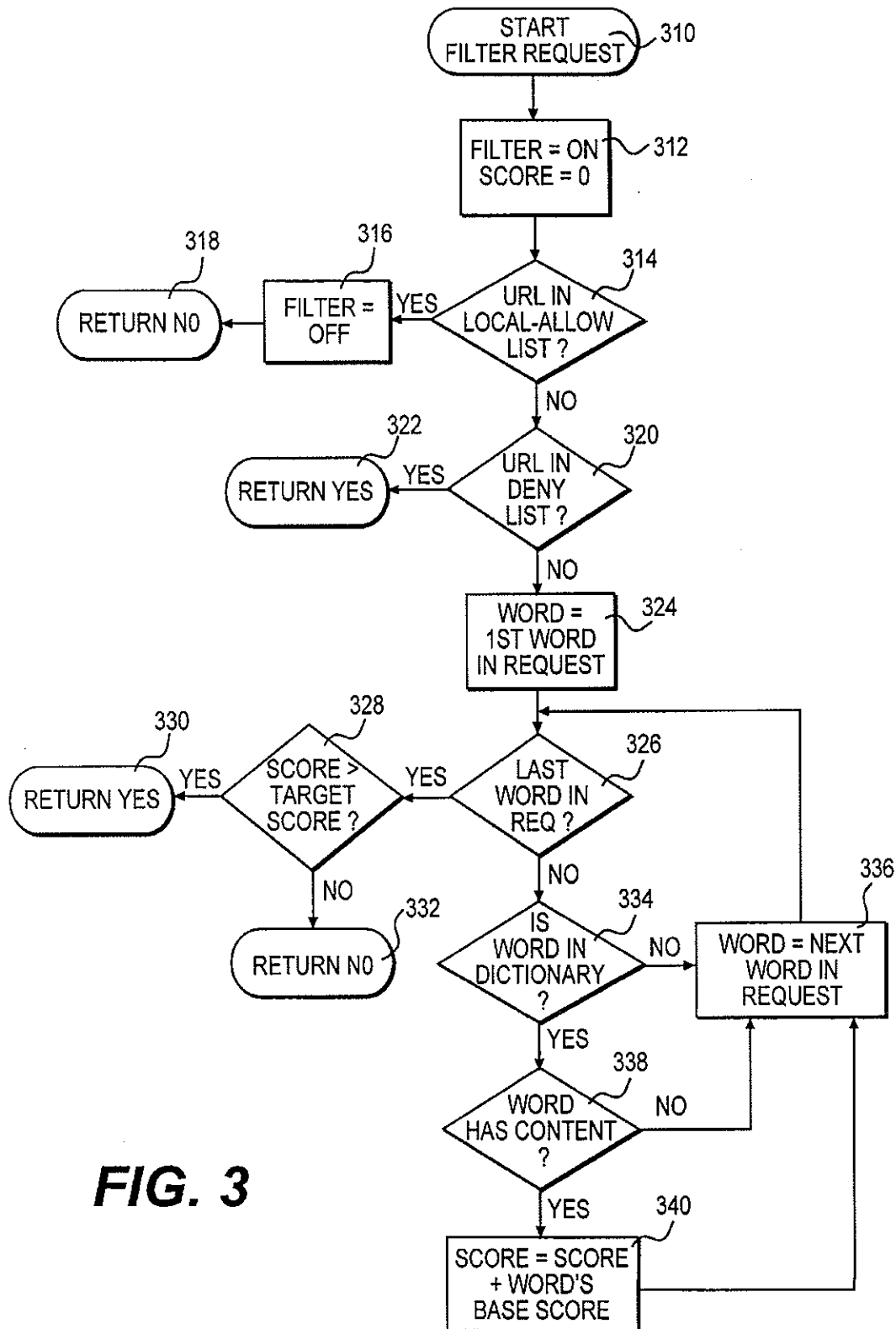
**FIG. 2**

U.S. Patent

Nov. 30, 1999

Sheet 3 of 5

5,996,011

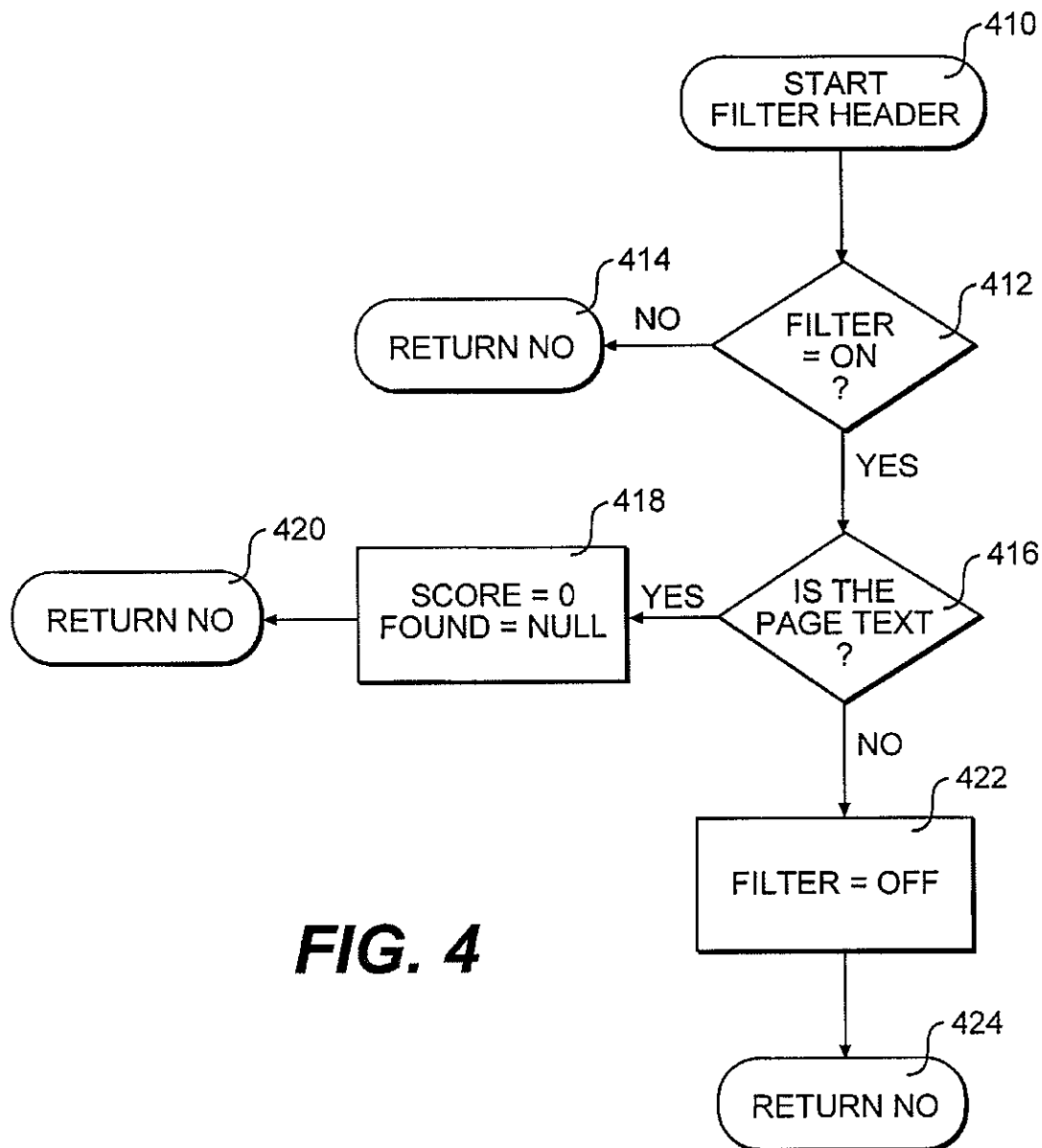
**FIG. 3**

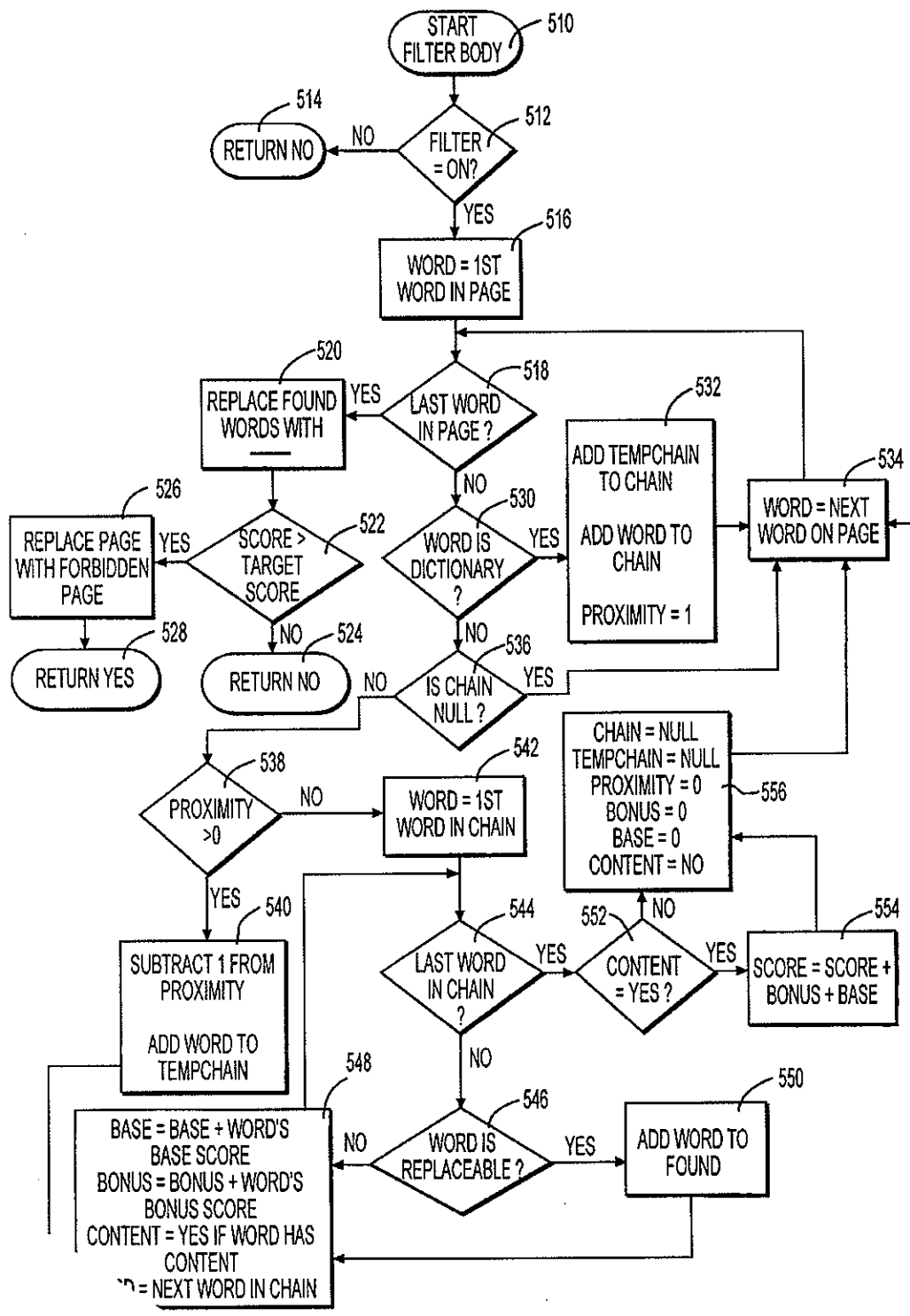
U.S. Patent

Nov. 30, 1999

Sheet 4 of 5

5,996,011

**FIG. 4**



**FIG. 5**

5,996,011

1

# SYSTEM AND METHOD FOR FILTERING DATA RECEIVED BY A COMPUTER SYSTEM

## BACKGROUND OF THE INVENTION

### 1. Field of the Invention

The present invention relates to a computer based system and method for filtering data received by a computer system and, in particular, to a computer based system and method for filtering text data from World Wide Web pages received by a computer system connected to the Internet.

### 2. Prior Art

While there are numerous benefits which accrue from the interconnection of computers or computer systems in a network, such an interconnection presents certain problems as well.

Broadly speaking, networks allow the various computers connected to the network to share information. Typically, if desired, access to certain information is restricted by providing access codes or the like to those individuals who are cleared to view or download the information. While this method of controlling access to information works fairly well for situations where each user is identifiable, it is very difficult to efficiently and effectively implement such a method in cases where there is a large number of unidentifiable users. Such is the situation with the vast interconnection of networks called the Internet.

The Internet is accessed by many millions of users every day and while it is somewhat possible to obtain some information with respect to identifying the computers through which a particular user accesses the Internet, it is very difficult, if not impossible, to identify a particular user beyond any self-identification provided by the user himself.

By far, most of the traffic on the Internet currently occurs on the World Wide Web. On the World Wide Web, both text and graphic information is typically provided on web pages and this information is transmitted via the Hyper Text Transfer Protocol ("HTTP"). A web page has a particular address associated with it called a Uniform Resource Locator ("URL").

A typical user accesses the World Wide Web via a modem connection to a proxy/cache server which is connected to the Internet. A browser is the software program which runs on the user's computer (client computer) and allows the user to view web pages. To view a particular web page, the user inputs the URL of the desired web page into his or her browser. The browser sends the request to the proxy/cache server and the server sends the request over the Internet to the computer on which the web page resides. A header as well as a copy of the body of the web page is then sent back to the user's browser and displayed on the user's computer.

While an incredible amount of information is available on the millions of web pages provided on the World Wide Web, some of this information is not appropriate for all users. In particular, although children can be exposed to a vast number of educational and entertaining web pages, many other web pages include adult content which is not appropriate for access by children.

One method which is used to control access to these adult web pages is to require an access code to view or download particular web pages. Typically, this access code is obtained by providing some identification, often in the form of a credit card number. The obvious drawbacks of this method are: 1) such a system will invariably deny or inhibit access to many adults as well as children because many adults do

2

not want to, or may not be able to, provide a credit card number; and 2) the system is not fool-proof because children may obtain access to credit cards, whether their's or their parents'.

Several services are available to parents and educators which provide a second method for preventing access to web pages containing adult content. These services provide software programs which contain a list of forbidden URLs. Service providers compile the list by searching the World Wide Web for web pages having objectionable material. When a user inputs a URL which appears on the forbidden list or "deny list," the program causes a message to be displayed indicating that access to that web page is forbidden. Although this method works well for denying access to web pages which are on the forbidden list, because thousands of web pages are being created and changed every day, it is simply impossible to provide an up-to-date list of every web page containing adult content. Therefore, these systems often allow children access to web pages which contain adult content but have not yet been added to the forbidden list.

A further drawback to the above-described access control systems is that they are simple admit/deny systems. That is, the user is either allowed to download and view the entire web page or he/she is forbidden from doing so. It is not practical, using either of these methods, to allow a particular user to download and view only the portions of the web page which are not objectionable.

## SUMMARY OF THE INVENTION

The present invention overcomes the disadvantages of the prior art by providing a system and method for restricting access to objectionable or "target" data received by a computer over a network by filtering objectionable data from the data received. The present invention provides for filtering the data as received, so called "on the fly," so that a newly created web page may be filtered as accurately as one that has been predetermined to contain objectionable material.

Although the embodiments of the invention are described below with respect to a system and method for filtering objectionable data from the data received, it should be understood that the present invention can be applied to process any type of target data from the data received. Thus, the present invention may be utilized to process desired data such that, for instance, only Web pages containing desired information are displayed on the user's computer.

In a preferred embodiment, the present invention provides a computer based method for filtering text data from World Wide Web pages which are received by a computer system connected to the Internet. Advantageously, the method of the present invention is carried out by the computer which acts as the proxy/server through which the user's computer is connected to the Internet. However, the method can be carried out by the user's computer as well.

According to the method, if the web page requested by the user contains only a minimum of objectionable or target data, the user receives a portion of the filtered web page for downloading and viewing on his or her computer. While, if the web page requested contains a large amount of objectionable material, the invention will cause a "forbidden" page to be displayed on the user's computer monitor.

In the preferred embodiment, the request is sequentially filtered at three different levels, if necessary. First, the URL requested is filtered to determine if the web page associated with that URL has been pre-approved or pre-denied. If the URL has not been pre-approved or pre-denied, the header of

Exhibit

2

Page

34



5,996,011

3

the web page is then filtered to determine if the web page contains text data (such as HTML). If so, the body of the web page is filtered. While the filter will decide whether or not to block access to the entire web page based on the URL, depending on its processing of the body of the web page, the filter may deny access completely to the web page, deny access to certain portions of the web page (i.e., filter out some objectionable words), or allow complete access to the web page.

The method of the present invention first compares the requested URL to an "allow list" which contains URLs of web pages which have been approved for display to the user. If the requested URL is found in the allow list, the entire associated web page is, accordingly, forwarded to the user for downloading or viewing. If, however, the requested URL is not found in the allow list, the requested URL is then compared to a "deny list," (or "forbidden list") which functions in much the same manner as that of the prior art systems. If the requested URL is found in the forbidden list, a message is transmitted to the user's computer indicating that access to the web page is forbidden (hereinafter referred to as a "FORBIDDEN" page).

If the requested URL is found in neither the allow list or the deny list, and if the header indicates that the page contains text data, then the method provides for filtering the text of the web page, as it is either received from the network or read out of cache, to determine if it contains objectionable or target text. If the page contains objectionable text, the method determines what kind of objectionable text (specific words), how much objectionable text, and the relative groupings of objectionable text. Depending on the settings of predetermined parameters, certain objectionable words (if found) are either replaced with an innocuous filler (such as "- - -" before the web page is forwarded to the user's computer, or only a "FORBIDDEN" page is forwarded to the user's computer. The settings of the predetermined parameters may be modified by those having access to the computer program through which the computer implements the program, such as the server operator or, perhaps, the user's parent.

Optionally, the HTTP header of the web page is filtered after the URL to determine if the page contains text data and, if not, the method does not filter the web page body, since the method for filtering the web page body is only capable of filtering text or other recognizable data patterns. The method provides for filtering the text of the web page by comparing each "word" (defined by groupings of letter/number D characters) in the web page to a "dictionary." The words in the dictionary are periodically updated.

Advantageously, each word in the dictionary has a number of variables associated with it, such as: 1) a variable that indicates whether the word, if found, should be replaced with the innocuous filler (or a specific replacement filler word may be indicated); 2) a variable that indicates what category of objectionableness the word belongs to (i.e., pornography, intolerance, crime, job hunting, etc.); 3) a variable that indicates what language the word is a part of (i.e., english, french, spanish, etc.); 4) a base score variable that indicates how objectionable the word is; and 5) a bonus score variable that indicates whether the word is more objectionable when used in combination with other objectionable words. In this advantageous embodiment, the method provides for filtering the body of the web page by comparing each word in the web page with the words in the dictionary. If a word in the web page matches, then that word will either be replaced or not replaced with the filler, as indicated by the variable. A running score is determined for

4

the entire web page, based on a particular algorithm, as the page is being filtered. If the final score for the page is above a predetermined threshold score, a "FORBIDDEN" page is forwarded to the user's computer instead.

The system of the present invention comprises a general purpose computer which is programmed to implement the method of the present invention. The computer of such a system is typically the proxy/cache server computer but it may also be the client computer or another computer.

While the preferred embodiment of the present invention is described in summary form above as applied to the filtering of web pages received over the Internet, it will be appreciated by one of ordinary skill in the art that this method is also applicable to filtering of data received by a computer from any network, including an intranet. Further, in addition to reviewing HTTP information received from the Internet, the present invention may be applied to review information posted to forms-based pages such as search engines, surveys, guest books, etc. (POST/GET data). If the words or phrases would yield objectionable results, the invention will prevent posting of the data to the remote HTTP server.

Other objects, features, and advantages of the present invention will be set forth in, or will become apparent from, the detailed description of the preferred embodiments of the invention which follows.

#### BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 shows a block diagram of a preferred embodiment of the system of the present invention.

FIG. 2 is a flowchart showing the broad steps of filtering a web page URL, header, and body in accordance with one embodiment of the method of the present invention.

FIG. 3 is a flowchart showing the steps for filtering a URL request in accordance with the embodiment of FIG. 2.

FIG. 4 is a flowchart showing the steps for filtering a web page header in accordance with the embodiment of FIG. 2.

FIG. 5 is a flowchart showing the steps for filtering a web page body in accordance with the embodiment of FIG. 2.

#### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

FIG. 1 shows a block diagram of a preferred embodiment of the system of the present invention. In this embodiment, a proxy/cache server computer 110 is connected to the Internet and is capable of receiving HTTP information over the World Wide Web. A client computer 120 (user's computer) is connected to the server computer 110, typically via an ethernet or modem connection. In accordance with the present invention, server computer 110 runs a computer program 112 which programs the server computer 110 to filter any request it receives for a web page from the client computer 120 and to output to the client computer 120 only those web pages, or portions of web pages, which are deemed appropriate for viewing by the user of the client computer 120. This filtering takes place in at least three stages, as is described below relative to FIGS. 2 through 5, which illustrate the method of the present invention.

Advantageously, the proxy/cache server 110 used in the system of the present invention is a SPARC workstation made by D Sun Microsystems, Inc. and the server 110 is programmed to filter the requests in the Perl programming language. The inventors of the present invention have determined that a SPARC workstation programmed in accordance with the method set forth below, is capable of filtering

Exhibit 2

Page 35

5,996,011

5

at least approximately 90 KB to 500 KB of data per second, based on the speed of the CPU.

The flowchart in FIG. 2 shows the broad steps of filtering a requested web page URL, header, and body in accordance with one embodiment of the method of the present invention. The method begins at terminal block 210 and continues to block 212 where the user selected URL is input. The URL request is filtered at decision block 214 and the filter decision is "yes" if the request is denied based on the URL and "no" if the request is not denied at this point. If decision block 214 returns a "yes," block 216 shows that a page indicating access is forbidden ("FORBIDDEN" page) is to be returned, which page will be output to the client computer. If decision block 214 returns a "no," the HTTP header is input at block 218 and filtered at decision block 220. If decision block 220 returns a "yes," block 222 shows that the "FORBIDDEN" page is to be returned and if decision block 220 returns a "no," the first portion of the body of the web page is input at block 224.

Decision block 226 checks to see if the end of the page has been reached and, if so, the method returns to await another user selected URL. If the end of the page has not been reached, that portion of the body of the web page is filtered at decision block 228. If decision block 228 returns a "yes" (objectionable material found), then that portion of the web page is modified and, if certain rules are met, it is later sent to the client computer, indicated at block 232. If decision block 228 returns a "no," indicating no objectionable material was found in that portion of the web page body, then control is returned to block 224 where the next portion of the web page body is input.

The filters indicated in decision blocks 214, 220, and 228 are shown in the flowcharts of FIGS. 3, 4, and 5, respectively. The filters shown in FIGS. 3 and 5 are based on a method involving a dictionary, which dictionary can be updated periodically, and a score threshold for indicating whether or not a web page should be forbidden.

Each word or phrase in the dictionary has several variables associated with it. Together each word in the dictionary and its associated variables form an entry in the dictionary. In a preferred embodiment, each entry in the dictionary has the following format:

Target:Replace:Category:Language:BaseScore:BonusScore where "Target" is the objectionable or target word. "Replace" indicates whether or not the word or phrase should be replaced in the text if it is found (R=replace, N=not replace) "Category" is the category the word belongs to (e.g., pornography, intolerance, crime, violence, etc.), and "Language" is the language the word belongs to (e.g., english, french, spanish, etc.). "BaseScore" is the score the word contributes to the total score of the web page body, or portion thereof, if it is found and "BonusScore" is the additional score the word contributes to the total score if it is found in close proximity to other objectionable words. A word may be assigned a negative score, which would subtract from the total score instead of adding to it.

In the preferred embodiment, the default threshold score at which a web page will be forbidden is 50, although this score can readily be changed by anyone having the required access.

Examples of possible words and associated variables in the dictionary are as follows.

nude:N:pornography:english:5:5

This entry would be for the word "nude." "Nude," if found would not be replaced in the text (N); it is in the pornography category; it is in the english language; it has a

6

score of 5; and it contributes a bonus score of 5 if found with other words in the dictionary.

hot:N:none:english:0:5

This entry, for the word "hot," is not to be replaced if found; is in no particular category; is in the english language; has no score; and has a bonus score of 5.

The flowchart of FIG. 3 shows the portion of the method which is represented by decision block 214 in FIG. 2, showing filtering of the URL request. This portion of the method begins at terminal block 310 and in block 312 a variable called "filter" is turned on and a variable called "score" is set to zero. The "filter" variable is used to later indicate whether the header or body of the web page should be filtered (on) or not filtered (off). "Score" is the variable which keeps track of the total score of the URL being filtered.

Decision block 314 asks whether the requested URL is in the Local-Allow list, which is a list of URLs associated with web pages that have been pre-approved so that they do not need to be filtered. In addition to speeding up transmission by allowing the request to be passed without filtering the web page, the Local-Allow list also provides for allowing full access to web pages which have text that would normally be filtered. This may prove advantageous, for instance, if a web page discusses certain objectionable text in a manner to educate children about its effects.

If the URL is in the Local-Allow list, the Filter variable is set to "Off" in block 316 and a "No" is returned in terminal block 318. If the URL is not in the Local-Allow list, decision block 320 checks to see if the URL is in the Deny List. The Deny List is a listing of URLs associated with web pages which have been predetermined to be objectionable. If the URL is in this list, terminal block 322 returns a "Yes." This Deny List feature is advantageous inter alia for designating web pages which may contain objectionable material other than text which may be filtered, such as objectionable pictures.

If the URL is not in the Deny List, the individual words in the URL and POST/GET data are then filtered. Block 324 indicates that the variable "Word" is set to the first word in the request. Decision block 326 checks to see if the last word in the request has already been filtered and, if so, decision block 328 determines whether the "Score" of the request at this point is greater than the predetermined threshold "Targetscore," which threshold may, advantageously, be changed by one having the appropriate access. If the "Score" variable is not greater than the "Targetscore" threshold, then terminal block 332 returns a "No," indicating that the page should not be forbidden. If the "Score" variable is greater than the "Targetscore" threshold, then terminal block 330 returns a "Yes," indicating the page should be forbidden. If decision block 326 indicates that the last word in the request has not already been filtered, then decision block 334 compares "Word" against the dictionary. If "Word" does not match an entry in the dictionary, then "Word" is set to the next word in the request and decision block 326 again checks to see if the last word has been filtered. If "Word" does match an entry in the dictionary, then decision block 338 determines whether "Word" has content, that is, is the category variable not set to "none" or, alternatively is the category variable set to a particular category of interest (for instance, "pornography"). If "Word" does not have content (category =none), then, in block 340, "Word" is again set to the next word in the request and, if "Word" does have content (category ≠"none" or category=a particular category), "Score" is set to the previous value of "Score"

Exhibit

2

Page

36

5,996,011

7

plus the base score of the word and "Word" is then set to the next word in the request. Control returns to decision block 326.

The filtering method for the header of the web page, indicated at decision block 220 of FIG. 2, is shown in detail in FIG. 4. This method essentially determines if the page is in text (such as HTML) such that it may be effectively reviewed by the filter.

The method begins at terminal block 410 and decision block 412 determines whether the "Filter" variable is set to "On." If the "Filter" variable is not set to "On," indicating that the header should not be filtered, then terminal block 414 returns a "No." If the "Filter" variable is set to "On," then decision block 416 determines whether the page contains text data and, if so, the "Score" variable is set to zero and the "Found" variable is set to Null at block 418, and terminal block 420 returns a "No," indicating the page should not be forbidden based on the header. If decision block 416 determines that the page does not contain text data, then the "Filter" variable is set to "Off" in block 422 and terminal block 424 returns a "No."

The filtering method for the body of the web page, uses the dictionary and the variables described above, however this method introduces new variables as well which enable it to consider the proximity of the objectionable words to each other, as well as other factors, in deciding whether or not to deny access to the page.

In this method, the body of the web page is filtered one word at a time. When a word is found which is in the dictionary, a "Chain" is started. Succeeding sequential words that are found in the dictionary are included in the "Chain" and the "Chain" is ended when two words in a row are not found in the dictionary. Accordingly, these last two words are not part of the "Chain". Thus, the determination of the "Chain" length is controlled by a "Proximity" variable.

After a "Chain" is formed, it is scored. To score a "Chain," all the base scores for each word in the "Chain" are added together. The bonus scores for each word are added to the total as well, if the "Chain" contains more than one word in the dictionary. If any of the words in the "Chain" have content, i.e., category not set to "none" or category set to a particular category, then the "Chain's" score is added to the total score for the page. Advantageously, a total score is kept for each category and specific thresholds are set for each category.

If the "Replace" variable for any of the words is "R," indicating replace, then the word is replaced with an innocuous word or symbol (e.g., "- - -"). Advantageously, the "Replace" variable for each word in the dictionary may instead be an alternate word which is used to replace the objectionable word, e.g., "damn" for "damn". Such replacement words may be displayed in an alternate color.

If the total score for the page exceeds the predetermined threshold, e.g., 50, then the entire page is replaced with a "FORBIDDEN" page. In an advantageous embodiment, only words in the same language are scored together and separate thresholds are predetermined for each category.

Optionally, the "FORBIDDEN" page provides additional information, such as the total score for the page, the score of the page in each category, the language of the objectionable words in the page, etc. This information may be viewed and the thresholds may be changed by one having the appropriate access, for instance, a parent or teacher.

The examples below illustrate how the scoring of a "Chain" is accomplished.

8

#### Sample Dictionary

hot:N:none:english:0:5  
fantasy:N:none:english:0:5  
pictures:N:none:english:0:5  
nude:N:pornography:english:5:5  
sexual:N:pornography:english:5:5  
harassment:N:none:english:0:-10

Sample "Chain"	Score	
sexual fantasy	15	(sexual has content and multiple words from the dictionary are in the "Chain" so bonus scores count. 5 + 5 from "sexual" and 5 from "fantasy")
sexual harassment	0	(sexual has content and multiple words are in the "Chain", so bonus scores count. 5 + 5 from "sexual" and -10 from "harassment")
pictures	0	("pictures" has no content)
hot pictures	0	(neither "hot" or "pictures" has content)
nude	5	("nude" has content, but only one word in "Chain", so only the base score applies)
hot nude pictures	20	(3 words in "Chain", 1 with content, all base and bonus scores apply)

One embodiment of the filtering method for the body of the web page, indicated at decision block 228 of FIG. 2, is shown in detail in FIG. 5. This method uses a "Proximity" variable to indicate whether a word found to be in the dictionary is within two words of the last word found to be in the dictionary, such that it should be part of the same "Chain." Also, a "Found" list is used to keep track of all the objectionable words found which should be replaced if the web page is to be displayed. Additionally, a "Tempchain" variable allows the method to determine if the next two sequential words are actually in the dictionary before they are added to the "Chain."

The method is begun at terminal block 510 and decision block 512 determines whether the "Filter" variable is set to "On." If not (indicating that the body of the web page should not be filtered), terminal block 514 returns a "No." If "Filter" is "On," then "Word" is set to the first word in the page at block 516. Decision block 518 determines whether the last word in the page has already been filtered and, if it has, block 520 indicates that all the words in the "Found" list are replaced with the replace word (e.g., "- - -"). Decision block 522 determines whether the "Score" for the page exceeds the predetermined "Targetscore" threshold and, if so, the page is replaced with the "FORBIDDEN" page in block 526 before a "Yes" is returned by terminal block 528, indicating that access was denied based on the web page body. If the score does not exceed the "Targetscore" threshold, a "No" is returned at terminal block 524.

If decision block 518 determines that the last word in the page has not been filtered, then decision block 530 is invoked to determine if "Word" is in the dictionary. If so, "Tempchain" is added to "Chain" along with "Word" and the "Proximity" variable is set to 1, in block 532. Block 534 then sets "Word" to the next word on the page. If "Word" is not in the dictionary, then decision block 536 checks to see if "Chain" is null (i.e., contains no words) and, if it is, block 534 sets "Word" to the next word on the page. If "Chain" is

Exhibit

2

Page

37



5,996,011

9

not null, decision block 538 determines if "Proximity" is greater than zero. If "Proximity" is greater than zero, indicating that the last word was in the dictionary, then 1 is subtracted from "Proximity" and "Word" is added to "Tempchain" in block 540. Block 534 then sets "Word" to the next word on the page. If "Proximity" is not greater than zero, then this indicates that the last two words were not in the dictionary and thus, the "Chain" is ended. The next portion of the method scores the "Chain."

In block 542, "Word" is set to the first word in the "Chain." If that is not the last word in the "Chain," as determined in decision block 544, then decision block 546 determines whether "Word" is replaceable. If yes, then "Word" is added to the "Found" list in block 550 and control is sent to block 548. If "Word" is not replaceable, then control is sent directly to block 548. Block 548 sets: "Base Score" to the previous "Base Score" plus the "Word's" base score; "Bonus Score" to the previous "Bonus Score" plus the "Word's" bonus score; "Content" to "Yes" if the "Word" has content; and "Word" to the next word in the "Chain". Block 544 then again determines if the "Word" is the last word in the "Chain".

If "Word" is the last word in the "Chain", then decision block 552 determines if "Content" is set to "yes." If so, then block 554 sets "Score" to the previous "Score" plus the "Bonus Score" and "Base Score," as last determined in block 548. Control is then sent to block 556. If "Content" is not set to "yes," then block 556 sets: "Chain" to null; "Tempchain" to null; "Proximity" to zero; "Bonus Score" to zero; "Base Score" to zero; and "Content" to "No." Block 534 then sets "Word" to the next word on the page.

One of ordinary skill in the art will recognize that the methods shown in the flowcharts of FIGS. 2 through 5 can readily be programmed into a computer using any of several computer programming languages. Advantageously, the method shown in FIG. 2 serves as the main loop of the program and the methods shown in FIGS. 3 through 5 serve as subroutines. The normal attendant initialization of variables, error checking, and the like, is programmed as required.

As noted above, while the system and method of the invention have been described with relation to filtering objectionable data from data received, the method can also be used to process data such that only Web pages containing desired data are passed to the user's computer.

Although the invention has been described in detail with respect to preferred embodiments thereof, it will be apparent to those skilled in the art that variations and modifications can be effected in these embodiments without departing from the spirit and scope of the invention.

I claim:

1. A computer readable memory containing a computer program for programming a general purpose computer to perform a method for filtering a block of text data containing words received over a network, wherein said method comprises the steps of:

- a) providing a listing of target words, each target word in said listing of target words having a respective score associated therewith, which score may be zero, wherein at least one of said target words has a negative score associated therewith;
- b) comparing each word in said block of text data to said listing to determine any word in said block of text data which matches one of said target words in said listing so as to determine matched text words and corresponding matched target words; and

10

c) determining a total score for said block of text data based on said score associated with each matched target word.

2. A computer readable memory as in claim 1, further comprising:

providing a respective replace-variable associated with each of said target words in said listing, each said replace-variable being set to either a true state or a false state; and

replacing each said word in said block of text data that matches one of said target words in said listing having its respective replace-variable set to a true state with a replacement-word to provide a new block of text data.

3. A computer readable memory as in claim 1, further comprising

replacing said block of text data with a substitute block of data if said total score for said block of text data exceeds a predetermined numerical threshold.

4. A computer readable memory as in claim 3, wherein said score associated with each of said target words in said listing comprises a base score and a bonus score, and wherein said negative score associated with said at least one of said target words comprises a negative bonus score and wherein said method further comprises determining said total score for said block of text data based on the respective bonus score associated with at least one of said matched target words.

5. A computer readable memory as in claim 4, wherein said method further comprises determining said total score for said block of text data based on the respective bonus score associated with each of said matched target words having a corresponding matched text word that is positioned within a predetermined proximity of another matched text word in said block of text data.

6. A computer readable memory as in claim 5, wherein said predetermined proximity comprises adjacent words.

7. A computer readable memory as in claim 5, wherein said predetermined proximity comprises a separation of at most one word.

8. A computer based method for filtering a block of text data containing words received over a network, said method comprising the steps of:

a) providing a listing of target words, each target word in said listing of target words having a respective score associated therewith, which score may be zero, wherein at least one of said target words has a negative score associated therewith;

b) comparing each word in said block of text data to said listing to determine any word in said block of text data which matches one of said target words in said listing so as to determine matched text words and corresponding matched target words; and

c) determining a total score for said block of text data based on said score associated with each matched target word.

9. A computer based method as in claim 8, further comprising:

providing a respective replace-variable associated with each of said target words in said listing, each said replace-variable being set to either a true state or a false state; and

replacing each said word in said block of text data that matches one of said target words in said listing having its respective replace-variable set to a true state with a replacement-word, to provide a new block of text data.

10. A computer based method as in claim 8, further comprising

Exhibit

2

Page

38

5,996,011

11

replacing said block of text data with a substitute block of data if said total score for said block of text data exceeds a predetermined numerical threshold.

11. A computer based method as in claim 10, wherein said score associated with each of said target words in said listing comprises a base score and a bonus score and wherein said negative score associated with said at least one of said target words comprises a negative bonus score, and wherein said total score for said block of text data is determined based on the respective bonus score associated with at least one of said matched target words.

12. A computer based method in claim 11, wherein said total score for said block of text data is determined based on the respective bonus score associated with each of said matched target words having a corresponding matched text word that is positioned within a predetermined proximity of another matched text word in said block of text data.

13. A computer based method as in claim 12, wherein said predetermined proximity comprises adjacent words.

14. A computer based method as in claim 12, wherein said predetermined proximity comprises a separation of at most one word.

15. A computer based method as in claim 8, further comprising:

d) providing a respective category-variable associated with each of said target words in said listing for expressing a category with which each of said target words in said listing is associated; and

e) providing an output comprising a record of the respective categories with which each matched target word is associated.

16. A computer based method as in claim 15, further comprising:

determining a total category score for said block of text data based on the respective score and category-variable associated with each matched text word; and providing said total category score for each category in said output.

17. A computer based method as in claim 16, further comprising:

replacing said block of text data with a substitute block of data if said total category score for a category exceeds a predetermined numerical threshold.

18. A computer based method for filtering a web page received over the World Wide Web and providing an output, said web page having a header portion, a body portion and an associated requested URL, said method comprising the steps of:

a) providing an allow-list of URLs associated with approved web pages;

b) providing a deny-list of URLs associated with disapproved web pages;

c) providing a listing of target words;

d) comparing said requested URL with said URLs in said allow-list, and if said requested URL matches any of said URLs in said allow-list, providing the web page as an output;

e) if said requested URL does not match any of said URLs in said allow-list, comparing said requested URL with said URLs in said deny-list, and, if said requested URL matches any of said URLs in said deny-list, providing an output indicating access to the web page is forbidden;

f) if said requested URL does not match any of said URLs in said deny-list, providing a computer based filter for

12

comparing each word in the header of the web page to said listing to determine any word in the header of the web page which matches one of said target words in said listing; and

g) providing an indication that access to the web page is forbidden or providing a modified version of the web page as an output based upon said determination in step f.

19. A computer based method as in claim 18, further comprising:

h) providing a computer based filter for comparing each word in the body of the web page to said listing to determine any word in the body of the web page which matches one of said target words in said listing; and

i) providing an indication that access to the web page is forbidden or providing a modified version of the web page as an output based upon said determination in step h.

20. A computer system for filtering a block of text data containing words received over a network, comprising:

a general purpose computer; and

a set of instructions for programming said general purpose computer to:

a) provide a listing of target words, each target word in said listing of target words having a respective score associated therewith, which score may be zero, wherein at least one of said target words has a negative score associated therewith;

b) compare each word in said block of text data to said listing to determine any word in said block of text data which matches one of said target words in said listing so as to determine matched text words and corresponding matched target words; and

c) determining a total score for said block of text data based on said score associated with each matched target word.

21. A computer system as in claim 20, wherein said general purpose computer is further programmed in step c to:

provide a respective replace-variable associated with each of said target words in said listing, each said replace-variable being set to either a true state or a false state; and

replace each said word in said block of text data that matches one of said target words in said listing having its respective replace-variable set to a true state with a replacement-word to provide a new block of text data.

22. A computer system as in claim 20, wherein said general purpose computer is further programmed to replace said block of text data with a substitute block of data if said total score for said block of text data exceeds a predetermined numerical threshold.

23. A computer system as in claim 22, wherein said score associated with each of said target words in said listing comprises a base score and a bonus score, and wherein said negative score associated with said at least one of said target words comprises a negative bonus score and wherein said total score for said block of text data is determined based on the respective bonus score associated with each of said matched target words having corresponding matched text word that is positioned within a predetermined proximity of another matched text word.

24. A computer readable memory containing a computer program for programming a general purpose computer to perform a method for filtering a block of text data containing

Exhibit

2

Page

39

5,996,011

13

words received over a network, wherein said method comprises the steps of:

- a) providing a listing of target words, each target word in said listing of target words having a respective score and bonus score associated therewith, which score and/or bonus score may be zero; 5
  - b) comparing each word in said block of text data to said listing to determine words in said block of text data which match one of said target words in said listing so as to determine matched text words and associated matched target words; 10
  - c) determining a total score for said block of text data based on:
    - (i) said score associated with each matched target word and 15
    - (ii) said bonus score associated with each matched target word only if said matched target word's associated matched text word is within a predetermined proximity to another matched text word in said block of text data. 20
25. A computer based method for filtering a block of text data containing words received over a network, said method comprising:
- a) providing a listing of target words, each target word in said listing of target words having a respective score and bonus score associated therewith, which score and/or bonus score may be zero; 25
  - d) comparing each word in said block of text data to said listing to determine words in said block of text data which match one of said target words in said listing so as to determine matched text words and associated matched target words; 30
  - e) determining a total score for said block of text data based on:

14

(i) said score associated with each matched target word and

(ii) said bonus score associated with each matched target word only if said matched target word's associated matched text word is within a predetermined proximity to another matched text word in said block of text data.

26. A computer based method for filtering a page of data received over the World Wide Web, wherein said page of data comprises a plurality of blocks of text data, said method comprising:

- a) providing a listing of target words, each target word in said listing of target words having a respective score associated therewith, which score may be zero;
- b) comparing each word in one of said blocks of text data to said listing to determine words in said first one of said blocks of text data which match one of said target words in said listing so as to determine matched text words and associated matched target words;
- c) determining a total score for said one of said blocks of text data based on said score associated with each matched target word;
- d) replacing said one of said blocks of text data with a substitute block of data if said total score for said one of said blocks of text data exceeds a predetermined numerical threshold;
- e) displaying said one of said blocks of text data or said substitute block of data;
- f) repeating steps a through e for each remaining block of text data in said page of data.

\* \* \* \* \*

# Exhibit 3



US007366919B1

(12) **United States Patent**  
Sobel et al.

(10) **Patent No.:** US 7,366,919 B1  
(45) **Date of Patent:** Apr. 29, 2008

(54) **USE OF GEO-LOCATION DATA FOR SPAM DETECTION**

(75) Inventors: William E. Sobel, Stevenson Ranch, CA (US); Bruce McCorkendale, Los Angeles, CA (US)

(73) Assignee: Symantec Corporation, Cupertino, CA (US)

(\*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 738 days.

6,088,803 A 7/2000 Tso et al.  
6,154,172 A \* 11/2000 Piccionelli et al. .... 342/357.1  
6,161,130 A 12/2000 Horvitz et al.  
6,167,434 A 12/2000 Pang  
6,249,807 B1 6/2001 Shaw et al.  
6,253,169 B1 6/2001 Apte et al.  
6,282,565 B1 8/2001 Shaw et al.  
6,289,416 B1 9/2001 Fukushima et al.  
6,298,351 B1 10/2001 Castelli et al.  
6,314,409 B2 11/2001 Schneck  
6,324,569 B1 11/2001 Ogilvie et al.  
6,347,310 B1 2/2002 Passera  
6,370,526 B1 4/2002 Agrawal et al.  
6,370,629 B1 \* 4/2002 Hastings et al. .... 711/163

(Continued)

(21) Appl. No.: 10/424,532

(22) Filed: Apr. 25, 2003

(51) Int. Cl.  
G06F 7/04 (2006.01)  
G06F 17/30 (2006.01)  
G06F 11/00 (2006.01)  
G06K 9/00 (2006.01)  
H04L 9/32 (2006.01)

(52) U.S. Cl. .... 713/201; 713/150; 713/189; 726/2; 726/22; 726/26

(58) Field of Classification Search .... 726/22, 726/24, 25; 713/150  
See application file for complete search history.

(56) **References Cited**

#### U.S. PATENT DOCUMENTS

5,675,710 A 10/1997 Lewis  
5,757,916 A \* 5/1998 MacDoran et al. .... 380/258  
5,778,304 A \* 7/1998 Grube et al. .... 455/456.4  
5,826,249 A 10/1998 Skeirik  
5,887,269 A \* 3/1999 Bruns et al. .... 701/208  
5,982,897 A \* 11/1999 Clark .... 380/258  
6,023,723 A 2/2000 McCormick et al.  
6,052,709 A 4/2000 Paul  
6,072,942 A 6/2000 Stockwell et al.

#### FOREIGN PATENT DOCUMENTS

WO WO 01/71499 A1 9/2001

#### OTHER PUBLICATIONS

outlook.spambully.com web pages [online]. Spam Bully [retrieved Jan. 16, 2003]. Copyright 2002. Retrieved from the Internet: <URL: http://outlook.spambully.com/about.php>.

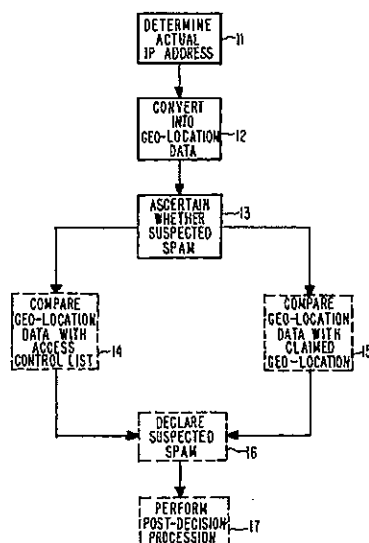
(Continued)

Primary Examiner—Nasser Moazzami  
Assistant Examiner—Chinwendu C Okoronkwo  
(74) Attorney, Agent, or Firm—Fenwick & West LLP

(57) **ABSTRACT**

Computer implemented methods, apparatus, and computer-readable media for detecting suspected spam in e-mail (24) originating from a sending computer (21). A method embodiment comprises the steps of determining (11) the actual IP address (23) of the sending computer (21); converting (12) the actual IP address (23) into geo-location data; and, using the geo-location data, ascertaining (13) whether the e-mail (24) contains suspected spam.

28 Claims, 2 Drawing Sheets





## US 7,366,919 B1

Page 2

## U.S. PATENT DOCUMENTS

6,397,200 B1 5/2002 Lynch, Jr. et al.  
 6,397,215 B1 5/2002 Kreulen et al.  
 6,421,709 B1 7/2002 McCormick et al.  
 6,424,960 B1 7/2002 Lee et al.  
 6,442,606 B1 8/2002 Subbaroyan et al.  
 6,453,419 B1\* 9/2002 Flint et al. .... 726/3  
 6,456,991 B1 9/2002 Srinivasa et al.  
 6,480,885 B1\* 11/2002 Olivier ..... 709/207  
 6,487,586 B2 11/2002 Ogilvie et al.  
 6,493,007 B1 12/2002 Pang  
 6,502,082 B1 12/2002 Toyama et al.  
 6,505,167 B1 1/2003 Horvitz et al.  
 6,546,416 B1 4/2003 Kirsch  
 6,640,301 B1 10/2003 Ng  
 6,643,685 B1 11/2003 Millard  
 6,650,890 B1 11/2003 Irlam et al.  
 6,654,787 B1 11/2003 Aronson et al.  
 6,687,740 B1 2/2004 Gough  
 6,691,156 B1 2/2004 Drummond et al.  
 6,697,942 B1 2/2004 L'Heureux et al.  
 6,701,347 B1 3/2004 Ogilvie  
 6,711,608 B1 3/2004 Ogilvie  
 6,732,157 B1 5/2004 Gordon et al.  
 6,757,713 B1 6/2004 Ogilvie et al.  
 6,757,830 B1 6/2004 Tarbotton et al.  
 6,859,791 B1\* 2/2005 Spagna et al. .... 705/51  
 6,901,346 B2 5/2005 Tracy et al.  
 6,928,553 B2 8/2005 Xiong et al.  
 7,155,484 B2\* 12/2006 Malik ..... 709/206  
 2002/0016831 A1\* 2/2002 Peled et al. .... 709/219  
 2002/0038308 A1 3/2002 Cappi  
 2002/0042687 A1 4/2002 Tracy et al.  
 2002/0083343 A1 6/2002 Crosbie  
 2002/0087641 A1 7/2002 Levosky  
 2002/0087649 A1 7/2002 Horvitz  
 2002/0087882 A1 7/2002 Schneier et al.  
 2002/0138525 A1 9/2002 Karadimitriou et al.  
 2002/0138581 A1 9/2002 MacIntosh et al.  
 2002/0147694 A1 10/2002 Dempsey et al.  
 2002/0147782 A1 10/2002 Dimitrova et al.  
 2002/0157020 A1 10/2002 Royer  
 2002/0165912 A1\* 11/2002 Wenocur et al. .... 709/203  
 2002/0199095 A1\* 12/2002 Bandini et al. .... 713/151  
 2002/0199186 A1 12/2002 Ali et al.  
 2002/0199194 A1 12/2002 Ali  
 2003/0033587 A1 2/2003 Ferguson et al.  
 2003/0037251 A1 2/2003 Frieder et al.  
 2003/0051026 A1 3/2003 Carter  
 2003/0105864 A1\* 6/2003 Mulligan et al. .... 709/225  
 2003/0149726 A1 8/2003 Spear  
 2003/0167311 A1 9/2003 Kirsch  
 2003/0191969 A1 10/2003 Katsikas  
 2003/0200334 A1 10/2003 Grynberg  
 2003/0220978 A1 11/2003 Rhodes  
 2003/0229672 A1 12/2003 Kohn  
 2003/0229801 A1\* 12/2003 Kouznetsov et al. .... 713/200  
 2003/0233415 A1 12/2003 Beyda  
 2004/0003283 A1 1/2004 Goodman et al.  
 2004/0024823 A1 2/2004 Del Monte  
 2004/0054887 A1 3/2004 Paulsen et al.  
 2004/0064734 A1 4/2004 Ehrlich  
 2004/0068534 A1 4/2004 Angermayr et al.  
 2004/0073617 A1 4/2004 Milliken et al.  
 2004/0093383 A1 5/2004 Huang et al.  
 2004/0093384 A1 5/2004 Shipp  
 2004/0111480 A1 6/2004 Yue  
 2004/0148358 A1 7/2004 Singh et al.  
 2004/0205173 A1 10/2004 Hall  
 2005/0097179 A1\* 5/2005 Orme ..... 709/207

2005/0144480 A1 6/2005 Kim et al.

## OTHER PUBLICATIONS

cauce.org web pages [online]. Coalition Against Unsolicited Commercial Email [retrieved Mar 17, 2003]. Retrieved from the Internet: <URL: <http://www.cauce.org/about/problem.shtml>>.  
 NBEC/NWOCA Anti-Spam Tools, [online] [retrieved Jul. 7, 2004] retrieved from <http://home.nwoca.org>, Jul. 7, 2004.  
 Kularski, C. "Compound Procedures for Spam Control," Highland School of Technology, Jan. 2004.  
 "Technical Responses to Spam," Nov. 2003, Taughannock Networks.  
 Cranor, Faith, L., LaMacchia, Brian A., "Spam!" Communications of the ACM, vol. 41, No. 8, pp. 74-83, Aug. 1998. U.S.A.  
 How it Works: Spam Recognition, <http://www.death2spam.net/docs/classifier.html>, retrieved Aug. 18, 2005, U.S.A.  
 Cavnar, William B. et al., "N-Gram-Based Text Categorization", Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, NV, USA, Apr. 13, 1994.  
 "N-Gram-Based Text Categorization", 2 pages, downloaded from <http://citescer.ist.psu.edu/68861.html>, Aug. 25, 2005 U.S.A.  
 TextCat Language Guesser, 2 pages, downloaded from <http://odur.lcl.rug.nl/~vannoord/Textcat/> on Aug. 25, 2005, U.S.A.  
 Spam Assassin, The Apache SpamAssassin Project, 2 pages, downloaded from <http://spamassassin.apache.org> on Aug. 25, 2005, U.S.A.  
 Basis Technology's Rosette Language Identifier, 2 pages, downloaded from <http://www.basistech.com/language-identification/> on Aug. 25, 2005, U.S.A.  
 Karp-Rabin algorithm, 3 pages, downloaded from <http://www-igm.univ-mlv.fr/~lecroq/string/node5.html> on Sep. 1, 2005, U.S.A.  
 Rabin-Karp string search algorithm, 5 pages, downloaded from [http://en.wikipedia.org/wiki/Rabin-Karp\\_string\\_search\\_algorithm](http://en.wikipedia.org/wiki/Rabin-Karp_string_search_algorithm) on Aug. 31, 2005 U.S.A.  
 The Rabin-Karp algorithm, String searching via Hashing, 5 pages, downloaded from <http://www.cccs.harvard.edu/~ellard/Q-97/HTML/root/node43> on Aug. 31, 2005 U.S.A.  
 Wikipedia.org web pages (online). Wikipedia (retrieved Mar. 17, 2003). Retrieved from the Internet: <URL: <http://www.wikipedia.org/wiki.php?title=Machine+learning+&printable=yes>>.  
 Outlook.spambully.com web pages [online] Spam Bully [retrieved Jan. 16, 2003] Copyright 2002, Retrieved from the Internet <URL: <http://outlook.spambully.com/about.php>>.  
 AirCERT web page, last updated Sep. 18, 2000 [online]. Cert.org [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://www.cert.org/kb/aircert/>, U.S.A.  
 Analysis Console for Intrusion Detection (ACID) web page [online]. Andrew.cmu.edu [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://www.andrew.cmu.edu/~rdanyliw/snort/>, U.S.A.  
 "Caltarian Security Technology Platform," Riptech web pages [online]. Symantec.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://enterprisesecurity.symantec.com/Content/displayPDF.cfm?SSSPDFID=35&EID=0>, U.S.A.  
 Change log for Analysis Console for Intrusion Detection (Acid), indicating release date of Sep. 8, 2000 [online]. Andrew.cmu.edu [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://www.andrew.cmu.edu/~rdanyliw/snort/CHANGELOG>, U.S.A.  
 Chung, C., Gertz, M. and Levitt, K., "DEMIDS: A Misuse Detection System for Database Systems," Department of Computer Science, University of California at Davis, Oct. 1, 1999, pp. 1-18.  
 CyberGuard Corporation, "CyberGuard and Webwasher: The Value Proposition," A CyberGuard Corporation White Paper, May 2004, 6 pages.  
 e=Security, Inc., Correlation Technology for Security Event Management, Oct. 7, 2002 [online]. eSecurityins.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: [http://www.esecurityinc.com/downloads/Correlation\\_WP.pdf](http://www.esecurityinc.com/downloads/Correlation_WP.pdf), Vienna, VA.  
 Lee, Sin Yeung; Low, Wai Lup and Wong, Pei Yuen, "Learning Fingerprints for a Database Intrusion Detection System," Computer Security Laboratory, DSO National Laboratories, Singapore, ESORICS Nov. 2002, LNCS 2502, pp. 264-279.

Exhibit

3

Page

42

US 7,366,919 B1

Page 3

Low, Wai Lup, et al., "DIDAFIT: Detecting Intrusions in Databases Through Fingerprinting Transactions," ICEIS 2002, Fourth International Conference on Enterprise Information Systems, vol. 1, Apr. 3-6, 2002, pp. 121-128, Ciudad Real, Spain.

Marketing, "Digital Certificates—Best Practices—A Microdasys Whitepaper," bestpractice.doc, Revision 1/1 (Jul. 31, 2003), 6 pages, Czech Republic.

Microdasys, "SCIP Secured Content Inspection: Protecting the Enterprise from CryptoHacks," 2003 by Microdasys Inc., 2 pages, Czech Republic.

MyNetWatchman.com web pages indicating Sep. 2000 beta release [online]. MyNetWatchman.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://www.mynetwatchman.com/mynetwatchman/relnotes.htm>, Alpharetta, GA.

Network Computing Solutions—"Microdasys SCIP" [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: <http://www.ncs.cz/index.php?language=en&menuitem=4&subitem=13>, 2 pages, Czech Republic.

Network Computing Solutions—NSC Homepage—News [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: <http://www.nsc.cz/index.php?language=en&menuitem=0&subitem=4&subitem=13>, 3 pages, Czech Republic.

Schneier, Bruce, Managed Security Monitoring: Network Security for the 21<sup>st</sup> Century, 2001 [online]. Counterpane.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://www.counterpane.com/msn.pdf>, U.S.A.

SCIP Product, Microdasys—"The need to control, inspect and manage encrypted webtraffic." [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: <http://www.microdasys.com/scipproduct+M54a708de802.html>. Author unknown, 2 pages, Czech Republic.

Slashdot.org web pages describing Dshield, dated Nov. 27, 2000 [online]. Slashdot.org [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://slashdot.org/article.pl?sid=00/11/27/1957238&mode=thread>, U.S.A.

"SSL Stripper Installation Guide," [online]. Retrieved in Mar. 2005 from the Internet: URL: <http://www.sslstripper.com>, 2 pages, U.S.A. SSL Stripper Home Page, "Security Solutions: SSL Stripper," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: <http://www.vroyer.org/sslstripper/index.html>, 2 pages, Oct. 15, 2004, U.S.A.

SSL Stripper Sample Screenshots, "Security Solutions: Sample Screenshots," [online]. Retrieved on Mar. 18, 2005. Retrieved from

the Internet: URL: <http://www.vroyer.org/sslstripper/screenshots.html>, 3 pages, Oct. 15, 2004, U.S.A.

Web page, announcing Nov. 11, 2000 release of Dshield [online]. Deja.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: URL: <http://groups.google.com/groups?selm=8vm48v%245pd%241%40nnrp1.deja.com&oe=UTF-8&output=gplain>, U.S.A.

Webwasher AG/Full feature set, "Full feature set," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/ssl\\_scanner/full\\_feature\\_set.html?l...](http://www.webwasher.com/enterprise/products/webwasher_products/ssl_scanner/full_feature_set.html?l...), 2 pages.

Webwasher AG/Webwasher 1000 CSM Appliance, "Webwasher 1000 CSM Appliance," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/scm\\_appliance/index...](http://www.webwasher.com/enterprise/products/webwasher_products/scm_appliance/index...) 2 pages.

Webwasher AG/Webwasher URL Filter, "Webwasher URL Filter," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/webwasher\\_url\\_filter..](http://www.webwasher.com/enterprise/products/webwasher_products/webwasher_url_filter..) 1 page.

Webwasher AG/Webwasher Anti Virus, "Webwasher Anti Virus," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/anti\\_virus/index.html...](http://www.webwasher.com/enterprise/products/webwasher_products/anti_virus/index.html...), 2 pages.

Webwasher AG/Webwasher Anti Spam, "Webwasher Anti Spam," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/anti\\_spam/index.htm...](http://www.webwasher.com/enterprise/products/webwasher_products/anti_spam/index.htm...), 1 page.

Webwasher AG/Webwasher Content Protection, "Webwasher Content Protection," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/content\\_protection/index.html](http://www.webwasher.com/enterprise/products/webwasher_products/content_protection/index.html), 2 pages.

Webwasher AG/Webwasher SSL Scanner, "Webwasher SSL Scanner," [online]. Retrieved on Mar. 18, 2005. Retrieved from the Internet: URL: [http://www.webwasher.com/enterprise/products/webwasher\\_products/ssl\\_scanner/index.html](http://www.webwasher.com/enterprise/products/webwasher_products/ssl_scanner/index.html), 2 pages.

2000 Review of eSecurity product on Network Security web page [online]. SCMagazine.com [retrieved Apr. 18, 2003]. Retrieved from the Internet: <URL: [http://www.scmagazine.com/scmagazine/2000\\_12/testc/network.htm#Open](http://www.scmagazine.com/scmagazine/2000_12/testc/network.htm#Open)>.

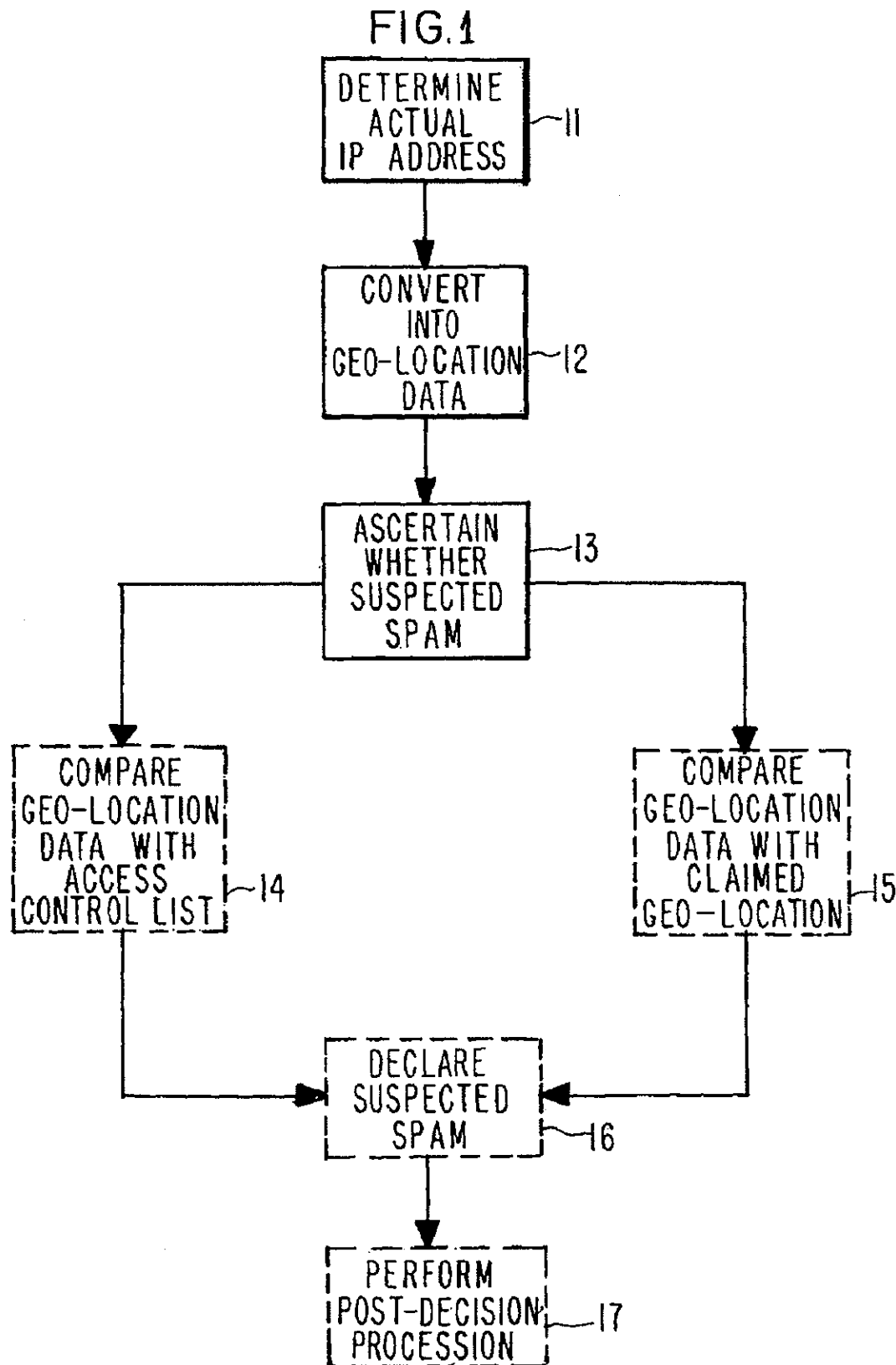
\* cited by examiner

U.S. Patent

Apr. 29, 2008

Sheet 1 of 2

US 7,366,919 B1



Exhibit

3

Page

44

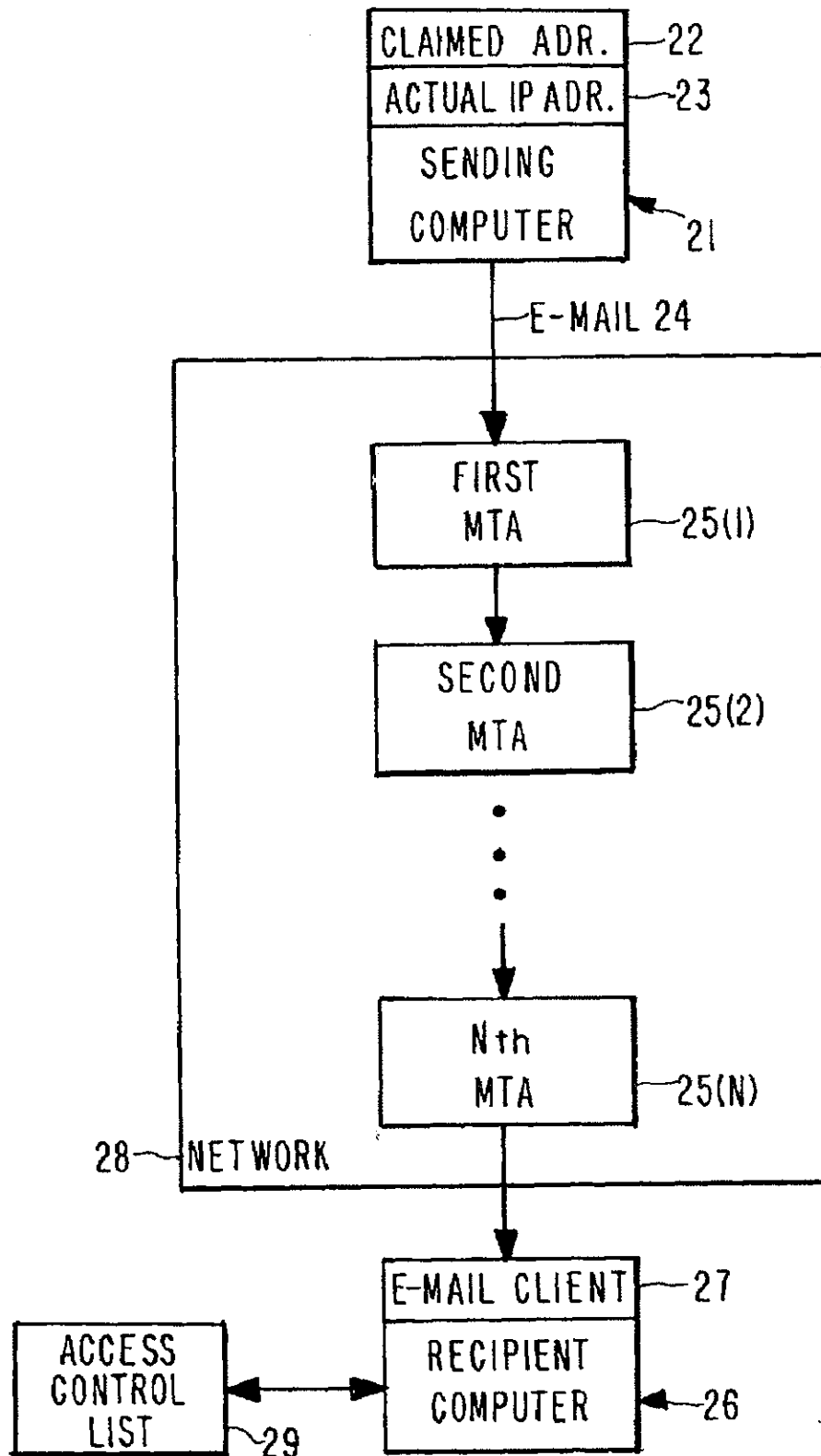
U.S. Patent

Apr. 29, 2008

Sheet 2 of 2

US 7,366,919 B1

FIG.2



US 7,366,919 B1

# 1

## USE OF GEO-LOCATION DATA FOR SPAM DETECTION

### TECHNICAL FIELD

This invention pertains to the field of reducing the amount of spam to which a computer is subjected.

### BACKGROUND ART

As used throughout this specification including claims, "spam" is any e-mail that is unwanted by the recipient. As spam has regrettably become more widely prevalent, techniques to combat spam are beginning to emerge. One such technique is to allow e-mail recipients to specify a list of addresses, domains, and/or top-level domains to be always blocked or automatically allowed. The "block" list is often referred to as a "blacklist", while the "allow" list is often referred to as a "whitelist". The inspiration behind blacklists and whitelists is the observation that most computer users exchange e-mail with a relatively small and fixed set of addresses. These addresses are on a smaller list of domains, and these domains are on an even smaller list of top-level domains. A significant amount of spam comes from addresses, domains, and top-level domains that a user rarely, if ever, legitimately interacts with. Blocking entire domains or top-level domains (as well as addresses) thus becomes a relatively easy way to block a significant amount of spam. There is a need to improve the use of whitelists and blacklists in fighting spam. Much spam also comes from addresses claiming to be on common domains such as yahoo.com, msn.com, aol.com, and hotmail.com. Blocking these domains would, for most computer users, block too much legitimate e-mail. Furthermore, many spammers falsely indicate that they are sending e-mails from such common domains when, in reality, they are not. In other words, the spammer is spoofing his or her address. There is a need to develop techniques to counter such spoofing.

### DISCLOSURE OF INVENTION

Computer implemented methods, apparatus, and computer-readable media for detecting suspected spam in e-mail (24) originating from a sending computer (21). A method embodiment comprises the steps of determining (11) the actual IP address (23) of the sending computer (21); converting (12) the actual IP address (23) into geo-location data; and, using the geo-location data, ascertaining (13) whether the e-mail (24) contains suspected spam.

### BRIEF DESCRIPTION OF THE DRAWINGS

These and other more detailed and specific objects and features of the present invention are more fully disclosed in the following specification, reference being had to the accompanying drawings, in which:

FIG. 1 is a flow diagram illustrating a method embodiment of the present invention.

FIG. 2 is a block diagram illustrating apparatus used in the present invention.

### DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

As used throughout this specification including claims, the following terms have the following meanings:

2

"OSI" is the Open System Interconnect model developed by the International Standardization Organization (ISO) in 1984. This model, described at <http://www.4d.com.docs/cmu/cmu79892.htm>, describes how data is transferred from an application on one computer to an application on another computer. The OSI model comprises seven different layers.

"TCP" is the Transmission Control Protocol. It operates at the transport layer (layer 4) of the OSI model to establish data transmission reliability.

"IP" is the Internet Protocol.

"IP address" is the unique address of a computer that is coupled to the Internet. The IP address has the form #.#.#.#, where each # is a number from zero to 255. For example, an IP address might be 66.120.211.171.

"IANA" is the Internet Assigned Number Authority, an agency given authority by the U.S. government to assign domain names.

"DNS" is the Domain Name System.

"DNS address" is an address of a computer, complying with the DNS and expressed in a form that is relatively user friendly compared with the IP address. An example of a DNS address is fenwick.com. In this address, "fenwick" is a domain and ".com" is the top-level domain. The top-level domain may also be a country.

"SMTP" is the Simple Mail Transfer Protocol, a protocol which currently governs all e-mail sent over the Internet.

"MTA" is Mail Transfer Agent, a computer such as a large server computer that forwards e-mail from a sending computer to a recipient computer.

"Access Control List" is a whitelist or a blacklist, as those terms have been defined above.

"Coupled" encompasses any direct or indirect coupling or connection.

"Network" is any wired or wireless network, such as the Internet, a Local Area Network (LAN), or a Wide Area Network (WAN).

In the present invention, and with reference to FIG. 2, a sending computer 21 sends an e-mail 24 to a recipient computer 26 over a network 28. An access control list 29 may be associated with recipient computer 26. If the e-mail 24 is unwanted by the recipient computer 26, the e-mail 24 is said to be spam or to contain spam. Within network 28 may be one or more MTA's 25. FIG. 2 illustrates N MTA's. N can be any positive integer.

Sending computer 21 has a unique actual IP address 23 as well as a claimed address 22, which may be expressed as an IP address or as a DNS address. If the user of sending computer 21 wishes to include spam within e-mail 24, said user is referred to as a spammer. A spammer may spoof an innocuous claimed address 22 that is not an actual address of sending computer 21. This may lull the user of recipient computer 26 into thinking that the e-mail 24 does not contain spam, because this claimed address 22 may be presented to the user of recipient computer 26 via e-mail client software 27 associated with recipient computer 26.

FIG. 1 illustrates the method of the present invention as having three generic steps: 11, 12, and 13. In step 11, the actual IP address 23 of sending computer 21 is determined. At step 12, the actual IP address 23 is converted into geo-location data, which is data giving the actual geographical location of sending computer 21. The geo-location data may be any type of geographical information such as city, county, state, country, or presence within a pre-selected radius of a geographical point. The conversion of the actual IP address 23 into geo-location data at step 12 may be performed by a software program such as Geobytes. Such software programs were designed for marketing purposes,



US 7,366,919 B1

3

e.g., letting the owner of a Website know where most of his hits are coming from so he can tailor his marketing approach accordingly. Geobytes includes with the geo-location data a confidence number from 1 to 100, with 1 representing virtually no confidence in the geo-location data offered by the program, and 100 representing complete confidence. In one embodiment, a pre-selected confidence threshold level between 1 and 100 is selected by the user of the present invention, and only geo-location data exceeding the pre-selected threshold is used; all other geo-location data is ignored as being unreliable.

At step 13, it is ascertained whether e-mail 24 contains suspected spam. Steps 11, 12, and 13 may be performed by one or more modules associated with recipient computer 26 and/or with one of the MTA's 25. Said modules can be implemented in hardware, firmware, and/or software. A module may take the form of a standalone software program, a plug-in module, or a proxy situated in front of (with respect to sending computer 21) computer 26 or MTA 25. When the modules of the present invention are embodied in software, they may reside on any computer-readable medium such as a hard disk, floppy disk, CD, DVD, etc.

It may be that the module performing step 12 may not be able to successfully convert the actual IP address 23 into geo-location data. In such a case, the converting module may indicate "unknown" rather than the name of a geographical entity or location. When this happens, in one embodiment, particularly useful when the claimed domain of sending computer 21 is not common, ascertaining step 13 is programmed to automatically declare that suspected spam is not present in e-mail 24. Examples of common domains are yahoo.com, msn.com, aol.com, and hotmail.com. In an alternative embodiment, particularly useful when the claimed domain within address 22 of sending computer 21 is common, the ascertaining step 13 is automatically programmed to declare that suspected spam is present in e-mail 24. The theory behind this alternative embodiment is that if the domain really is in fact is a common actual domain, and not a spoofed domain, the geo-location data should not be returned as "unknown".

Alternative embodiments for implementing step 13 are illustrated in FIG. 1 by means of steps 14 and 15. In a first embodiment of the present invention, at step 14 the actual geo-location data of sending computer 21 is compared with an access control list, which might be a whitelist or a blacklist. To illustrate the use of a whitelist 29, let us assume that whitelist 29 contains the name of the city Cincinnati. Then, when the geo-location data shows that the actual location of the sending computer 21 is Cincinnati, recipient computer 26 automatically accepts the e-mail 24, without any further analysis as to whether e-mail 24 contains spam or not. To illustrate the use of a blacklist 29, let us assume that blacklist 29 contains an entry for the country of China. Then, whenever the geo-location data indicates that the actual location of sending computer 21 is China, computer 26 or MTA 25 automatically treats e-mail 24 as containing suspected spam (step 16).

Once a declaration has been made that e-mail 24 contains suspected spam at step 16, post-decision processing can be performed at step 17. For example, e-mail 24 can be refused by recipient computer 26; the suspected spam can be deleted from e-mail 24; e-mail 24 can be subjected to further processing by a spam filter; e-mail 24 can be tagged as containing suspected spam; e-mail 24 can be moved to a

4

special folder where the user of recipient computer 26 can later check it in case there was a false positive; a composite spam score maintained by recipient computer 26 can be augmented; and/or e-mail 24 can be modified. When optional step 17 is not used, the declaration at step 16 is that "spam is present" rather than "suspected spam is present".

The determination at step 11 of the actual IP address 23 can be made at a time when sending computer 21 connects with an MTA 25. At that time, pursuant to the SMTP protocol, sending computer 21 initiates the sending of an e-mail 24 by issuing a HELO or EHLO command. Following the word HELO or EHLO, sending computer 21 inserts a claimed address 22, which can be spoofed. However, the actual IP address 23 can be determined by a conventional module associated with computer 26 or MTA 25, by examining what is happening at the transport layer using a knowledge of TCP. Alternatively, or in addition to said method for determining the actual IP address 23, the actual IP address 23 can be determined by a conventional module associated with computer 26 or MTA 25, by examining a return path header associated with e-mail 24, again, by using a knowledge of TCP to observe what is transpiring at the transport layer. An example of a set of return path headers is giving in the following:

```

1 Return-path: <ronronron@eudoramail.com>
2 Received: from 207.118.30.214 (unverified
  [216.96.57.133]) by
3 mail01.corp.xyz.com
4 (SMTPRS) with SMTP id
5 5c-B0002012226@mail01.corp.xyz.com> for
  <bill@xyz.com>;
6 Mon, 24 Mar. 2003 03:37:15-0800
7 Received: from 285 bpq3wz2mfu [25.149.92.80] by
8 206.117.30.214 with ESMTP id
9 JRAZX; Mon, 24 Mar 03 06:33:15 +0400
10 Received: from 6zd6.gs5gs4 [78.32.232.240] by
11 25.149.92.80 with ESMTP id
12 HLXCVRMFX; Mon, 24 Mar 03 06:17:15 +0400
13 Message-ID: <ple3-S3z2ig27xzw@i7t.mvuv>
14 From: "Carmine
  Opera"<ronronron@eudoramail.com>
15 To: bill@xyz.com
16 Date: Mon, 24 Mar 03 06:17:15 GMT
17 X-Priority: 3
18 X-MSMail-Priority: Normal
19 X-Mailer: MIME-tools 5.503 (Entity 5.501)
20 MIME-Version: 1.0
21 Content-Type: multipart/alternative;
22 boundary="1_F_14_A.3"
23 X-SYMC-SmtMailFrom:
  ronronron@eudoramail.com
24 X_SYMCFilter-IP: 216.96.57.133
25 X-SYMCFilter-Path-1: 216.96.57.133 [last relay]
26 (US;US;United States;KS;Kansas;Effingham;96)
27 X-SYMCFilter-Path-2: 25.149.92.80 [prior relay](un-
  known)
28 X-SYMCFilter-Path-3: 78.32.232.240 [prior relay]
  (unknown)
29 X-SYMCFilter-Reason: bill@xyz.com bill@xyz.com
  1 Found
30 10.0.0.14 on BackupTrap list
31 Subject: [FILTERED] Admin.the nature of the search
32 engines uaieyfosbjlld

```

The return path header that should be examined is that associated with the MTA 25 that is closest to the sending

Exhibit

3

Page

47

US 7,366,919 B1

5

computer 21, as long as said closest MTA 25 and each MTA 25 situated between said closest MTA 25 and said recipient computer 26 is trusted by recipient computer 26. "Trust" can be defined in a number of ways. For example, recipient computer 26 can be said to trust an MTA 25 when the MTA 25 is co-located with recipient computer 26 in a common enterprise, such as a corporation, university, or government agency. Alternatively, recipient computer 26 can be said to trust an MTA 25 when the MTA 25 appears on a list of trusted computers kept by recipient computer 26. Alternatively, recipient computer 26 can be said to trust an MTA 25 when the recipient computer 26 has never been spoofed by the MTA 25.

In the above exemplary set of return path headers, the DNS address ronronron@eudoramail.com appearing on line 1 is a spoofed address 22 claimed by sending computer 21. The IP address 207.118.30.214 appearing on line 2 is the IP address corresponding to this spoofed DNS address 22. Such an IP address can be determined, for example, by consulting the WHOIS database. The IP address 216.96.57.133 appearing on line 2 is the actual IP address 23 of computer 21 as determined by an inventive module as described herein. In this example, there are three MTA's 25 situated between sending computer 21 and recipient computer 26. The return path header for the last MTA 25(3) is given on line 25. The return path header for the second MTA 25(2) is given on line 27. The return path header for the first MTA 25(1) is given on line 28. As can be seen from line 25, the IP address of the last MTA 25(3) is 216.96.57.133. This was determined by step 11 using TCP, as described above. Line 26 shows the results of applying step 12 to this IP address 23. The converting means of step 12 has determined that this MTA 25(3) is located in Effingham, Kansas, United States of America, with a confidence level of 96. As can be seen from the word "unknown" appearing on lines 27 and 28, the converting means of step 12 was not able to produce geo-location data for MTA 25(2) or MTA 25(1), respectively.

In an alternative embodiment of the present invention, at step 15 the geo-location data produced at step 12 is compared with a claimed geo-location claimed by sending computer 21. This claimed geo-location may be derived from domain information inserted by sending computer 21 in an outgoing e-mail 24. For example, in the SMTP protocol, the sending of an e-mail 24 is initiated by sending computer 21 indicating its desire to connect to an MTA 25 by generating a HELO or EHLO command having an address as an argument. At this time, a spammer in control of sending computer 21 may insert a bogus address following the word HELO or EHLO. There are other places where the spammer can falsify the address 22, e.g., as part of a MAILFROM command. This bogus address 22 may be presented to the user of recipient computer 26 via e-mail client software 27 associated with recipient computer 26.

In this embodiment of the present invention, a DNS address 22 claimed by the sending computer 21 is resolved into a location or a set of locations corresponding to the claimed address 22. If the actual geo-location obtained in step 12 matches one of these resolved locations, ascertaining step 13 determines that e-mail 24 does not contain suspected spam. If, on the other hand, the actual geo-location obtained from step 12 does not match one of these resolved locations, a declaration is made at step 16 that e-mail 24 contains

6

suspected spam. For example, if it is known that the domain from the claimed address 22 has an MTA 25 in five countries, but the actual geo-location of computer 21 is in a sixth country, suspected spam is declared at step 16. Similarly, if it is known that said domain does not have an MTA 25 in China, and the geo-location report from step 12 indicates that the actual location of computer 21 is in China, suspected spam is likewise declared at step 16.

The resolved locations may be obtained by examining the domain name within the claimed DNS address 22. One or more databases can be created over time giving resolved locations for each of a set of domains. These databases can be created by any one or more of a number of various techniques. For example, the owner of the domain (such as aol.com) may publish locations where its MTA's 25 are located. Alternatively, database entries may be compiled by machine learning techniques such as neural networks, Bayesian classifiers, or support vector machines. For example, the training process may identify those domains responsible for the most e-mail traffic to an MTA 25 and the associated geo-locations for those domains. The training process can be performed by updating a database containing the domain name, its associated IP address, geo-location data for this IP address, and a count of each incoming e-mail for that domain. One can have a pre-selected confidence threshold level. For example, let us assume that this threshold is set at 5000. If the training process indicates that there are at least 5000 matches of the domain "aol.com" with the location "Boise, Id.", one can safely assume that there is in fact an aol MTA 25 located in Boise, Id. In this example, spammers who are spamming the domain name aol.com will likely have their locations resolved to a large plurality of locations, each of which will experience far fewer than 5000 counts.

A database trained by machine learning techniques can be offered by a software publisher to its customers 26 so that each customer computer 26 does not have to perform this training process itself.

In an alternative embodiment, a database of resolved locations can be compiled by simply noticing locations from which a large amount of e-mail has emanated. In all of these embodiments, a database may be compiled only with respect to e-mail allegedly emanating from common domains, on the theory that e-mail allegedly emanating from uncommon domains will not be prolific enough to make for a reliable database. Examples of common domains include yahoo.com, msn.com, aol.com, and hotmail.com. Another way of defining "common" is to require that, during a machine learning training phase such as described above, at least a certain preselected threshold percentage worth of total e-mail traffic observed during the training phase must be exceeded in order for the domain to be classified as "common".

The above description is included to illustrate the operation of the preferred embodiments and is not meant to limit the scope of the invention. The scope of the invention is to be limited only by the following claims. From the above discussion, many variations will be apparent to one skilled in the art that would yet be encompassed by the spirit and scope of the present invention. For example, the principles of the present invention can be applied to any protocol in any electronic messaging system where the actual location of the

Exhibit

3

Page

48

US 7,366,919 B1

7

sender of the electronic message can be determined, and thus checked against where the sender claims to be, to see whether the sender is lying.

What is claimed is:

1. A computer implemented method for detecting suspected spam in e-mail originating from a sending computer, said method comprising the steps of:

determining an actual IP address of the sending computer; converting the actual IP address into actual geo-location data; and using the actual geo-location data, ascertaining whether the e-mail contains suspected spam.

2. The method of claim 1 further comprising, when the ascertaining step finds that the e-mail contains suspected spam, at least one of the following steps:

refusing the e-mail;  
deleting the suspected spam from the e-mail;  
processing the e-mail by a spam filter;  
tagging the e-mail as containing suspected spam;  
moving the e-mail to a special folder;  
augmenting a composite spam score;  
modifying the e-mail.

3. The method of claim 1 wherein the actual IP address is determined when the sending computer connects to a mail transfer agent.

4. The method of claim 3 wherein the actual IP address is gleaned from a transport layer using TCP.

5. The method of claim 1 wherein the actual IP address is determined from a return path header associated with the e-mail.

6. The method of claim 5 wherein:

the e-mail is received by a recipient computer;  
there are a plurality of mail transfer agents situated between the sending computer and the recipient computer; and

the determining step comprises deriving the IP address from the return path header associated with the mail transfer agent that is closest to the sending computer, as long as said closest mail transfer agent, and each mail transfer agent situated between said closest mail transfer agent and said recipient computer, is trusted by the recipient computer.

7. The method of claim 6 wherein a mail transfer agent is trusted by the recipient computer when one of the following conditions is satisfied:

the mail transfer agent is co-located with the recipient computer in a common enterprise;  
the mail transfer agent appears on a list of trusted computers that is kept by the recipient computer;  
the recipient computer has never been spoofed by the mail transfer agent.

8. The method of claim 1 wherein:

the converting step is not able to provide geo-location data;  
the sending computer's claimed domain is not common; and

the ascertaining step does not deem that suspected spam is present in the e-mail.

9. The method of claim 1 wherein:

the converting step is not able to provide actual geo-location data;  
the sending computer's claimed domain is common; and the ascertaining step deems that suspected spam is present in the e-mail.

8

10. The method of claim 1 wherein the actual geo-location data has a confidence number associated therewith; and the ascertaining step is performed only when the confidence number exceeds a pre-selected threshold.

11. The method of claim 1 wherein the ascertaining step comprises comparing the actual geo-location data with an access control list.

12. The method of claim 11 wherein the access control list is a whitelist.

13. The method of claim 11 wherein the access control list is a blacklist.

14. The method of claim 11 wherein the actual geo-location data is data pertaining to an entity from the group of entities comprising city, county, state, country, and presence within a preselected radius of a geographical point.

15. The method of claim 1 wherein the ascertaining step comprises comparing the actual geo-location data with a claimed geo-location claimed by the sending computer.

16. The method of claim 15 wherein the claimed geo-location is derived from domain information inserted by the sending computer as part of a HELO or EHLO command pursuant to a SMTP protocol.

17. The method of claim 15 wherein:  
the e-mail is received by a recipient computer; and  
the claimed geo-location is derived from a DNS address appearing in e-mail client software associated with the recipient computer.

18. The method of claim 15 wherein the claimed geo-location:

is derived from a DNS address claimed by the sending computer; and  
is a location where a domain within the DNS address has a mail transfer agent.

19. The method of claim 18 wherein locations where the domain has a mail transfer agent are listed in a database from the group of databases comprising:

a database listing locations disclosed by an owner of the domain;  
a database compiled by machine learning techniques;  
a database compiled by noticing locations from which a large amount of e-mail has emanated.

20. The method of claim 19 wherein the database is compiled only with respect to e-mail allegedly emanating from common domains.

21. The method of claim 20 wherein a domain is deemed to be common during a training phase by virtue of being responsible for at least a certain preselected threshold percentage worth of total e-mail traffic observed during the training phase.

22. The method of claim 1 wherein at least one of the determining, converting, and ascertaining steps is performed by a client computer receiving the e-mail.

23. The method of claim 1 wherein at least one of the determining, converting, and ascertaining steps is performed by a mail transfer agent computer that processes the e-mail.

24. A computer program product for detecting suspected spam in e-mail originating from a sending computer, comprising:

a computer-readable medium; and  
computer program code, encoded on the computer-readable medium, for:  
determining an actual IP address of the sending computer;  
converting the actual IP address into actual geo-location data; and  
using the actual geo-location data, ascertaining whether the e-mail contains suspected spam.

Exhibit

3

Page

49



US 7,366,919 B1

9

25. The computer program product of claim 24 wherein the ascertaining step comprises comparing the actual geo-location data with an access control list.

26. The computer program product of claim 24 wherein the ascertaining step comprises comparing the actual geo-location data with a claimed geo-location claimed by the sending computer.

27. An apparatus for detecting suspected spam in e-mail originating from a sending computer, said apparatus comprising:

10

means for determining an actual IP address of the sending computer;

coupled to the determining means, means for converting the actual IP address into actual geo-location data; and coupled to the converting means, means for ascertaining, using the actual geo-location data, whether the e-mail contains suspected spam.

28. The method of claim 1 wherein the geo-location data comprises data identifying an actual geographical location of the sending computer.

\* \* \* \* \*

**UNITED STATES DISTRICT COURT  
CENTRAL DISTRICT OF CALIFORNIA**

**NOTICE OF ASSIGNMENT TO UNITED STATES MAGISTRATE JUDGE FOR DISCOVERY**

This case has been assigned to District Judge Josephine Tucker and the assigned discovery Magistrate Judge is Robert N. Block.

The case number on all documents filed with the Court should read as follows:

**SACV10- 1513 JST (RNBx)**

Pursuant to General Order 05-07 of the United States District Court for the Central District of California, the Magistrate Judge has been designated to hear discovery related motions.

All discovery related motions should be noticed on the calendar of the Magistrate Judge

===== :  
**NOTICE TO COUNSEL**

*A copy of this notice must be served with the summons and complaint on all defendants (if a removal action is filed, a copy of this notice must be served on all plaintiffs).*

Subsequent documents must be filed at the following location:

☐ **Western Division**  
312 N. Spring St., Rm. G-8  
Los Angeles, CA 90012

☒ **Southern Division**  
411 West Fourth St., Rm. 1-053  
Santa Ana, CA 92701-4516

☐ **Eastern Division**  
3470 Twelfth St., Rm. 134  
Riverside, CA 92501

Failure to file at the proper location will result in your documents being returned to you.

ORIGINAL

AO 440 (Rev. 12/09) Summons in a Civil Action

## UNITED STATES DISTRICT COURT

for the  
Central District of California

SYMANTEC CORPORATION

*Plaintiff*

v.

M86 SECURITY, INC.

*Defendant*

Civil Action No. SACV10-1513-JST (RNB)

## SUMMONS IN A CIVIL ACTION

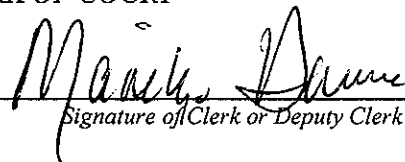
To: *(Defendant's name and address)*M86 Security, Inc.  
828 West Taft Avenue  
Orange, California 92865

A lawsuit has been filed against you.

Within 21 days after service of this summons on you (not counting the day you received it) — or 60 days if you are the United States or a United States agency, or an officer or employee of the United States described in Fed. R. Civ. P. 12 (a)(2) or (3) — you must serve on the plaintiff an answer to the attached complaint or a motion under Rule 12 of the Federal Rules of Civil Procedure. The answer or motion must be served on the plaintiff or plaintiff's attorney, whose name and address are:

If you fail to respond, judgment by default will be entered against you for the relief demanded in the complaint. You also must file your answer or motion with the court.

CLERK OF COURT

Date: 7 OCT 2010

  
Signature of Clerk or Deputy Clerk

AO 440 (Rev. 12/09) Summons in a Civil Action (Page 2)

Civil Action No. \_\_\_\_\_

**PROOF OF SERVICE***(This section should not be filed with the court unless required by Fed. R. Civ. P. 4(l))*

This summons for *(name of individual and title, if any)* M86 SECURITY, INC.,  
 was received by me on *(date)* \_\_\_\_\_.

☐ I personally served the summons on the individual at *(place)* \_\_\_\_\_  
 \_\_\_\_\_ on *(date)* \_\_\_\_\_; or

☐ I left the summons at the individual's residence or usual place of abode with *(name)* \_\_\_\_\_  
 \_\_\_\_\_, a person of suitable age and discretion who resides there,  
 on *(date)* \_\_\_\_\_, and mailed a copy to the individual's last known address; or

☐ I served the summons on *(name of individual)* \_\_\_\_\_, who is  
 designated by law to accept service of process on behalf of *(name of organization)* M86 SECURITY, INC.,  
 \_\_\_\_\_ on *(date)* \_\_\_\_\_; or

☐ I returned the summons unexecuted because \_\_\_\_\_; or

☐ Other *(specify)*: \_\_\_\_\_.

My fees are \$ \_\_\_\_\_ for travel and \$ \_\_\_\_\_ for services, for a total of \$ \_\_\_\_\_.

I declare under penalty of perjury that this information is true.

Date: \_\_\_\_\_

\_\_\_\_\_  
*Server's signature*

\_\_\_\_\_  
*Printed name and title*

\_\_\_\_\_  
*Server's address*

Additional information regarding attempted service, etc:

**UNITED STATES DISTRICT COURT, CENTRAL DISTRICT OF CALIFORNIA  
CIVIL COVER SHEET**

<b>I (a) PLAINTIFFS</b> (Check box if you are representing yourself <input type="checkbox"/> ) Symantec Corporation	<b>DEFENDANTS</b> M86 Security, Inc.
<b>(b) Attorneys</b> (Firm Name, Address and Telephone Number. If you are representing yourself, provide same.)  Quinn Emanuel Urquhart & Sullivan, LLP 50 California Street, 22nd Floor San Francisco, California 94111-4788	<b>Attorneys</b> (If Known)

<b>II. BASIS OF JURISDICTION</b> (Place an X in one box only.)  <input type="checkbox"/> 1 U.S. Government Plaintiff <input checked="" type="checkbox"/> 3 Federal Question (U.S. Government Not a Party)  <input type="checkbox"/> 2 U.S. Government Defendant <input type="checkbox"/> 4 Diversity (Indicate Citizenship of Parties in Item III)	<b>III. CITIZENSHIP OF PRINCIPAL PARTIES - For Diversity Cases Only</b> (Place an X in one box for plaintiff and one for defendant.) <table style="width:100%; border: none;"> <tr> <td style="width:33%; border: none;">Citizen of This State</td> <td style="width:10%; border: none; text-align: center;">PTF</td> <td style="width:10%; border: none; text-align: center;">DEF</td> <td style="width:33%; border: none;"></td> <td style="width:10%; border: none; text-align: center;">PTF</td> <td style="width:10%; border: none; text-align: center;">DEF</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none; text-align: center;"><input type="checkbox"/> 1</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 1</td> <td style="border: none;">Incorporated or Principal Place of Business in this State</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 4</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 4</td> </tr> <tr> <td style="border: none;">Citizen of Another State</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 2</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 2</td> <td style="border: none;">Incorporated and Principal Place of Business in Another State</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 5</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 5</td> </tr> <tr> <td style="border: none;">Citizen or Subject of a Foreign Country</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 3</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 3</td> <td style="border: none;">Foreign Nation</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 6</td> <td style="border: none; text-align: center;"><input type="checkbox"/> 6</td> </tr> </table>	Citizen of This State	PTF	DEF		PTF	DEF		<input type="checkbox"/> 1	<input type="checkbox"/> 1	Incorporated or Principal Place of Business in this State	<input type="checkbox"/> 4	<input type="checkbox"/> 4	Citizen of Another State	<input type="checkbox"/> 2	<input type="checkbox"/> 2	Incorporated and Principal Place of Business in Another State	<input type="checkbox"/> 5	<input type="checkbox"/> 5	Citizen or Subject of a Foreign Country	<input type="checkbox"/> 3	<input type="checkbox"/> 3	Foreign Nation	<input type="checkbox"/> 6	<input type="checkbox"/> 6
Citizen of This State	PTF	DEF		PTF	DEF																				
	<input type="checkbox"/> 1	<input type="checkbox"/> 1	Incorporated or Principal Place of Business in this State	<input type="checkbox"/> 4	<input type="checkbox"/> 4																				
Citizen of Another State	<input type="checkbox"/> 2	<input type="checkbox"/> 2	Incorporated and Principal Place of Business in Another State	<input type="checkbox"/> 5	<input type="checkbox"/> 5																				
Citizen or Subject of a Foreign Country	<input type="checkbox"/> 3	<input type="checkbox"/> 3	Foreign Nation	<input type="checkbox"/> 6	<input type="checkbox"/> 6																				

  
**IV. ORIGIN** (Place an X in one box only.)  
☒ 1 Original Proceeding     ☐ 2 Removed from State Court     ☐ 3 Remanded from Appellate Court     ☐ 4 Reinstated or Reopened     ☐ 5 Transferred from another district (specify):     ☐ 6 Multi-District Litigation     ☐ 7 Appeal to District Judge from Magistrate Judge
   
  
**V. REQUESTED IN COMPLAINT: JURY DEMAND:** ☒ Yes     ☐ No (Check 'Yes' only if demanded in complaint.)  
**CLASS ACTION** under F.R.C.P. 23: ☐ Yes     ☒ No     **MONEY DEMANDED IN COMPLAINT:** \$ \_\_\_\_\_
   
  
**VI. CAUSE OF ACTION** (Cite the U.S. Civil Statute under which you are filing and write a brief statement of cause. Do not cite jurisdictional statutes unless diversity.)  
 35 U.S.C. § 101 et seq., action for patent infringement
   
  
**VII. NATURE OF SUIT** (Place an X in one box only.)
 

<b>OTHER STATUTES</b> <input type="checkbox"/> 400 State Reapportionment <input type="checkbox"/> 410 Antitrust <input type="checkbox"/> 430 Banks and Banking <input type="checkbox"/> 450 Commerce/ICC Rates/etc. <input type="checkbox"/> 460 Deportation <input type="checkbox"/> 470 Racketeer Influenced and Corrupt Organizations <input type="checkbox"/> 480 Consumer Credit <input type="checkbox"/> 490 Cable/Sat TV <input type="checkbox"/> 810 Selective Service <input type="checkbox"/> 850 Securities/Commodities/Exchange <input type="checkbox"/> 875 Customer Challenge 12 USC 3410 <input type="checkbox"/> 890 Other Statutory Actions <input type="checkbox"/> 891 Agricultural Act <input type="checkbox"/> 892 Economic Stabilization Act <input type="checkbox"/> 893 Environmental Matters <input type="checkbox"/> 894 Energy Allocation Act <input type="checkbox"/> 895 Freedom of Info. Act <input type="checkbox"/> 900 Appeal of Fee Determination Under Equal Access to Justice <input type="checkbox"/> 950 Constitutionality of State Statutes	<b>CONTRACT</b> <input type="checkbox"/> 110 Insurance <input type="checkbox"/> 120 Marine <input type="checkbox"/> 130 Miller Act <input type="checkbox"/> 140 Negotiable Instrument <input type="checkbox"/> 150 Recovery of Overpayment & Enforcement of Judgment <input type="checkbox"/> 151 Medicare Act <input type="checkbox"/> 152 Recovery of Defaulted Student Loan (Excl. Veterans) <input type="checkbox"/> 153 Recovery of Overpayment of Veteran's Benefits <input type="checkbox"/> 160 Stockholders' Suits <input type="checkbox"/> 190 Other Contract <input type="checkbox"/> 195 Contract Product Liability <input type="checkbox"/> 196 Franchise <b>REAL PROPERTY</b> <input type="checkbox"/> 210 Land Condemnation <input type="checkbox"/> 220 Foreclosure <input type="checkbox"/> 230 Rent Lease & Ejectment <input type="checkbox"/> 240 Torts to Land <input type="checkbox"/> 245 Tort Product Liability <input type="checkbox"/> 290 All Other Real Property	<b>TORTS</b> <b>PERSONAL INJURY</b> <input type="checkbox"/> 310 Airplane <input type="checkbox"/> 315 Airplane Product Liability <input type="checkbox"/> 320 Assault, Libel & Slander <input type="checkbox"/> 330 Fed. Employers' Liability <input type="checkbox"/> 340 Marine <input type="checkbox"/> 345 Marine Product Liability <input type="checkbox"/> 350 Motor Vehicle <input type="checkbox"/> 355 Motor Vehicle Product Liability <input type="checkbox"/> 360 Other Personal Injury <input type="checkbox"/> 362 Personal Injury-Med Malpractice <input type="checkbox"/> 365 Personal Injury-Product Liability <input type="checkbox"/> 368 Asbestos Personal Injury Product Liability <b>IMMIGRATION</b> <input type="checkbox"/> 462 Naturalization Application <input type="checkbox"/> 463 Habeas Corpus-Alien Detainee <input type="checkbox"/> 465 Other Immigration Actions	<b>TORTS</b> <b>PERSONAL PROPERTY</b> <input type="checkbox"/> 370 Other Fraud <input type="checkbox"/> 371 Truth in Lending <input type="checkbox"/> 380 Other Personal Property Damage <input type="checkbox"/> 385 Property Damage Product Liability <b>BANKRUPTCY</b> <input type="checkbox"/> 422 Appeal 28 USC 158 <input type="checkbox"/> 423 Withdrawal 28 USC 157 <b>CIVIL RIGHTS</b> <input type="checkbox"/> 441 Voting <input type="checkbox"/> 442 Employment <input type="checkbox"/> 443 Housing/Accommodations <input type="checkbox"/> 444 Welfare <input type="checkbox"/> 445 American with Disabilities - Employment <input type="checkbox"/> 446 American with Disabilities - Other <input type="checkbox"/> 440 Other Civil Rights	<b>PRISONER PETITIONS</b> <input type="checkbox"/> 510 Motions to Vacate Sentence Habeas Corpus <input type="checkbox"/> 530 General <input type="checkbox"/> 535 Death Penalty <input type="checkbox"/> 540 Mandamus/Other <input type="checkbox"/> 550 Civil Rights <input type="checkbox"/> 555 Prison Condition <b>WARRANTS</b> <input type="checkbox"/> 610 Agriculture <input type="checkbox"/> 620 Other Food & Drug <input type="checkbox"/> 625 Drug Related Seizure of Property 21 USC 881 <input type="checkbox"/> 630 Liquor Laws <input type="checkbox"/> 640 R.R. & Truck <input type="checkbox"/> 650 Airline Regs <input type="checkbox"/> 660 Occupational Safety /Health <input type="checkbox"/> 690 Other	<b>LABOR</b> <input type="checkbox"/> 710 Fair Labor Standards Act <input type="checkbox"/> 720 Labor/Mgmt. Relations <input type="checkbox"/> 730 Labor/Mgmt. Reporting & Disclosure Act <input type="checkbox"/> 740 Railway Labor Act <input type="checkbox"/> 790 Other Labor Litigation <input type="checkbox"/> 791 Empl. Ret. Inc. Security Act <b>PROPERTY RIGHTS</b> <input type="checkbox"/> 820 Copyrights <input type="checkbox"/> 830 Patent <input type="checkbox"/> 840 Trademark <b>SOCIAL SECURITY</b> <input type="checkbox"/> 861 HIA (1395ff) <input type="checkbox"/> 862 Black Lung (923) <input type="checkbox"/> 863 DIWC/DIWW (405(g)) <input type="checkbox"/> 864 SSID Title XVI <input type="checkbox"/> 865 RSI (405(g)) <b>FEDERAL TAX SUITS</b> <input type="checkbox"/> 870 Taxes (U.S. Plaintiff or Defendant) <input type="checkbox"/> 871 IRS-Third Party 26 USC 7609
---	--	--	---	--	---

FOR OFFICE USE ONLY: Case Number: \_\_\_\_\_

SACV10-1513

AFTER COMPLETING THE FRONT SIDE OF FORM CV-71, COMPLETE THE INFORMATION REQUESTED BELOW.

**UNITED STATES DISTRICT COURT, CENTRAL DISTRICT OF CALIFORNIA**  
**CIVIL COVER SHEET**

**VIII(a). IDENTICAL CASES:** Has this action been previously filed in this court and dismissed, remanded or closed? ☒ No ☐ Yes

If yes, list case number(s): \_\_\_\_\_

**VIII(b). RELATED CASES:** Have any cases been previously filed in this court that are related to the present case? ☒ No ☐ Yes

If yes, list case number(s): \_\_\_\_\_

Civil cases are deemed related if a previously filed case and the present case:

- (Check all boxes that apply) ☐ A. Arise from the same or closely related transactions, happenings, or events; or  
☐ B. Call for determination of the same or substantially related or similar questions of law and fact; or  
☐ C. For other reasons would entail substantial duplication of labor if heard by different judges; or  
☐ D. Involve the same patent, trademark or copyright, and one of the factors identified above in a, b or c also is present.

**IX. VENUE:** (When completing the following information, use an additional sheet if necessary.)

(a) List the County in this District; California County outside of this District; State if other than California; or Foreign Country, in which **EACH** named plaintiff resides.

☐ Check here if the government, its agencies or employees is a named plaintiff. If this box is checked, go to item (b).

County in this District:*	California County outside of this District; State, if other than California; or Foreign Country
	Symantec Corporation has a principal place of business in Santa Clara County, California.

(b) List the County in this District; California County outside of this District; State if other than California; or Foreign Country, in which **EACH** named defendant resides.

☐ Check here if the government, its agencies or employees is a named defendant. If this box is checked, go to item (c).

County in this District:*	California County outside of this District; State, if other than California; or Foreign Country
On information and belief, M86 Security, Inc. has a principal place of business in Orange County, California.	

(c) List the County in this District; California County outside of this District; State if other than California; or Foreign Country, in which **EACH** claim arose.

**Note:** In land condemnation cases, use the location of the tract of land involved.

County in this District:*	California County outside of this District; State, if other than California; or Foreign Country
On information and belief, each claim for patent infringement arose in Orange County, California.	

\* Los Angeles, Orange, San Bernardino, Riverside, Ventura, Santa Barbara, or San Luis Obispo Counties

**Note:** In land condemnation cases, use the location of the tract of land involved

**X. SIGNATURE OF ATTORNEY (OR PRO PER):**

Date October 7, 2010

**Notice to Counsel/Parties:** The CV-71 (JS-44) Civil Cover Sheet and the information contained herein neither replace nor supplement the filing and service of pleadings or other papers as required by law. This form, approved by the Judicial Conference of the United States in September 1974, is required pursuant to Local Rule 3-1 is not filed but is used by the Clerk of the Court for the purpose of statistics, venue and initiating the civil docket sheet. (For more detailed instructions, see separate instructions sheet.)

Key to Statistical codes relating to Social Security Cases:

Nature of Suit Code	Abbreviation	Substantive Statement of Cause of Action
861	HIA	All claims for health insurance benefits (Medicare) under Title 18, Part A, of the Social Security Act, as amended. Also, include claims by hospitals, skilled nursing facilities, etc., for certification as providers of services under the program. (42 U.S.C. 1935FF(b))
862	BL	All claims for "Black Lung" benefits under Title 4, Part B, of the Federal Coal Mine Health and Safety Act of 1969. (30 U.S.C. 923)
863	DIWC	All claims filed by insured workers for disability insurance benefits under Title 2 of the Social Security Act, as amended; plus all claims filed for child's insurance benefits based on disability. (42 U.S.C. 405(g))
863	DIWW	All claims filed for widows or widowers insurance benefits based on disability under Title 2 of the Social Security Act, as amended. (42 U.S.C. 405(g))
864	SSID	All claims for supplemental security income payments based upon disability filed under Title 16 of the Social Security Act, as amended.
865	RSI	All claims for retirement (old age) and survivors benefits under Title 2 of the Social Security Act, as amended. (42 U.S.C. (g))