(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
16 March 2006 (16.03.2006)

PCT

(10) International Publication Number
WO 2006/029222 A2

(51) International Patent Classification:
H04L 9/00 (2006.01)

(21) International Application Number:
PCT/US2005/031901

(22) International Filing Date:
7 September 2005 (07.09.2005)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

10/935,337	7 September 2004 (07.09.2004)	US
10/935,639	7 September 2004 (07.09.2004)	US
10/935,260	7 September 2004 (07.09.2004)	US

(71) Applicant *(for all designated States except US)*: **ICONIX, INC.** [—/US]; 1055 Joaquin Rd., Suite 101, Mountain View, California 94043 (US).

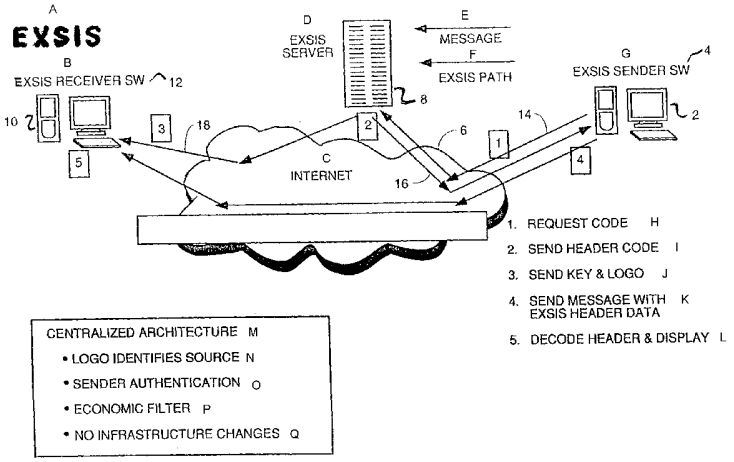
(71) Applicants and
(72) Inventors: **ZAGER, Robert, Philip** [US/US]; 20292 Calle Montalvo, Saratoga, California 95070 (US). **AMES, William** [US/US]; 6545 Sloping Meadow Ct., San Jose, California 95135 (US). **PICAZO, Jose, Jesus** [US/US]; 741 Kalthoff Common, Livermore, California 94550 (US). **VEMPATY, Nageshwara, Rao** [US/US]; 3506 Waverley Street, Palo Alto, California 94306 (US). **DUVVOORI, Vikram** [IN/US]; 17561 Vierra Cyn. Rd., No. 140, Salinas, California 93907 (US). **TRYTTEN, Chris, David** [US/US]; 1230 Crescent Terrace, Sunnyvale, California 94087 (US).

(74) Agents: **STEFFEY, Chuck** et al.; Schwegman, Lundberg, Woessner & Kluth, P.O. 2938, Minneapolis, Minnesota 55402 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, ZA, ZM, ZW.

[Continued on next page]

(54) Title: USER INTERFACE AND ANTI-PHISHING FUNCTIONS FOR AN ANTI-SPAM MICROPAYMENTS SYSTEM



(57) Abstract: Techniques for protected email transmission using micropayments and a segregated inbox in which protected emails are displayed are provided. The techniques may involve authentication of the sender to defeat phishers and an opt-out option, which can be used to block protected emails from sources from which the user no longer wishes to receive emails even if the source has made a micropayment. Custom indication techniques for email is also taught where a sender of protected emails can pay extra to have a miniature version of its custom indication displayed with its email in the segregated inbox. A white list is maintained (along with the opt out black list) so that recipients can designate specific senders who may send email to that recipient without paying a micropayment and still have the protected email displayed in the segregated inbox.

WO 2006/029222 A2

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

USER INTERFACE AND ANTI-PHISHING FUNCTIONS FOR AN ANTI-SPAM MICROPAYMENTS SYSTEM

Claim of Priority

The present application claims the priority benefit of U.S. Application Serial Number 10/935,260 filed September 7, 2004, U.S. Application Serial Number 10/935,639 filed September 7, 2004, and U.S. Application Serial Number 10/935,337 filed September 7, 2004, all entitled "USER INTERFACE AND ANTI-PHISHING FUNCTIONS FOR AN ANTI-SPAM MICROPAYMENTS SYSTEM." The entire contents of all three of these applications is incorporated herein by reference.

Background

Unsolicited email is becoming a large problem on the Internet. One solution to this problem is the use of a micropayments system where emails from sources not on a recipient's white list do not get to recipients unless the sender has taken steps to make a small payment called a micropayment. Emails for which micropayments have been made include in them coding which is called a stemp and which indicates the micropayment has been made.

Micropayments are effective to stop mass emailers who send out millions or tens of millions of unsolicited emails per day to users who do not want to receive them. Even a payment of 0.01 cents per email will be a considerable amount of money when multiplied by millions of emails per day. However, bulk emailing of unsolicited emails is not the only problem the Internet suffers from.

Phishing has become a large problem in email communications over the Internet. Phishing involves passing oneself off as a legitimate business from which a user may actually want to receive unsolicited emails. Phishing has experienced exponential growth over the Internet. Spam typically gets a response rate of 0.01%. However, phishing, because of the air of legitimacy and the strong call to action, can receive a 5% response rate.

Phishing is done with email messages which look very similar or identical to email messages that come from the legitimate business and contain an urgent call to action with a request to click on a link. Clicking on the link

takes the unsuspecting user to the phisher's web page where personal information such as credit card numbers or bank account numbers is sought. If the user supplies the information, an identity theft often follows.

There are two kinds of phishing today. The first uses a long Universal Resource Locator (URL) address to obscure the criminal's actual domain name. The second, called social phishing, uses perfectly legitimate domain name registrations, which can be purchased for extremely low prices from some sources but which include a famous trademark of another company.

Email presents four challenges to commercial senders.

1. Filters: Filters that block commercial email for some service providers.
2. Phishing: The problem with phishing has been discussed above.
3. Clutter: Marketers need a way to break out of the clutter associated with user's email inboxes.
4. Opt-outs: As legitimate companies get mailing lists from numerous sources and use numerous outside agencies to send email, opting out becomes very hard for the senders of email to administer.

Summary

According to one aspect of the invention, a machine-implemented method is provided where a transaction number is received from a payment service. An email sent from a sender is inspected to acquire an encrypted account number from its header. The identity of the sender is requested of the payment service by parsing the transaction number and the encrypted account number. A reply is received from the payment service that authenticates or does not authenticate the identity of the sender.

In another aspect of the invention, a machine-implemented method is provided where information is decrypted to acquire the identity of sender of a message; the message is directed to a recipient and a transaction occurs before the recipient receives the message from the sender. Next, it is determined whether the sender has an account with sufficient funds to pay for the transaction or whether the sender is permitted to proceed with sending the message free of charge. Funds are acquired from the account if appropriate, and if the sender is

verified a stemp is encrypted with the transaction number and an encrypted account number. The stemp is sent back to the sender to use with the message.

In yet another aspect, a machine-implemented method is provided where protected emails are presented and a selection for one of the protected emails is received. A logo or stemp is decrypted to acquire a transaction number for the selection and the transaction number used to acquire a key. The key used to decrypt the stemp further for an account number of the sender. The account number is used to authenticate an identity of the sender.

In another aspect, a machine-implemented method is provided where an opt-out request is received from a recipient. An identity for a particular sender associated with the opt-out request is obtained and the identity is added to an opt-out list associated with recipient.

In yet another aspect, a machine-implemented method is provided where an interface is presented for viewing and managing emails directed to a recipient. A navigation pane is displayed having a paid email icon, a free email icon, and an additional icon not associated with paid or free emails. A content pane is also displayed. Emails are activated in the content pane when the icons to which they are associated are activated in the navigation pane.

In still a further aspect, a machine-implemented method is provided where a request for a stemp is received from a sender of a message; the message being directed to a recipient. The identity of the sender is authenticated and an account of the sender is debited if payment is dictated by a policy of the recipient. Next, the stemp is generated as an encrypted version of the sender's account number and a transaction number for the message. The stemp is then sent to the sender for use when sending the message to the recipient.

In an aspect of the invention, a machine-implemented method is provided where an email is received from a sender and a header of the email is inspected for a stemp. Authentication of the stemp is requested and the email is routed to a segregated inbox if the stemp was present in the header and authenticated.

According to an aspect, a system is provided that includes a recipient interface and a protected email service. The recipient interface is for requesting that the protected email service authenticate senders of emails. The protected

email service collects payments from senders of the emails and authenticates the identities of the senders on behalf of the recipient interface.

In another aspect, a machine-accessible medium having instructions embedded thereon is provided. The instructions when accessed by a machine routes an email from a sender to a segmented inbox in response to a stemp that was included in a header of the email. The instructions, when accessed, also send the header to an authentication service to authenticate the identity of the sender. The email or information about the email is then displayed or discarded in response to information received from the authentication service.

In still another aspect, instructions are provided within a medium that, when accessed by a machine, supplies stemps to senders of emails directed to recipients and authenticates the stemps on behalf of the recipients. Also, a number of senders may be blocked from sending their emails if so requested by the recipients. Moreover, funds may be debited from accounts of the senders if a policy of the recipient necessitates payments.

Brief Description of the Drawings

Figure 1 is a diagram of example micropayment system hardware.

Figures 2A and 2B show a flowchart of an example embodiment of a basic micropayment protocol.

Figure 3 is a drawing of an example screen shot for the user interface of an embodiment which uses split screens segregation of emails, custom indications on emails with stemps and a separate folder for micropayment email.

Figure 4 shows an example user interface display which uses a single pane content window to display emails that either contain stemps or are from white list recipients.

Figure 5 is a screen shot representing another class of example embodiments which requires users to ask for particular emails to be authenticated by selecting the email and selecting an authentication command and gives a user an option to opt out.

Figures 6A and 6B are a flowchart of an example process carried out by a recipient computer and server to implement the opt-out option.

Figure 7 is a flowchart of an example embodiment of a manual authentication process.

Figures 8A and 8B show a flowchart of an example generic manual authentication process.

Figures 9A, 9B and 9C show a flowchart of the embodiment of a protocol to send and receive protected email.

Figure 10 shows an example a table or database which can be used by a server to map from a transaction number to an identity.

Figure 11 shows a diagrammatic representation of machine in the exemplary form of a computer system within which a set of instructions, for causing the machine to perform any one or more of the methodologies discussed herein, may be executed.

Detailed Description

Referring to Figure 1, there is shown a diagram of an example micropayment system. A computer 2 of a sender executes sender client software 4 which controls the computer 2 to communicate in accordance with a protocol to be described below over the Internet 6 with a server 8. Typically, the sender client software 4 is software which is downloaded from a server 8 when the user establishes an account. The server 8 implements a micropayment process and communicates both with the sender computer 2 and a recipient computer executing receiver client software 12. The server 8 may be referred to as a "protected email server" and the sender software 4 and receiver software 12 may be referred to as "protected email software".

The receiver software 12 may be software which is downloaded from the server 8 when the user establishes an account. Users of sender computer 2 and recipient computer 10 may simply download and install the software and a parallel universe for segregating paid and white-listed email within their computers may be established. In an embodiment, the sender software 4 and the receiver software 12 are the same piece of software which has capabilities described herein to both send and receive email. Advanced users can manage the sender and receiver software using the rules of their email if they so choose.

The sender and receiver computers and the server are typical computers with keyboards, pointing devices, displays, central processing units and monitors.

Figure 2, comprising Figures 2A and 2B, is a flowchart of an example embodiment of a micropayment protocol. When the sender wishes to send a paid email or a white list email (e.g., free but allowed in by the receiver software 12 because the sender is on the recipient's white list of friends, associates, family members and other entities the recipient consents to receive from), a command is given to the sender software indicating the desire to send an email to a particular, identified recipient (step 1). A message is then composed for sending and when the message is ready to send, the send command is given (step 3). The input data includes a user name and password identifying the sender in the protected email environment. The password (and user name in some embodiments) will be encrypted and sent to the server in an initiation message with a request for an encrypted stemp. In some embodiments, the user may enter her account number which will be encrypted and sent with the request for the stemp in the initiation message. The user name and password or account numbers may be referred to as sender identification information.

In response to the send command, the sender software 4 sends an initation message to the server identifying the recipient with the sender identification information (e.g., sender identification may be provided by user name and encrypted password or by account number or by user name, password and some secure information that only an authentic sender would know) and requesting the server to send back a code (e.g., a stemp) which may cause the message to be not blocked by the recipient software 12 (step 5). In this example embodiment, this message includes a user name and password (e.g., encrypted) and this information is used to look up the sender's account number. In other embodiments, this message may include the encrypted account number of the sender. It also includes the header of the email, and this header includes the sender's IP address and the recipient's email address. The server uses the user name and decrypted password or the decrypted account number to determine the identity of the sender, and this process acts as an automatic authentication of the user. The requested code can, in some embodiments, be an encrypted Truemark™ logo in the case of a paid email which is identified with the sender's

brand. It is to be understood that the Truemark™ may actually be any custom indication that is graphical and/or audible in nature and permits ready identification. The requested code can also be an encrypted generic logo which serves as a paid email code indicating that the sender paid to transmit the email with his validated identity encoded therein and wants to be segregated into the email inbox to be away from all the unpaid spam but has not paid to have the company logo displayed. Or the requested code can be an encrypted white list logo indicating the sender is on the recipient's white list. Thus, there may be three classes of protected email as distinguished by their stemps: 1) custom identification stemps which are associated with paid senders who conspicuously identify themselves with their own trademarks, custom graphics, and/or custom audio; 2) generic stemps associated with paid senders who do not conspicuously identify themselves; and 3) white list stemps associated with senders that the recipient has white listed. All three types of stemps are issued, in an automatic authentication embodiment, by using the sender's account information as identified from his user name and password and the validation process.

The custom indication stemps may in some cases be logos or trademarks. The paid email generic logo type stemp may be yellow in an embodiment. The unpaid email from a white list (e.g., a white list stemp) may be green in an embodiment. In alternative embodiments, different shapes can be used instead of different colors. The key thing is any person can distinguish the types (and the sender in the case of custom indications) with no training. Of course, with whitelist senders, the purported sender shows up in the "from column," so odds are the recipient knows who sent the email, and can trust that "from" designation because there is also the validation of the whitelist stemp.

For the authentication results (e.g., sender ID data screen) the server can display anything on the recipient computer display to identify the authenticated sender for various embodiments. One embodiment shows the trademark registration for the logo, but generic stemps and whitelist stemps would be substantially less informative -- just showing whatever ID was used to set up the protected email account. White list senders get their integrity from the fact that the recipient designated who is on the white list. Generic stemps have a penny's worth of integrity (only a small micropayment was paid to have the generic

stemp issued) -- that's why they may be yellow and that is why, in some embodiments, they are not permitted. However, they may have lots of integrity; for example, they may cost $4M to send 400 million emails.

Returning to the consideration of Figure 2, the server, in the basic embodiment, receives the initiation message in step 5 and may, for example, do the following things. 1) It may use information in the header of the email to be sent such as the IP address of the sender and/or the sender's account number and/or the user name and decrypted password and/or other secure information only the sender would know to authenticate the sender as actually being the sender the email purports to be from (step 7). This involves decrypting the account number in the incoming message, and looking up the identity of the entity with that account number. 2) It may make sure the sender has a micropayments account with sufficient money in it to cover the cost of the stemp and the sender is not on a black list or opt out list of the recipient or the sender is on the recipient's white list (step 9). 3) If conditions in step 2 are determined to be proper (e.g., an account with sufficient money in it exists for this authenticated user, no black list entry in recipient's black list for this sender or sender is on recipient's white list) a transaction number may be assigned, a custom indication or stemp that contains the ID of the sender and the transaction number may be encrypted, the encryption key may be saved, the amount of the stemp may be deducted from the sender's account if the sender is not on the recipient's white list, and the encrypted stemp may be sent back to the sender software 4 (step 11 and message 16 in Figure 1). The stemp may be generated by generating a transaction number when the sender has been authenticated. In some embodiments, this transaction number is encrypted and the account number of the sender is encrypted. In these example embodiments, these two pieces of encrypted data may then be concatenated or otherwise merged and a hash of the result is performed. The resulting hash is the stemp or code which links the sender to his custom indication. This code may then be sent back to the sender process by incorporating it into the proposed email message header and sending the header back to the sender. 4) The server may then send the encryption key, the encrypted transaction number and the corresponding custom

indication logo for the sender to the receiver software 12 (step 13 and message 18 in Figure 1).

In step 15, the sender software 4 receives the header with the encrypted stemp in it back from the server. This header is put on the email message to be sent to recipient, and the email message is then sent to the recipient through normal channels on the Internet or some other WAN.

In steps 17 and 19, the receiver software receives the email with encrypted custom indication, generic stemp or white list stemp, and retrieves the encryption key used to encrypt the custom indication, generic stemp or white list stemp. The custom indication, generic stemp or white list stemp may be dehashed using the reverse hash algorithm. The key is then used to attempt to decrypt the account number and the transaction number encoded in the custom indication, generic stemp or white list stemp, and if the custom indication, generic stemp or white list stemp decrypts properly, the transaction number should match the transaction number received from the server and the account number should match a table entry matching account numbers to custom indications (which table may be stored in the recipient computer in some embodiments). If these items match what the recipient computer has, the email is stored in an Exsis (payment service) email folder with its custom indication. The emails in this folder will be displayed whenever selection of the custom indication icon in the navigation pane of the recipient's browser or email client occurs.

In some embodiments, the custom indications, stemps or generic logos that are displayed with the emails in the email folder contents are the custom indications, stemps or generic logos received from the server. In other embodiments, the recipient computer maintains a table that associates each account number with a corresponding custom indication or generic logos, and that table is consulted to find and display the appropriate custom indication or generic logo when a custom indication, generic stemp or white list stemp in a received email is received and properly decrypted and the account number is retrieved therefrom.

In order to spoof this system, a phisher would have to do the following things: Spoof the server into generating a transaction number by sending a

message with the proper account number of the sender the phisher wants to pretend to be and properly encrypt that account number. In other words, the phisher would have to know the account number of the business he wanted to pass him self off as and know the proper encryption algorithm and key that that business would use to encrypt its account number. Then the phisher would have to obtain the transaction number that had been sent to a recipient in response to the phisher's request or decrypt the transaction number sent from the server to the recipient. The phisher would then have to receive the encrypted stemp it spoofed the server into generating or know the encryption and hashing algorithms used to generate the stemp and generate the same stemp the server would generate had a request from the authentic sender been sent to the server.

Referring to Figure 3, there is shown an example drawing representing several species within a genus of browser user interfaces to segregate paid email and white list email from other email that does not fall into either of those categories because it was not sent as a result of an exchange of messages with the server. There are several species within this genus each of which will be described below by way of example. Figure 3 represents an example first species where both paid and white list emails are simultaneously displayed. This species is represented by the panes 48 and 58 which are simultaneously displayed when the email service icon 68 in navigation pane 44 is selected. Another example species is represented by the Logo Mail and Friends tabs 86 and 88. In this species, when the email service icon 68 is selected and the Logo Mail tab 86 is selected, paid emails from the current day show up in pane 48, and paid emails from the previous day show up in pane 58. When the friends tab 88 is selected, emails from friends and family on the white list from the current day show up in pane 48, and white list emails from the previous day show up in pane 58.

We now proceed to a more detailed discussion of the example elements of the display shown in Figure 3. The display shown in Figure 3 may represent the browser mail user interface for a computer which has been programmed with software to use segregation of paid and friends emails from other email using an email service folder in the navigation pane, and to use company custom indications on emails with stemps and tabs to segregate micropayment email

from white list email. The browser window outline is shown at 30. The normal drop down menu commands such as "file", "edit" etc. are shown at 32 and 34 as examples. The conventional "back" and "forward" commands to navigate the browser back to the previous web page and forward to the next web page in a sequence of web pages already visited are shown at 36 and 38, respectively. Icons, for normal email commands such as "get email" etc., are shown at 40 and 42.

Box 44 represents an example navigation pane. The items on the navigation pane can be selected to control what is displayed in the content pane 46. The conventional items in the navigation pane are shown by way of example to include the inbox icon 60, the sent mail icon 62, the trash icon 64, and the "my folders" icon 66. When the inbox icon 60 is selected, all contents of the inbox (e.g., emails not sent as a result of an exchange with a micropayment server) may be displayed in the content pane. When the sent mail icon 62 is selected, the subject lines and other inforamation about all the emails the user has sent may be displayed in the content pane 46. When the trash icon is selected, all the emails the user has deleted may be displayed in the content pane 46.

All species within the genus may, for example, include an email service icon shown at 68 and located somewhere in the navigation pane 44 (an alternative location is shown at 72). This icon, when selected may cause all emails to be displayed in the content pane 46. These emails may be emails that are paid for and white list email sent as a result of an exchange between a client process on the sender's computer and the micropayment server.

In some embodiments, the email service icon 68 is displayed in the top half of the navigation pane above the address book icon 70. In alternative embodiments, the email service icon 68 is shown in the lower half of the navigation pane such as at 72. In each embodiment within the genus, the email service icon 68 may show up somewhere in the navigation pane and selection of may will cause the segregated email to be displayed in the content pane 46.

When the email service icon 68 is selected, as it is in Figure 3, as indicated by the box around the icon indicating highlighting in a user interface, the contents of the content pane may be a listing each of the emails, e.g., emails

that have been sent after an exchange between the micropayment server and the client browser that sent the email. Some of the emails sent as a result of this exchange with the micropayment server may have encrypted stemps representing micropayments embedded in the email header. These emails may usually be from businesses who want to send a message to one of their customers and make sure it gets seen by the customer and not thrown out or blocked by the anti-spam functions of the recipient browser. These emails may be displayed with custom indications (e.g., a custom image, graphic or audio snippet of the sender). In some embodiments, the custom indications are only displayed after the sender of a paid email has been authenticated. In other embodiments, the paid emails will be displayed with their custom indications provisionally when received, and, if after a user has requested authentication of the email, the email turns out to not be authentic, the email and its custom indication may be removed from the inbox.

In the example simultaneous paid and white list embodiments shown, the content pane has been split into two separate panes, the top pane 48 being for paid emails which include stemps, and the bottom pane 58 being for unpaid emails from friends or other people or organizations on the recipient's white list. In some other embodiments, the contents of pane 48 is displayed if the Logo Mail tab shown at 86 is selected, and the contents of pane 58 are shown if the friends tab at 88 is selected. In still other embodiments, both panes 48 and 58 are shown simultaneously in a split pane format as shown in Figure 3 and organized by date of the email with paid emails being shown if the Logo Mail tab is selected and white list email being shown if the friends tab is selected. In other embodiments, only a single window to the right of the navigation window 44 is shown, and its contents display email with stemps if the Logo Mail tab 86 is selected and its contents display the white list emails if the friends tab 88 is selected.

Thus, in an embodiment, paid emails in panes 48 are segregated from unpaid "white list" emails in pane 58. Emails that appear in the panes 48 and 58 may be emails that have been processed by an exchange with the server. All other emails that are sent from a sender to a recipient by normal channels without an exchange with the micropayment server are the emails that show up

in the recipient's inbox when the inbox icon 60 in the navigation pane 44 is selected.

In the case of the unpaid "white list" emails in pane 58, the sender's email client may include one or more processes that communicate with the micropayment server telling it that the sender would like to send an email to recipient whose inbox display of the recipient browser or email client is shown in Figure 3. In an embodiment, the sender's email client need not send the entire text of the email to the server but does send the header. The server checks the white list it maintains for the recipient and finds that the sender is on the white list. The server then sends back the header of the email with coded data added to it which indicates that the sender is on the white list of the recipient and need not include a micropayment stemp in the email header. The sender's email client then attaches the header received from the server to the email and sends it to the recipient via, for example, normal channels on the Internet. This process of communicating with the server to tell it the sender wants to send email to a particular recipient and sending the header of the email and receiving back a modified header from the server coded as "white list", attaching the modified header to the email and sending the resulting email to the recipient through normal channels is implemented by sender software executing on the sender's computer and cooperating with the operating system and email client and/or browser thereof. There is no need for the email to pass through the server, although this may be done in some species.

When the email arrives at the recipient computer, the email client or browser may invoke an add-on process implemented by recipient software. In some embodiments, this add on process in the recipient computer is a plug-in which is active at all times. This add-on process may check the header of the incoming email for coding indicating it has a stemp or is a white list email. If it has a stemp, an authentication process to be described later herein is made, and, if the email is authentic, an entry is made in pane 48. If the email is coded as a white list email, its source is automatically authenticated in one embodiment, and, if the email is actually from the white list person or company it purports to be from, an entry is made in pane 58. If the email is authentic and has a stemp, an indication as to who the email is from is given in the email listing in pane 48.

If the email is authentic and is from a white list recipient, an indication of who the email is from is displayed such as is shown by the names at 76, 78 etc.

In an embodiment, the indication of who an authentic email bearing a stemp is from is given in the form of an icon which may be a miniature duplication of the custom indication of the company or person that sent the email. The custom indication can be stored in the recipient computer in some embodiments or, in other embodiments, is downloaded from a server such as the authentication server or the server (also referred to herein as the protected email server or the secure micropayment server). An example of such a custom indication icon indicating the authentic source of the email is shown at 74 in pane 48. This example email is from Amazon.com, and the fact that the custom indication icon at 74 is displayed at all is a representation to the recipient that the email has been authenticated in the manner described below and is actually from Amazon.com (in some embodiments, the custom indication is displayed on all incoming protected emails even if they have not already been automatically authenticated, and the user knows that the custom indication can be authenticated using the manual procedures described herein). If an email is from a white list recipient who does not have a custom indication, the icon displayed after authentication can be a generic icon provided by the email service for some small payment such as the cost of a stemp. In other embodiments, white list icons can be anything the recipient configures in his address book such as a miniature digital photo of the person, their name or a nickname.

The custom indication logos may be protected from phishers because of the process of obtaining them. When a legitimate company with an account makes a request for a stemp with the proper user name and password, he is authorizing the server 8 to access his financial account. Assuming the user name and password are correct, an account exists and it has sufficient money in it, the stemp will be issued and the logo of the sender will be sent to the recipient. The logo will be either automatically authenticated before sending or it will be automatically authenticated when received by the recipient software or the user may request the logo be authenticated by making a manual request. As long as the owner of the custom indication logo maintains adequate control of his user name and password, no phisher will be able to palm off a fake logo in the

segregated inbox. In an embodiment, recipients know this so they can view emails messages in the inbox at any time because they know the message has either already been authenticated or the recipient can authenticate it with a couple simple keystrokes using, for example, an example dialog box 100 in Figure 5.

Every custom indication, even the generic logo will be encoded with the ID of the user who affixed it to his email. This happens when the account of the sender is accessed after the sender provides his user name and password. The server knows the entity that has the user name and password entered with the request to issue a stemp and send the user's logo. In order to get an account associated with a branded stemp, e.g., a logo called a custom indication, a person establishing the account may have to go through a due diligence investigation. This may be conducted by employees and may require evidence that the sender/user attempting to establish the account and custom indication is currently employed by a company they say they are employed by, and has the authority to establish the account on behalf of that company. What may distinguish a custom indication or branded stemp from trusted certificates includes the following: 1) the custom indication does not require a genius to understand it; 2) the custom indication is extremely consumer friendly; 3) the custom indication may work in Outlook and webmail whereas the trusted certificates may not; 4) the quality of trusted certificates may be variable with some certificates being not so trustworthy.

A similar due diligence process may be performed for white list senders who wish to establish an account and use a generic logo. However, the due diligence process may be eliminated or highly attenuated in some embodiments since the phishing problem is peculiar to large companies who lend credibility to the phisher's call to action. They generally do not pass themselves off as individuals. Technically, a phisher may be able to establish a false account in somebody else's name and set up a user name and password, if they know enough about that person to perform an identity theft. But it is unlikely that such a spoof would pass the due diligence if the account set up was attempted by a phisher in the name of a big corporation. Encoding the sender's ID in the custom indication or stemp will help block phishers because even if the phisher is able to

spoof the system and establish a fraudulent account, when the recipient's complain, the custom indication or generic logo will be sent back to email service and decoded to reveal the identity of the person who established the fraudulent account. This will lead to arrests and prosecutions and cause phishers to avoid an environment implementing the teachings presented herein.

The target customers for custom indications are entities whose brand identity is being damaged by phishers.

The functionality of the native browser or email client is used in some embodiment by the add-on process to display the subject line of each email, e.g. 80, 82 and 84, and to display conventional icons or other user interface tools such as bold or highlighting of unread messages to indicate whether a message has already been read or not. In other embodiments, the add-on process may do all the work of displaying the emails including all the work normally done by the conventional browser or email client.

If an email comes in, which does not have a stemp nor a coding to indicate it is from a white list recipient, it may be placed in the inbox which is displayed when the inbox icon 60 is selected.

In summary, in an example embodiment, every time an email is received, the add-on process executing in the recipient computer may automatically authenticate the source of the email by a process to be described below, and will not display it in pane 48 or 58 unless it is authentic. Authentic emails are displayed with custom indication icons in pane 48 and with whatever the recipient configures in his white list address book for white list recipients.

Figure 4 is a diagram of an example user interface display which uses a single pane content window to display emails that either contain stemps or are from white list recipients. In the display of Figure 4, an email service icon 90 is displayed in a navigation pane 92 and is selected. This causes a content window 94 to display a single pane of emails which both contain stemps and which are from senders on the recipient's white list. For example, the email from CFO.com whose custom indication is displayed at 96 (e.g., because it has been authenticated as actually being from CFO.com) is displayed along with an unpaid email from a person Bobby on the white list of the recipient, with an icon displayed at 98 indicating this email has been authenticated. The user interface

of Figure 4 works the same way as the user interface of Figure 3 except that only a single pane of emails is displayed with stemp email mixed in with non stemp email. No Logo Mail or Friends tab is necessary therefore, and these tabs 86 and 88 from Figure 3 are omitted in the display of Figure 4. In the particular species of Figure 4, the content window 94 is divided into sections for emails dated today and emails dated yesterday, but in other embodiments, only a single pane with both stemp and non stemp emails mixed together is used with the ability to sort on any field.

A common element between example embodiments of user interfaces according to one genus, two examples of which are represented by Figures 3 and 4, is that the white list emails are displayed or can be displayed in the email service folder along with the paid emails. This attracts users to click on the folder which they would be less inclined to do if they knew that the folder only contained paid emails. This gives advertisers a way to use the Internet to send cheap advertising that is known to the users to be authentic and which is segregated out from all the spam in the user's inbox by virtue of being segregated from the other email in the inbox 60 by being placed in the folder. The fact that white list emails will also be in this folder gives such advertisers a much improved chance that the users will click on the folder navigation icon 68 or 90 and have their emails viewed. The automatic authentication process or authentication on demand protects such users from fear they are being "phished" and further improves the chances of a response to a legitimate paid email. This user interface genus provides advertisers with an entirely new, less expensive, secure way to get advertising messages to a targeted list of customers or potential customers.

The particulars of the conventional browser interface and email commands can vary from one type of browser and email client to the next, and are not critical to the example embodiments. For the user interface claims, the details of how the recipient computer add-on process processes incoming emails, interacts with the operating system or interacts with the conventional browser or email client are not critical to the example embodiments. It is only how the computer displays the email service icon in the navigation pane and how the paid

and white list emails are segregated and displayed alone by selection or together simultaneously that constitute the user interface embodiments.

Each of the example embodiments discussed so far may include automatic authentication of emails to verify they are from who they purport to be from. Figure 5 is an example screen shot representing another class of example embodiments which requires users to ask for particular emails to be authenticated by selecting the email and selecting an authentication command and gives a user an option to opt out. The opt out option means the user can request the server to block further emails from the sender of a particular email even if the sender attempts to attach a paid up stemp to the email. The options are provided in a pop up dialog box 100. This pop up dialog box can be caused to appear in many different ways such as by typing a hot key combination when a particular email is selected or the cursor is resting on the email or the custom indication of the email, or by clicking on the custom indication or logo of the email or by a drop down menu selection added to the conventional drop down menus. Any user interface mechanism that provides the ability of a user to select a particular email in the inbox and provides commands to authenticate or opt out will suffice. In an embodiment, the pop up dialog box 100 is caused to appear when the user double clicks on the custom indications of an email he wants to have options regarding. For example, suppose a user wants to opt out of receiving any further emails from CFO.com or wants to validate an email purporting to be from amazon.com. To opt out of emails from CFO.com, the user may double click on the CFO custom indication 96 in Figure 5, and that may cause dialog box 100 to appear. The user would then double click on the "opt out" command in the dialog box 100. This would cause the process shown in the flowchart of Figures 6A and 6B to occur.

In addition or instead, the receiver software 12 in Figure 1 may include the ability to send email messages to friends, family and others the recipient wants on his white list informing them they have been placed on his white list. Simultaneously, a message may sent to the server 8 which requests the server 8 to put the friend, etc. on the recipient's white list and giving the server 8 the email address of the person or other entity to be added to the white list. The server responds by putting the email address of the sending on the white list of

the recipient who sent the message requesting a particular person or entity to be added to the white list. This gives users of the system the ability to cause certain persons emails to be sent for free and to be segregated into the user's inbox.

Figures 6A and 6B show a flowchart of example processes which occur at the recipient computer, the server and the sender computer when a user has selected an opt out command from the pop up dialog box on the recipient's computer. Step 100 represents the process of the user selecting a particular email and giving a command to cause the pop up box 100 to appear. In an embodiment, both things (selecting the email and causing the pop up box 100 to appear) are accomplished simultaneously by double clicking on the custom indication of the email to be subjected to the opt out process. In alternative embodiments, this may be done by clicking once on the email to be selected, typing a hot key combination to cause the dialog box 100 to appear and then double clicking on the opt out function. The dialog box 100 can also be made to appear in any other way such as by dragging the cursor over or double clicking on an icon on the screen somewhere or selecting it as a command bar icon or a drop down menu item from some other command bar icon such as the edit icon 34 in Figure 3.

Step 102 represents the process of selecting the opt out icon. In an embodiment, double clicking the custom indication causes the pop up box to appear, but then the user must double click on the opt out command to start the opt out process.

Step 104 represents the process carried out by example email service plug-in software in the recipient computer of sending the required information from the header of the email selected by the user for opt out back to the server with a message that indicates this user wishes to opt out from receiving any more paid or unpaid emails from the sender identified in the header information transmitted back to the server. Typically, this information from the header will be the encrypted micropayment account number and the transaction number or the IP address of the sender, etc. IP addresses are not reliable though since IP addresses are dynamically assigned to most users by DHCP servers at the ISP or router at the edge of the LAN on which the sender computer resides.

In step 106, the Exsis server receives the opt out request message and uses the information from the header of the email selected for opt out to look up the sender of the email. Typically, this is done by using the transaction number to look up the encryption key used to encrypt the micropayment account number and using that key to decrypt the account number. Then the account number is used to look up the identity of the sender. If the account number does not decrypt properly, the email is not from whom it purports to be from, and a message is sent to the recipient computer indicating the email is not from whom it purports to be from.

In step 108, the server adds the sender's identity to an opt out list maintained on the server for the recipient.

Step 110 is optional. If this step is performed, the server sends the opt out request message and recipient identity to the sender software of the sender computer of the sender identified in the opt out request. This allows the sender to remove this recipient from its lists of target recipients thereby saving the micropayments for emails to that recipient.

After adding the sender to the recipient's opt out list, the server receives in step 112 requests from a sender process running on a computer of a blocked sender. The server then looks up the sender identity from the message received from the sender and compares that identity (after authentication) to the opt out list of the recipient to whom the sender indicated it wants to send a message.

If the sender is on the opt out list of the recipient, the server refuses to send back an encrypted stemp or white list code to the sender thereby effectively blocking that sender from getting any email messages into the email service folder of the recipient, as represented by step 114. If the sender sends email anyway, the receiver process software shunts such non email into the normal inbox of the recipient email client or browser or blocks it altogether.

This opt out function works because of the centralized nature of the economic process to obtain stemps. By simply blocking the process of sending back a stemp (or a white list code) when a sender is on an opt out list, a sender can be effectively blocked from getting any email into the inbox of the recipient.

Figure 7 is a flow chart of an example manual authentication process of an embodiment. Step 118 represents the process of manually launching a

separate, fresh browser window by invoking a browser icon on a desktop or in an applications folder. In an embodiment, this browser should not be launched from another application such as a word processor, accounting program, etc.

Step 120 represents the process on the recipient computer of receiving user input to point the browser to URL www.whofrom.com. Typically, this involves receiving user keystrokes to enter this URL address in the address bar of the fresh instance of a browser and receiving a command to visit that web page. This web page is typically implemented on the server, but it could be implemented by another server on the Internet which has copies of (or shared) account numbers, decryption keys, and transaction numbers stored on the server.

In response to invoking the www.whofrom.com address, the fresh browser instance retrieves the web page and displays it (step 122). This web page has a "drop stemp/custom indication from selected email here" box displayed. The recipient computer then receives input from the user who may drag to this box the custom indication or stemp icon from an email displayed in the email service folder whose source is to be authenticated (step 124). This may cause the recipient computer to send the header or at least the stemp of the header of the selected email to the server with a request to authenticate the source thereof (step 126).

The reason this example drag and drop to a web page user interface is used is to prevent the user from being victimized by a spoofed stemp linked to a criminal link. The bad guys now have strong ability to use Java to paint over any screen, so that the manually entered URL on a fresh browser screen is essential to prevent spoofing.

In step 128, the server receives header information and dehashes the stemp. It then decrypts the transaction number encoded into the stemp and uses that transaction number to look up the decryption key used to encrypt the account number encoded into the stemp. That key is then used to decrypt the account number. Finally, in step 130 the server uses the decrypted account number to look up the identity of the sender and sends the identity back to the recipient computer.

Figure 7 represents an embodiment of an example manual authentication process because of the use of the box and the drag and drop user interface to

drag the icon or stemp to the box. A more generic process for manual authentication is symbolized by the example flowchart of Figure 8. Step 132 represents the process of receiving in the receiver client computer 10 a transaction number. This transaction number is sent from the server 8 in Figure 1 which is referred to as the secure micropayments server or the protected email server. This transaction number is sent to the recipient computer 10 in communication 18 in Figure 1 after the sender computer has made transmission 14 in Figure 1 and received back communication 16 there and after the server has generated the transaction number as a result of communication 14.

Step 134 represents the process of receiving from a sender client computer 2 in Figure 1 a protected email. This email corresponds to message path 5 in Figure 1. This protected email is sent after the sender client computer performs any of the sending protocols earlier described with the server 8 to request a stemp or a code which indicates the server is on the recipient's white list and the server sends back the stemp or code and generates a transaction number which is sent to the recipient computer in the communication labelled 18 in Figure 1.

Step 136 represents the process of displaying the protected email on the recipient computer with some indication of who it purports to be from. In an embodiment, the recipient computer may receive a custom indication or generic logo with the transaction number, and it is this custom indication or generic logo that is displayed along with the name of the sender.

Step 138 represents the process of displaying on the recipient computer some user interface mechanism that the user can invoke to select a particular email and launch the authentication process. It is this manual step which is why this process is called the manual authentication process although most of the work is done by computers. The particular user interface mechanism selected is not important. It can be a single mouse click on a particular email or on the logo of that email followed by selection of a drop down menu command, entry of a hot key combination, or selection of a radio button. Whatever it is, two things must be accomplished, and any mechanism to do these two things will suffice: 1) selection of a particular email; and 2) giving some indication that authentication of its source is desired.

Step 140 represents the process of responding to selection of a particular email and giving of a command to authenticate the email. The recipient computer responds by sending a message to the secure micropayment server requesting authentication of a protected email message identified by information sent in the authentication request message. The authentication request message includes the encrypted micropayments account number from the header of the protected email and the transaction number received from the secure micropayments server. The recipient computer keeps a map or table of which emails correspond to which transaction numbers, and, as a general measure of security, matches the transaction number received from the secure micropayments server with the transaction number encrypted in the custom indication or logo of the protected email.

In step 142, the secure micropayments server uses the transaction number to look up the encryption key used to encrypt the secure micropayments account number when the sender computer first requested a stemp and carried out the sending protocol with the secure micropayments server. That encryption key was used to encrypt the account number and transaction number that was embodied in the stemp, custom indication or generic logo sent back to the sender computer after verification that the sender had an account with sufficient money to pay the cost of the stemp or was on the white list.

In step 144, the encryption key is used to decrypt the account number received from the receiver computer with the authentication request message. If the micropayment account number does not decrypt properly, sending a message to the recipient computer that the selected email is not from who it purports to be from. If the decryption succeeds, proceeding to step 146.

In step 146, the decrypted account number is used to look up the identity of the sender of the message to be authenticated, and a message is sent back to the recipient computer with the identity of the owner of the account.

In step 148, the recipient computer compares the identity received from the secure micropayments server with the identity of the sender who purports to have sent the protected email and determines if there is a discrepancy and warns the user if there is. In some embodiments, the actual sender ID is displayed, and

in other embodiments, a message is displayed indicating whether the sender is or is not authentic.

Figure 9 is a flowchart of an example embodiment of a protocol to send and receive protected email. In step 150, the sender software receives a command to send email to a particular, identified recipient. In step 152, the sender software receives keystrokes or other data to compose the email message, identify the intended recipient and supply identifying information that identifies the sender. In the embodiment, the identifying information is a user name and password (encrypted) and some other secure information that only the sender would know and/or to which a phisher would not have access. This other information can be a key that changes over time, a fixed, secret key, or any other secret information only the authentic sender would know such as the micropayments account number, a birthplace or time of birth.

Step 154 represents the process where the sender software sends to the server an initiation message including any secure identification information such as user name and password and other secure information such as an authentication key or other secret information only the authentic sender possesses. The initiation message requests a stemp, and includes the header of the proposed email which includes information identifying the proposed recipient.

Step 156 represents the process of the server using the identifying information in the initiation message to authenticate the identity of the sender to verify that the initiation message is from who it purports to be from. This may be done by using the identifying information to look up the account number and then looking up the identity of the client which owns said account number. This identity of the owner of the account is compared to the purported identity of the sender, and, if there is a match, the sender's identity is authenticated. If there is no match, a warning message is generated and sent. It may be sent to the recipient computer, or it may be sent back to the sender computer. In either event, the request for a stemp is denied, all as symbolized by steps 158 and 160. These steps thwart phishers because the phisher will not be in possession of the user name, password and other secret authenticating information that only the authentic senders possess.

Test 162 happens if the sender is authenticated in step 158. If the sender's identity is authenticated, the server (8 in Figure 1) determines if the request for a stemp in the initiation message (14 in Figure 1) is within policy limits of a stemp policy. The stemp policy can be anything, but, in general, it is a policy to thwart high volume spammers such as a limit on the number of stemps that can be issued to one sender in one 24 hour period, or a limit on the dollar volume of stemps that can be issued to one sender in any 24 hour period. If the request is not within the policy limits, step 164 is performed to deny the stemp request and processing returns to step 150. This part of the process may defeat any bulk spammer who decides to make the micropayments to send out the spam because the stemp policy will have limits set to bar this type of activity.

If the request for the stemp is within the stemp policy, step 166 is performed. In this step, the server uses an encryption key and encrypts unique information which directly or indirectly gives the identity of the sender into a stemp. The server also encrypts some information from the header of the proposed email message (or from the main body of the message in some embodiments where the main body or some portion thereof is sent with the initiation message) into the stemp so as to tie the stemp to the message. The identifying information that identifies the sender can be unique to this stemp or it can be unique to the stemp when considered in the context of other information such as the date, the account number, etc. In an embodiment, the important thing is that whatever this unique information is that is encoded into the stemp, the unique information directly or indirectly gives the identity of the sender who requested the stemp and that it or the sender's identity can be mapped to a custom indication, generic logo or white list source icon associated with this particular sender.

For example, the identifying information may be a transaction number which is then stored in a table in the server along with the identity of the sender, the encryption key used and the encrypted version of the stemp and any custom indication, generic logo or white list source icon associated with this particular sender. Figure 10 is an example of such a table or database which can be used by the server to map from a transaction number to an identity. Row 165 stores a record for one protected email sent by Fed Ex. Column 167, row 165 stores the

transaction number encrypted into a stemp, and column 169, row 165 stores the encryption key used to encrypt the stemp.

In some embodiments, the stemp is multiple layers with a standard key known to the server used to encrypt the outer layer and a unique session key used to encrypt certain critical information at the next layer such as the sender ID, the sender ID'S account number or custom indication, etc. In other words, the stemp may comprise a piece of unique information such as a transaction number and sender ID or sender account number data encrypted with a unique session key. The combination of the transaction number and the encrypted sensitive information is then encrypted with the outer layer standard key. In such a case, Column 169 stores the unique session key used to encrypt the sensitive information within the stemp that directly or indirectly points to the sender's identity.

Column 171, row 165 stores the sender ID (in this example case Fed Ex) or the sender's account number or a pointer to the ID of the sender. In some embodiments, this column is omitted, and the sender identity must be looked up in a separate database or table after the transaction number is decrypted from the stemp. Column 173, row 165 stores the encrypted version of the stemp. Column 173, row 165 allows the server to map from encrypted version of a stemp to the key that was used to encrypt it so that the key can be used to decrypt the stemp, retrieve the transaction number and look up the identity of the sender.

In embodiments where some information from the main body is encrypted into the stemp, the method is to follow some predetermined protocol to which all the recipient computers know without being told to extract some information from the main body of the message or develop a hash of all or some of the main body of the message for encryption into the stemp. For example, a predetermined hash function of the recipient email address and the first 10 words of the email main body might be used.

Returning to step 166, Figure 10 is an example of how the transaction number encrypted or other identifying information encrypted in the stemp and stored in column 167 can, for any particular protected email (row in Figure 10) indirectly lead to the identity of the sender. After the stemp is encrypted, the

transaction number or other identifying information encrypted into the stemp is stored in column 167 of a row of the table or in a corresponding field of a database record represented by Figure 10. The encryption key is then stored in column 169 of the same row (or a corresponding field of a database record), and the identity of the sender (or his account number or a pointer to his identity) is stored in column 171 of the same row (or a corresponding field of a database record). The encrypted version of the stemp is stored in column 173 of the same row (or a corresponding field of a database record pertaining to this particular protected email). The identity of the sender stored in column 171 is also mapped (often in another table stored in the server) to an associated source icon so that once the server determines the identity of the sender, it can immediately determine whether that sender has paid to have a custom indication source icon displayed with its email in the segregated inbox on the recipient computer or the email is to be displayed with a white list source icon or a generic logo. Column 175 stores the identity of or a pointer to the particular source icon to be displayed with the protected email on the recipient computer's display of the contents of the inbox. In the case of row 165, Fed Ex has paid to have its Fed Ex icon displayed in the inbox with its email for better brand identification. Accordingly, row 165, column 175 stores the identity of the Fed Ex custom indication or a pointer to its storage location. In the case of row 177, this protected email is from a white list member of the intended recipient. Accordingly, column 175 stores a pointer to or the identity of the white list source icon for display with the email from sender Bobbie White on the recipient computer's display of the contents of the inbox.

The server then checks to determine if the sender is on a white list of the intended recipient, and, if so sends the encrypted stemp to the sender without deducting a micropayment amount from the sender's account. If the sender is not on a white list of the intended recipient, the server deducts the micropayment amount from the sender's account and sends the encrypted stemp back to the sender.

The stemp is sent back to the sender in step 166 along with information that ties this particular stemp to this particular email message. Typically, this is done by encrypting part of the original header sent to the server into the stemp so

as to tie the stemp to the original message for which it was generated. The server then sends the whole header back to the sender along with the stemp. This may prevent the theft of the stemp and use on a false message from a phisher.

In step 168, the sender software attaches the header with the encrypted stemp received from the server to the email and sends the email through regular channels to the recipient.

In step 170, the recipient computer's receiver software 12 receives the email from the sender and examines the header of the email for the presence of a stemp. If test 172 determines that no stemp is present, the email is directed in step 171 to the regular non protected email client process on the recipient computer for display in the regular, non segregated inbox. Examples of such non protected email client processes or applications are Netscape Communicator's Messenger module, Outlook Express and many other email clients.

In step 174, the recipient computer's receiver software process attempts to validate the stemp. This is done by sending the header of the received message back to the server. This step is done automatically in some embodiments, or manually by giving the validation command after selecting a particular email in an embodiment. In some embodiments some predetermined portion of the main body of the received message is also sent back to the server or a predetermined hash thereof is sent back to the server. The purpose of sending some information from the main body back may be to provide information the server needs to verify that the stemp in the header was attached to the original message for which it was requested and not diverted to some other email message. Therefore, the part of the main body sent back to the server with the request to validate the stemp needs to be the same part which the server originally encrypted into the stemp in the first place. Not all embodiments require part of the main body to be sent back with the header. In some embodiments, the recipient computer can figure out the key needed to decrypt the stemp and decrypt the stemp and validate it without sending the stemp back to the server. In such the embodiments, the same key may be used to encrypt every stemp, or some predetermined algorithm known to both the recipient

computer and the server may be used to generate the key from some information in the header and/or main body of the message itself.

The process to validate the stemp may be an attempt by the receiver software process to answer three questions: 1) is the stemp a valid, paid for stemp? 2) is the sender of this email who he claims to be? and 3) is there any associated logo, jingle or other information to be displayed/played to the user which maps to this message?

In step 176, the server receives the validation request and uses the encrypted version of the stemp or the decrypted transaction number to search a database or table such as that shown in Figure 10 to find the unique session key used to encrypt the sender ID information encrypted into the stemp. In the case of a single level stemp, with one unique session key used to encrypt all the data in the stemp, step 176 represents using the encrypted version of the stemp to search the table or database to find the record which pertains to this particular protected email. That record can then be searched for the rest of the information the sender needs. In the case of a multiple level stemp, the server uses the standard key to decrypte the first layer of the stemp to extract the transaction number. This transaction number is then used to search the database or table to find the record that pertains to this particular email. That record is then searched to

In step 178, this key is used to decrypt the stemp to recover the unique information (hereafter referred to as the transaction number) encrypted into the stemp that directly or indirectly gives the identity of the sender.

In step 180, the unique information such as the transaction number which is decrypted from the stemp is used to look up the identity of the sender who originally requested the stemp. The transaction number is also used to look up the properties of the stemp such as whether it is a valid, paid for stemp and whether there is a custom indication, generic logo, jingle or white list logo that is associated with this transaction number. This can be determined by checking the appropriate row and column 175 of the table in the server and represented by Figure 10.

In step 182, the server determines whether the sender ID information derived by decrypting the stemp matches the ID of the sender who sent the

message the recipient computer received. In some embodiments, the server also decrypts linking information from the stemp, and determines if the linking information from the header and/or main body which was encrypted into the stemp and which ties the stemp to a particular message matches the appropriate information from the header and/or main body of the message received by the recipient computer and which caused said validation request message to be sent. If all comparisons indicate a match, the sender is validated, and processing proceeds to step 188 where a message that the sender has been validated is sent back to the recipient computer. This message includes any custom indication, generic logo or white list source icon that needs to be displayed in the inbox for this email or any jingle or soundmark that needs to be played when the email is displayed in the inbox.

If one or more of the comparisons indicates a mismatch, processing proceeds to step 186 where a warning message is sent back to the recipient computer indicating the message is not from who it purports to be from.

In step 190, the recipient computer receives the valid message and puts the protected email in its inbox and displays the appropriate custom indication, generic logo or white list source icon received with the valid message from the server. This custom indication, generic logo or white list source icon is displayed in the inbox with the protected email indicating to the user the source of the email has been validated. If a jingle or soundmark is associated with the protected email, that jingle or soundmark is played to the user when the protected email is first displayed or is selected by the user.

Figure 11 shows a diagrammatic representation of machine in the exemplary form of a computer system 300 within which a set of instructions, for causing the machine to perform any one or more of the methodologies discussed herein, may be executed. In alternative embodiments, the machine operates as a standalone device or may be connected (e.g., networked) to other machines. In a networked deployment, the machine may operate in the capacity of a server or a client machine in server-client network environment, or as a peer machine in a peer-to-peer (or distributed) network environment. The machine may be a personal computer (PC), a tablet PC, a set-top box (STB), a Personal Digital Assistant (PDA), a cellular telephone, a web appliance, a network router, switch

or bridge, or any machine capable of executing a set of instructions (sequential or otherwise) that specify actions to be taken by that machine. Further, while only a single machine is illustrated, the term "machine" shall also be taken to include any collection of machines that individually or jointly execute a set (or multiple sets) of instructions to perform any one or more of the methodologies discussed herein.

The exemplary computer system 300 includes a processor 302 (e.g., a central processing unit (CPU), a graphics processing unit (GPU) or both), a main memory 304 and a static memory 306, which communicate with each other via a bus 308. The computer system 300 may further include a video display unit 310 (e.g., a liquid crystal display (LCD) or a cathode ray tube (CRT)). The computer system 300 also includes an alphanumeric input device 312 (e.g., a keyboard), a user interface (UI) navigation device 314 (e.g., a mouse), a disk drive unit 316, a signal generation device 318 (e.g., a speaker) and a network interface device 320.

The disk drive unit 316 includes a machine-readable medium 322 on which is stored one or more sets of instructions and data structures (e.g., software 324) embodying or utilized by any one or more of the methodologies or functions described herein. The software 324 may also reside, completely or at least partially, within the main memory 304 and/or within the processor 302 during execution thereof by the computer system 300, the main memory 304 and the processor 302 also constituting machine-readable media.

The software 324 may further be transmitted or received over a network 326 via the network interface device 320 utilizing any one of a number of well-known transfer protocols (e.g., HTTP).

While the machine-readable medium 322 is shown in an exemplary embodiment to be a single medium, the term "machine-readable medium" should be taken to include a single medium or multiple media (e.g., a centralized or distributed database, and/or associated caches and servers) that store the one or more sets of instructions. The term "machine-readable medium" shall also be taken to include any medium that is capable of storing, encoding or carrying a set of instructions for execution by the machine and that cause the machine to perform any one or more of the methodologies of the present invention, or that is

capable of storing, encoding or carrying data structures utilized by or associated with such a set of instructions. The term "machine-readable medium" shall accordingly be taken to include, but not be limited to, solid-state memories, optical and magnetic media, and carrier wave signals.

Although the invention has been described in terms of some embodiments disclosed herein, those skilled in the art will appreciate that modifications and improvements may be made without departing from the scope of the invention. All such modifications are intended to be included within the scope of the claims appended hereto.

CLAIMS

What is Claimed is:

1. A machine-implemented method, comprising:

receiving, from a payment service, a transaction number that is associated with a payment of a sender;

acquiring an encrypted account number from a header of an electronic message (email) that is sent from a sender,

requesting, from the payment service, that an identity of the sender be authenticated using the transaction number and the encrypted account number;

receiving a reply from the payment service that authenticates the identity or does not authenticate the identity in response to the transaction number and the encrypted account number.

2. The method of claim 1 further comprising:

receiving a logo representing the identity of the sender from the payment service in connection with the transaction number; and

receiving the logo from the sender with the email.

3. The method of claim 2, wherein requesting further includes accessing a link associated with an interface that contacts the payment service, wherein the interface receives the transaction number and the encrypted account number or the logo.

4. A machine-implemented method, comprising:

decrypting information to verify an identity of a sender of a message, wherein the message is directed to a recipient and a transaction occurs before the recipient acquires the message from the sender;

determining, if the sender is verified, whether an account associated with the decrypted information has a sufficient balance to pay for the transaction, whether the sender has previously paid for the transaction, or whether the sender is permitted to proceed free of charge with the transaction;

acquiring, if the sender is verified and if the payment has not been previously paid and is not free of charge, funds from the account to pay for the transaction; and

encrypting, if the sender is verified, a stemp having an encrypted version of the account number and a transaction number for the message and sending the stemp back to the sender of the message.

5. The method of claim 4 further comprising, sending, if the sender is verified, the transaction number and the stemp to a recipient that is the object of the message from the sender.

6. The method of claim 4, wherein encrypting further includes representing the stemp as at least one of a logo for the sender having the account, a generic stemp for the sender that previously paid for the transaction, and a white list stemp for the sender that is permitted to proceed free of charge.

7. A machine-implemented method, comprising:

presenting protected emails;

receiving a selection of one of the protected emails;

decrypting a logo or stemp to acquire a transaction number for the selection;

acquiring the transaction number to acquire a key to decrypt the logo or stemp further for an account number associated with a sender of the selection; and

authenticating an identity of the sender in response to a decrypted version of the account number.

8. The method of claim 7 further comprising, receiving the logo or stemp as a drag and drop action within an interface associated with a recipient.

9. The method of claim 8 further comprising, sending a response to the recipient to indicate whether the identity is authenticated for the sender associated with the selection.

10. The method of claim 7 further comprising, identifying the logo as being associated with a paid version of the transaction, a generic stemp as being associated with a paid version of the transaction where the identity is masked from a recipient, and a white list stemp as being associated with a free transaction.

11. The method of claim 7, wherein presenting further includes presenting the protected emails within a browser accessible to a recipient.

12. A machine-implemented method, comprising:

receiving an opt out request from a recipient, wherein the request includes an encrypted account number and a transaction number associated with a particular sender that previously sent an electronic mail message (email) to the recipient;

acquiring an identity for the particular sender in response to the encrypted account number and the transaction number; and

adding the identity to an opt-out list associated with the recipient.

13. The method of claim 12, wherein acquiring further includes locating a key to decrypt the account number in response to the transaction number.

14. The method of claim 12 further comprising, blocking a subsequent message sent from the particular sender to the recipient in response to the identity being associated with the opt-out list.

15. A machine-implemented method, comprising:

presenting an interface for viewing and managing electronic mail messages (emails) to a recipient;

displaying a navigation pane and a content pane within the interface, the navigation plane including a paid email icon, a free email icon, and an additional icon for emails not associated with being paid or being free;

activating emails associated with being paid in the content pane if the paid email icon is selected within the navigation pane;

activating emails associated with being free in the content pane if the free email icon is selected within the navigation pane; and

activating emails not associated with being paid or being free in the content pane if the additional icon is selected within the navigation pane.

16. The method of claim 15 further comprising:

presenting within a first section of the content pane a friends tab and a logo mail tab;

activating emails associated with being free in a second section of the content pane if the friends tab is selected; and

activating emails associated with being paid in the second section of the content pane if the logo mail tab is selected.

17. The method of claim 15 further comprising presenting header information for the emails within the content pane when selections are made of the icons within the navigation pane.

18. The method of claim 15 further comprising, splitting the content pane into two or more sections and permitting multiple selections of the icons from the navigation pane to populate appropriate ones of the emails to appropriate ones of the two or more sections.

19. The method of claim 15, wherein accessing the emails associated with being paid in the content pane further includes adding a source icon to a number of the emails, wherein the source icon indicates that a particular email to which it is associated has a sender that has been authenticated.

20. The method of claim 15, wherein accessing the emails associated with being paid in the content pane further includes adding a logo to a number of the emails, wherein the logo indicates that a particular sender associated with that

particular email having the logo has paid extra to have the logo displayed in the content pane.

21. A machine-implemented method, comprising:

receiving a request for a stemp from a sender of a message, wherein the message is directed to a recipient;

authenticating an identity for the sender;

locating an account for the sender;

debiting the account of the sender if dictated by a policy between the sender and the recipient;

generating the stemp as an encrypted version of the account and a transaction number for the message; and

sending the stemp back to the sender.

22. The method of claim 21 further comprising:

receiving a header of the message or the stemp from the recipient;

authenticating the sender for the recipient by decrypting the stemp to identify the account and transaction number; and

notifying the sender that the sender is authentic.

23. The method of claim 21 further comprising:

determining that the sender is on an opt-out list of the recipient; and

revoking the stemp or recalling the stemp from the sender preventing the sender from sending the message to the recipient.

24. The method of claim 21, wherein generating further includes including information in the encrypted stemp that permits the recipient to display and/or route the message in configurable manners when subsequently received from the sender.

25. A machine-implemented method, comprising:

receiving an electronic mail message (email) from a sender;

inspecting a header of the email for a stemp;

requesting authentication of the stemp, if present in the header; and

routing the email to a segregated inbox, if the stemp was present in the header and if authenticated.

26. The method of claim 25 further comprising, routing the email to an unprotected email inbox if the stemp is not present in the header.

27. The method of claim 25 further comprising, receiving a logo or special icon if the stemp is authenticated and using the logo or special icon to display with the email in the segregated inbox.

28. A system, comprising:

a recipient interface to process on a recipient machine; and

a protected electronic mail (email) service to processing on a server machine, wherein the recipient interface is to request authentication of senders of email from the protected email service using headers of the email, and wherein the protected email service is to collect payment from senders and authenticate identities of the senders on behalf of the recipient interface.

29. The system of claim 28 further comprising, a sender interface to interact with the protected email service to acquire a stemp for emails directed to the recipient interface, wherein the stemp identifies an account of the sender and a transaction number associated with a given email, and wherein the protected email service encrypts the stemp in the headers of the email.

30. The system of claim 28, wherein the recipient interface presents a navigation pane and a content pane, the navigation pane includes an icon for paid email, an icon for free email, and an icon for unprotected email, if one of the icons are selected the emails associated with the selected icon are presented in the content pane along with their headers.

31. The system of claim 28, wherein the protected email service is to further maintain an opt-out list on behalf of the recipient interface, wherein the opt-out list identifies senders that are not permitted to send emails to the recipient.

32. The system of claim 28, wherein the recipient interface is to further route emails in response to stemps included in the headers received from the senders into segmented inboxes associated with the recipient interface.

33. The system of claim 28, wherein the protected email service is to further provide stemps to the senders prior to emails being sent from the senders to the recipient interface, and wherein the stemps are associated with paid arrangements where identities of the senders are to be published to the recipient interface, paid arrangements where the identities of the senders are to be concealed from the recipient interface, and free arrangements where the emails are deliverable from the senders to the recipient interface without charge.

34. A machine-accessible medium having instructions embedded thereon, the instructions when accessed by a machine perform a method, comprising:

routing an electronic mail message (email) from a sender to a segmented inbox in response to a stemp included a header of the email;

sending the header to an authentication service to authenticate an identity of the sender; and

displaying or disregarding the email in response to information received from the authentication service.

35. The medium of claim 34 further including instructions for requesting that the authentication service block the sender from sending future emails.

36. The medium of claim 34 further including instructions for configuring how the header and the email are presented in response to the information received from the authentication service.

37. A machine-accessible medium having instructions embedded thereon, the instructions when accessed by a machine perform a method, comprising:

supplying stemps to senders of electronic mail messages (emails) directed to recipients;

authenticating the stemps received in headers of the emails and received from the recipients;

blocking a number of the senders from receiving one of the stemps or revoking a previous one of the stemps if requested to do so by a number of the recipients; and

debiting funds from accounts of the senders if a policy dictates a particular one of the recipients should receive payment, wherein debiting occurs with the supplying of the stemps.

38. The medium of claim 37 further including instructions for generating the stemps as encrypted versions of accounts using identities for the senders and transaction numbers associated with the emails.

39. The medium of claim 37 further including instructions for providing three category of stemps, a paid category where an identity of the senders are published to the recipients, a paid category where the identity is not published to the recipients, and a free category where no payment is required and where the identity is published to the recipients.

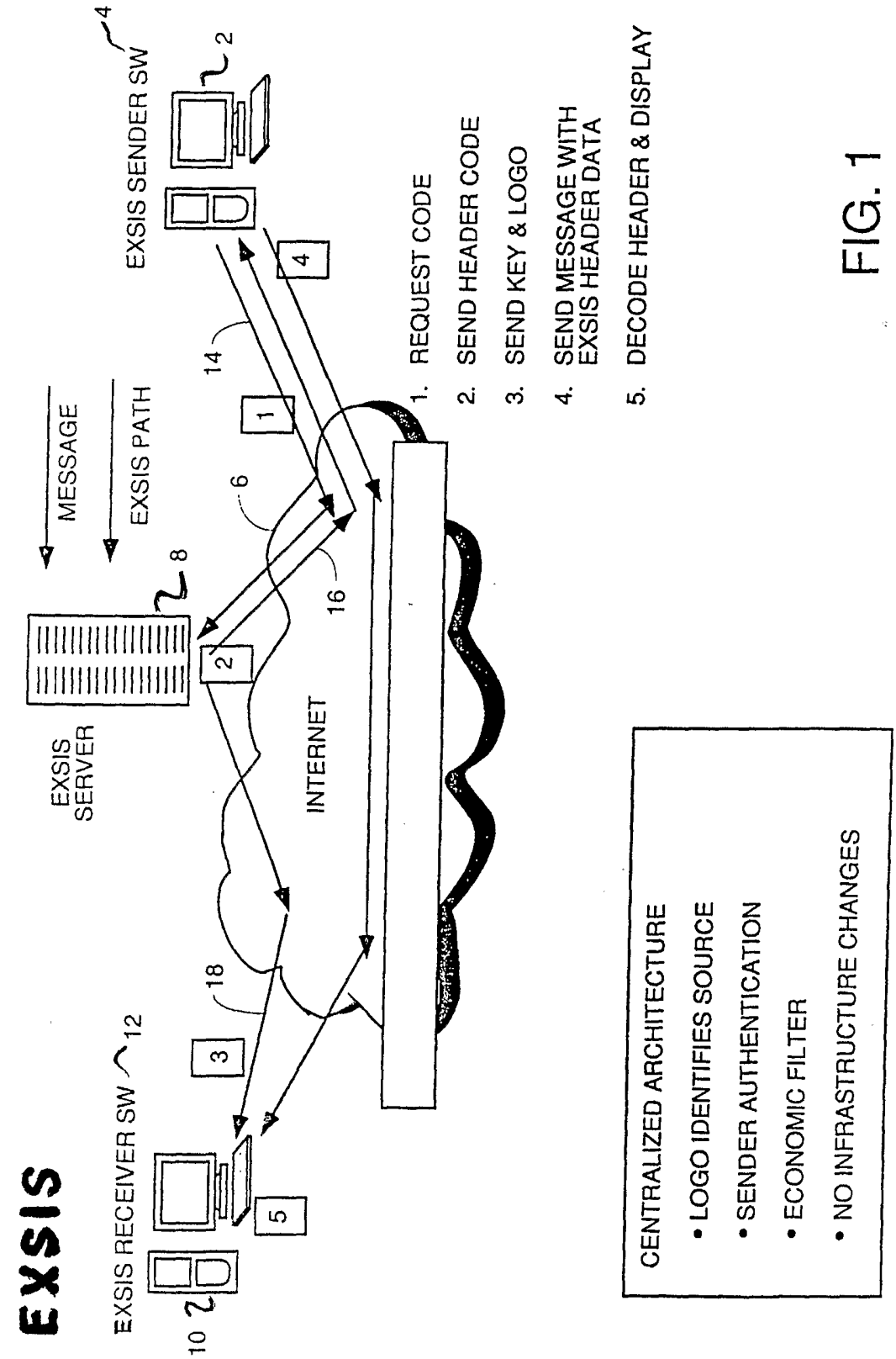

EXSIS
EXSIS RECEIVER SW
EXSIS SENDER SW
EXSIS SERVER
INTERNET
MESSAGE
EXSIS PATH
1. REQUEST CODE
2. SEND HEADER CODE
3. SEND KEY & LOGO
4. SEND MESSAGE WITH EXSIS HEADER DATA
5. DECODE HEADER & DISPLAY
CENTRALIZED ARCHITECTURE
• LOGO IDENTIFIES SOURCE
• SENDER AUTHENTICATION
• ECONOMIC FILTER
• NO INFRASTRUCTURE CHANGES

FIG. 1

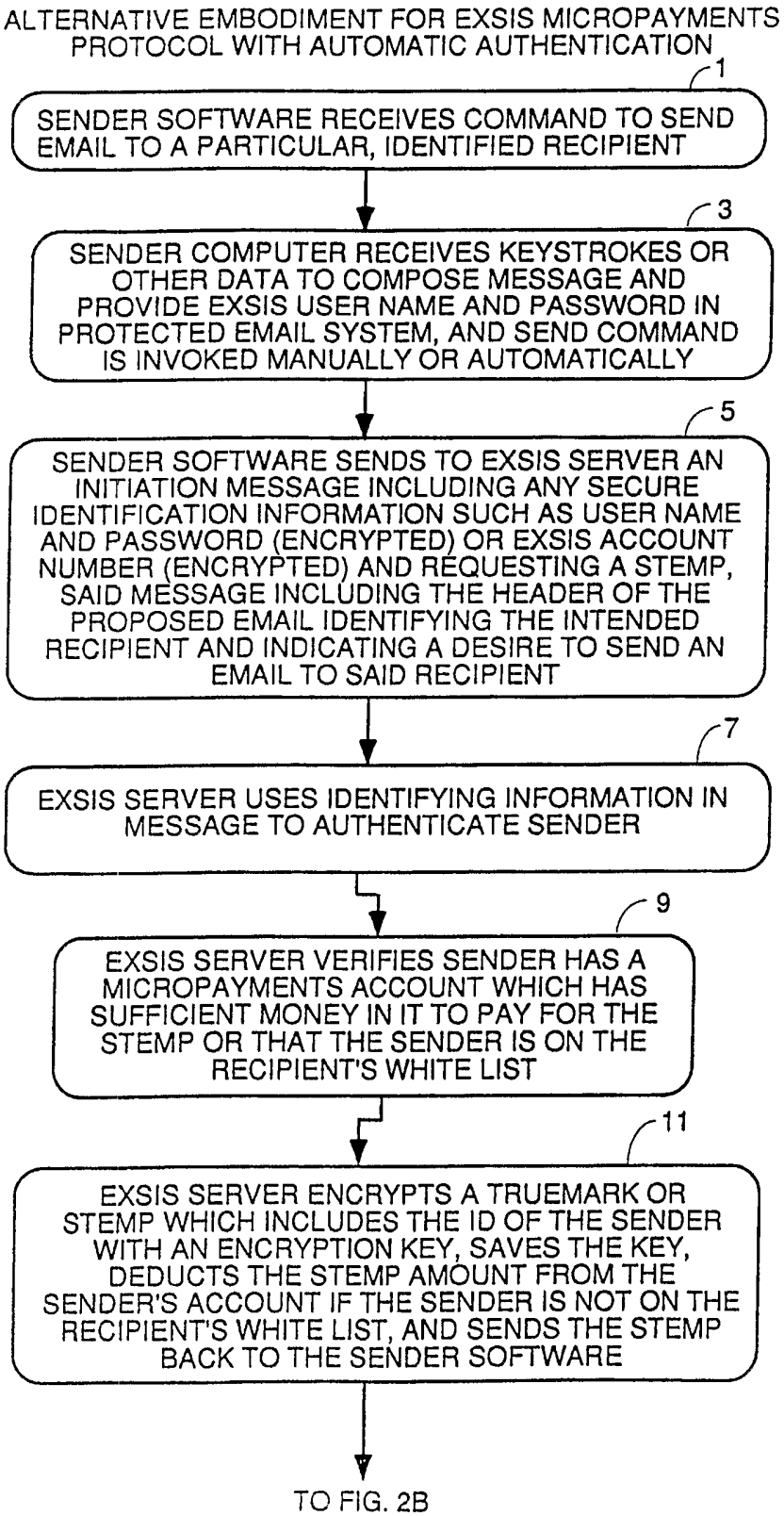

ALTERNATIVE EMBODIMENT FOR EXSIS MICROPAYMENTS PROTOCOL WITH AUTOMATIC AUTHENTICATION
1
SENDER SOFTWARE RECEIVES COMMAND TO SEND EMAIL TO A PARTICULAR, IDENTIFIED RECIPIENT
3
SENDER COMPUTER RECEIVES KEYSTROKES OR OTHER DATA TO COMPOSE MESSAGE AND PROVIDE EXSIS USER NAME AND PASSWORD IN PROTECTED EMAIL SYSTEM, AND SEND COMMAND IS INVOKED MANUALLY OR AUTOMATICALLY
5
SENDER SOFTWARE SENDS TO EXSIS SERVER AN INITIATION MESSAGE INCLUDING ANY SECURE IDENTIFICATION INFORMATION SUCH AS USER NAME AND PASSWORD (ENCRYPTED) OR EXSIS ACCOUNT NUMBER (ENCRYPTED) AND REQUESTING A STEMP, SAID MESSAGE INCLUDING THE HEADER OF THE PROPOSED EMAIL IDENTIFYING THE INTENDED RECIPIENT AND INDICATING A DESIRE TO SEND AN EMAIL TO SAID RECIPIENT
7
EXSIS SERVER USES IDENTIFYING INFORMATION IN MESSAGE TO AUTHENTICATE SENDER
9
EXSIS SERVER VERIFIES SENDER HAS A MICROPAYMENTS ACCOUNT WHICH HAS SUFFICIENT MONEY IN IT TO PAY FOR THE STEMP OR THAT THE SENDER IS ON THE RECIPIENT'S WHITE LIST
11
EXSIS SERVER ENCRYPTS A TRUEMARK OR STEMP WHICH INCLUDES THE ID OF THE SENDER WITH AN ENCRYPTION KEY, SAVES THE KEY, DEDUCTS THE STEMP AMOUNT FROM THE SENDER'S ACCOUNT IF THE SENDER IS NOT ON THE RECIPIENT'S WHITE LIST, AND SENDS THE STEMP BACK TO THE SENDER SOFTWARE
TO FIG. 2B

FIG. 2A

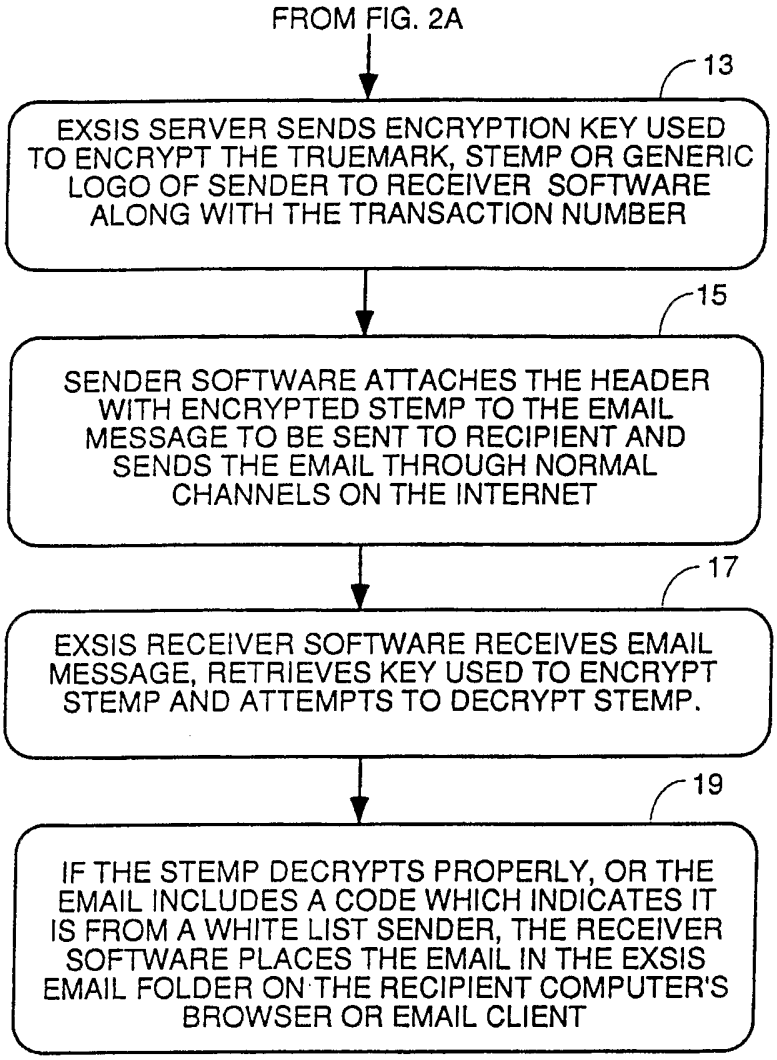

FROM FIG. 2A
13
EXSIS SERVER SENDS ENCRYPTION KEY USED TO ENCRYPT THE TRUEMARK, STEMP OR GENERIC LOGO OF SENDER TO RECEIVER SOFTWARE ALONG WITH THE TRANSACTION NUMBER
15
SENDER SOFTWARE ATTACHES THE HEADER WITH ENCRYPTED STEMP TO THE EMAIL MESSAGE TO BE SENT TO RECIPIENT AND SENDS THE EMAIL THROUGH NORMAL CHANNELS ON THE INTERNET
17
EXSIS RECEIVER SOFTWARE RECEIVES EMAIL MESSAGE, RETRIEVES KEY USED TO ENCRYPT STEMP AND ATTEMPTS TO DECRYPT STEMP.
19
IF THE STEMP DECRYPTS PROPERLY, OR THE EMAIL INCLUDES A CODE WHICH INDICATES IT IS FROM A WHITE LIST SENDER, THE RECEIVER SOFTWARE PLACES THE EMAIL IN THE EXSIS EMAIL FOLDER ON THE RECIPIENT COMPUTER'S BROWSER OR EMAIL CLIENT

FIG. 2B

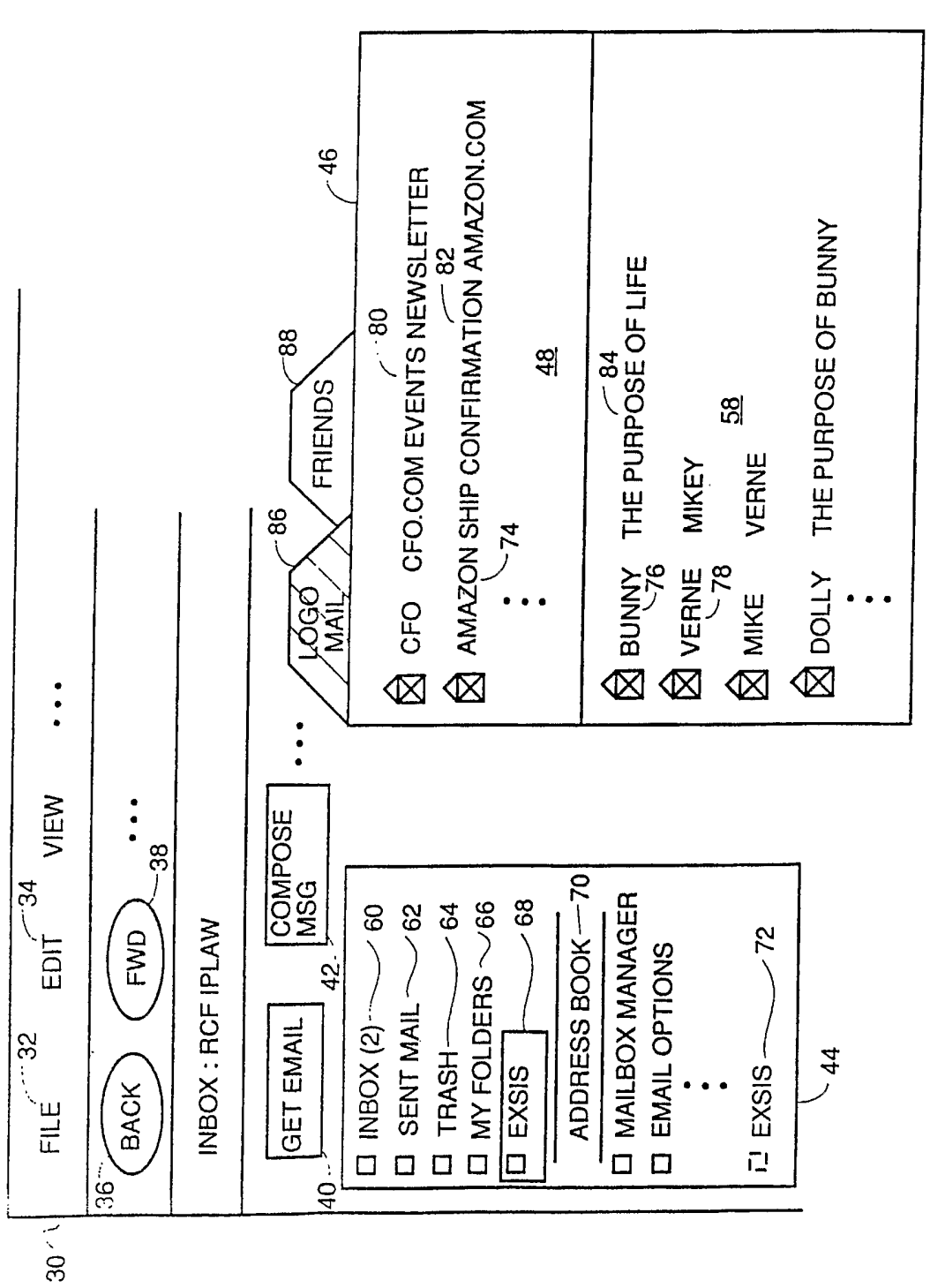

30
FILE
32
EDIT
34
VIEW
BACK
36
FWD
38
INBOX : RCF IPLAW
GET EMAIL
40
COMPOSE MSG
42
INBOX (2)
60
SENT MAIL
62
TRASH
64
MY FOLDERS
66
EXSIS
68
ADDRESS BOOK
70
MAILBOX MANAGER
EMAIL OPTIONS
EXSIS
72
44
LOGO MAIL
86
FRIENDS
88
46
CFO CFO.COM EVENTS NEWSLETTER
80
AMAZON SHIP CONFIRMATION AMAZON.COM
82
74
48
BUNNY THE PURPOSE OF LIFE
84
76
VERNE MIKEY
78
MIKE VERNE
58
DOLLY THE PURPOSE OF BUNNY

FIG. 3

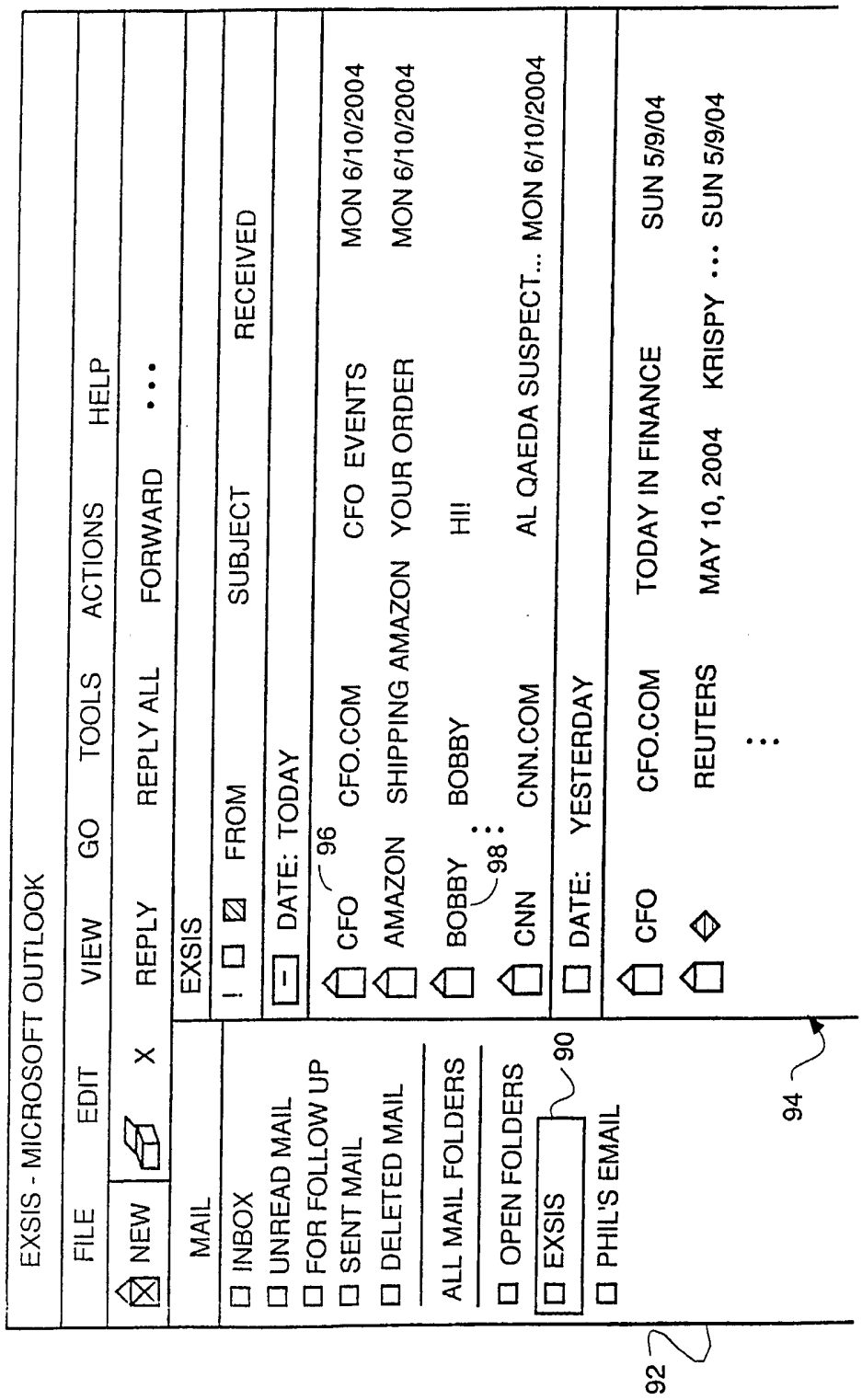

EXSIS - MICROSOFT OUTLOOK
FILE EDIT VIEW GO TOOLS ACTIONS HELP
NEW X REPLY REPLY ALL FORWARD • • •
MAIL
EXSIS
INBOX
UNREAD MAIL
FOR FOLLOW UP
SENT MAIL
DELETED MAIL
ALL MAIL FOLDERS
OPEN FOLDERS
EXSIS
PHIL'S EMAIL
! FROM SUBJECT RECEIVED
DATE: TODAY
CFO CFO.COM CFO EVENTS MON 6/10/2004
AMAZON SHIPPING AMAZON YOUR ORDER MON 6/10/2004
BOBBY BOBBY HI!
CNN CNN.COM AL QAEDA SUSPECT... MON 6/10/2004
DATE: YESTERDAY
CFO CFO.COM TODAY IN FINANCE SUN 5/9/04
REUTERS MAY 10, 2004 KRISPY • • • SUN 5/9/04
90
92
94
96
98

FIG. 4

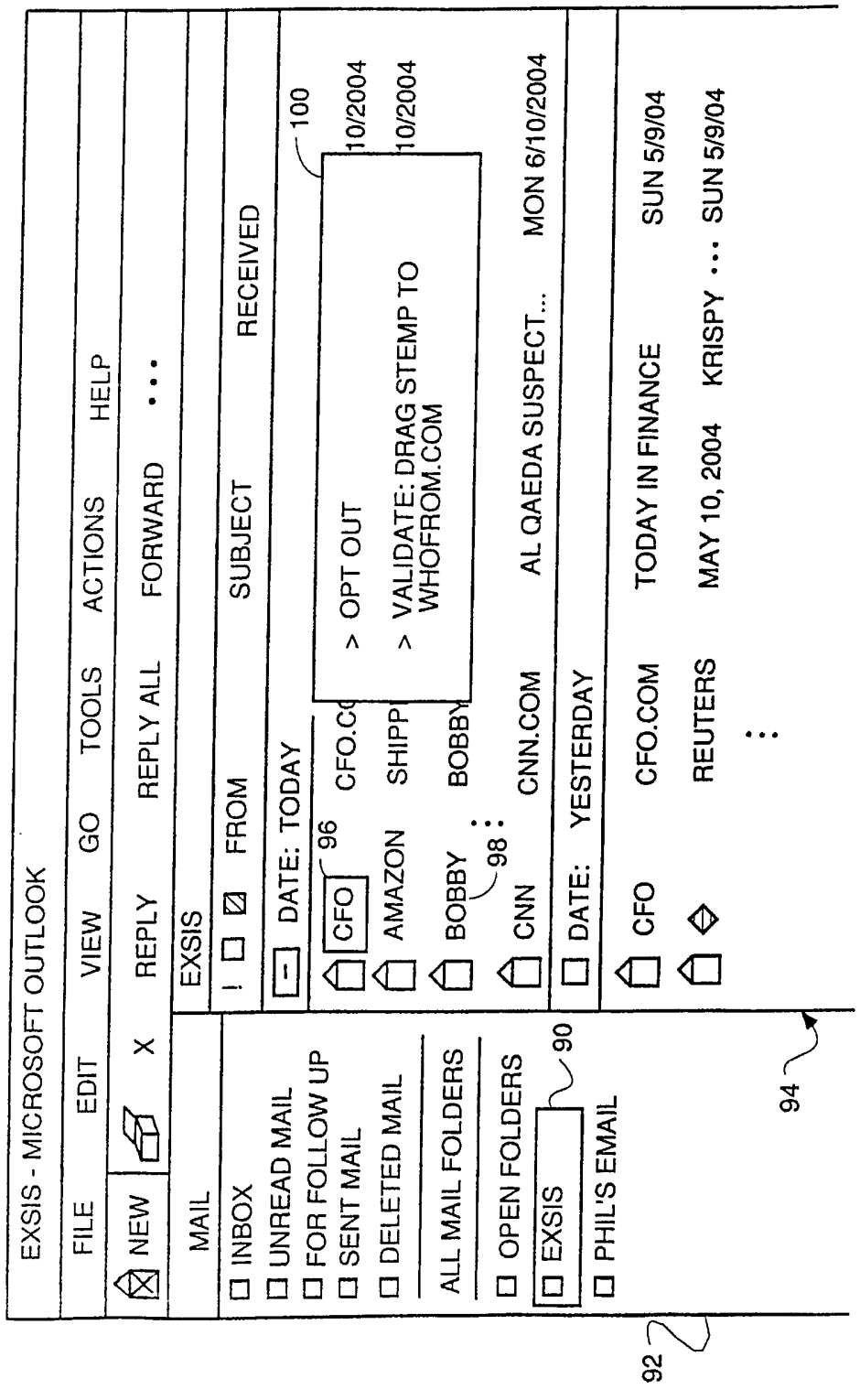

EXSIS - MICROSOFT OUTLOOK
FILE EDIT VIEW GO TOOLS ACTIONS HELP
NEW X REPLY REPLY ALL FORWARD
MAIL
EXSIS
INBOX
UNREAD MAIL
FOR FOLLOW UP
SENT MAIL
DELETED MAIL
ALL MAIL FOLDERS
OPEN FOLDERS
EXSIS
PHIL'S EMAIL
FROM SUBJECT RECEIVED
DATE: TODAY
CFO
AMAZON
BOBBY
CNN
> OPT OUT
> VALIDATE: DRAG STEMP TO WHOFROM.COM
10/2004
10/2004
CNN.COM AL QAEDA SUSPECT... MON 6/10/2004
DATE: YESTERDAY
CFO CFO.COM TODAY IN FINANCE SUN 5/9/04
REUTERS MAY 10, 2004 KRISPY SUN 5/9/04
90
92
94
96
98
100

FIG. 5

OPT OUT PROCESSING BY RECIPIENT COMPUTER, EXSIS SERVER AND SENDER CPU

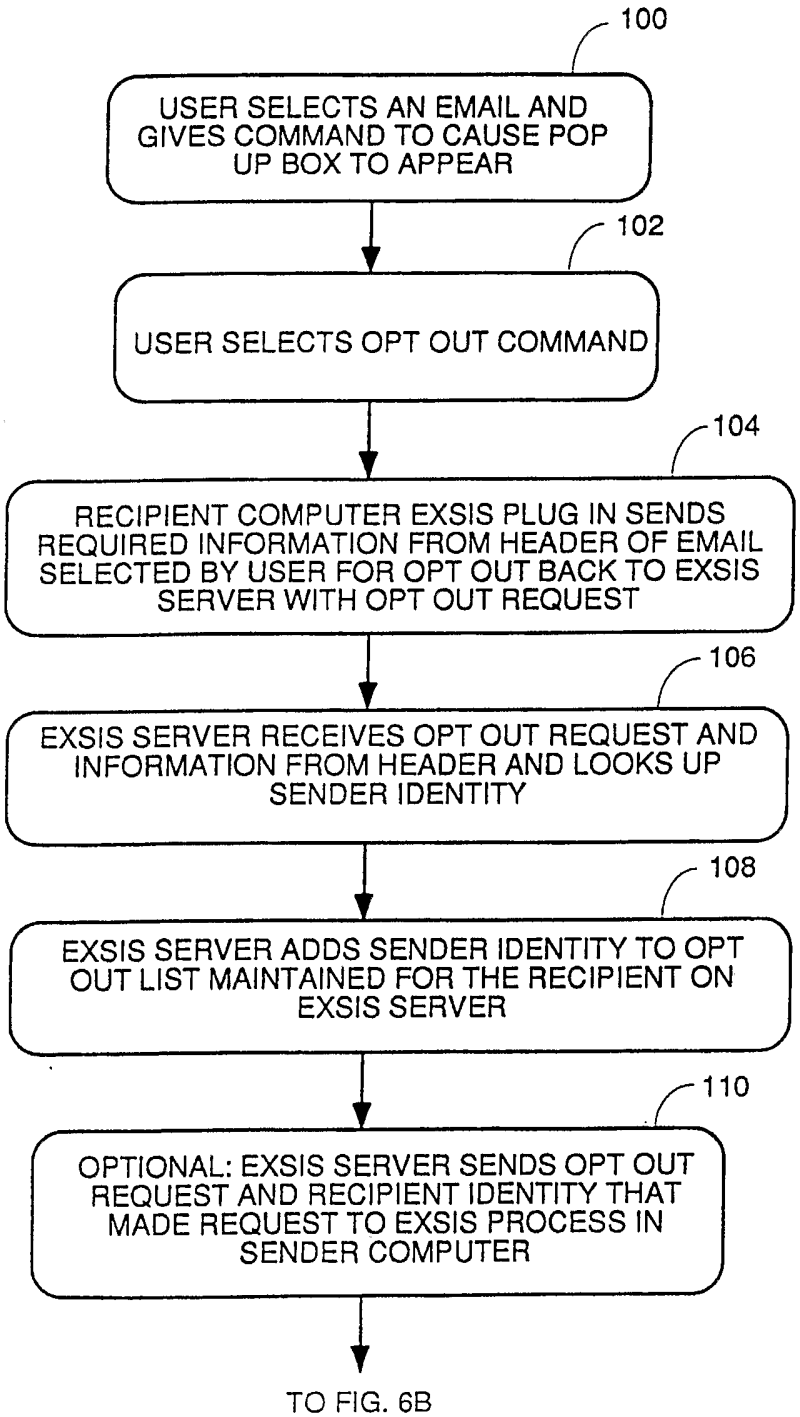


FIG. 6A

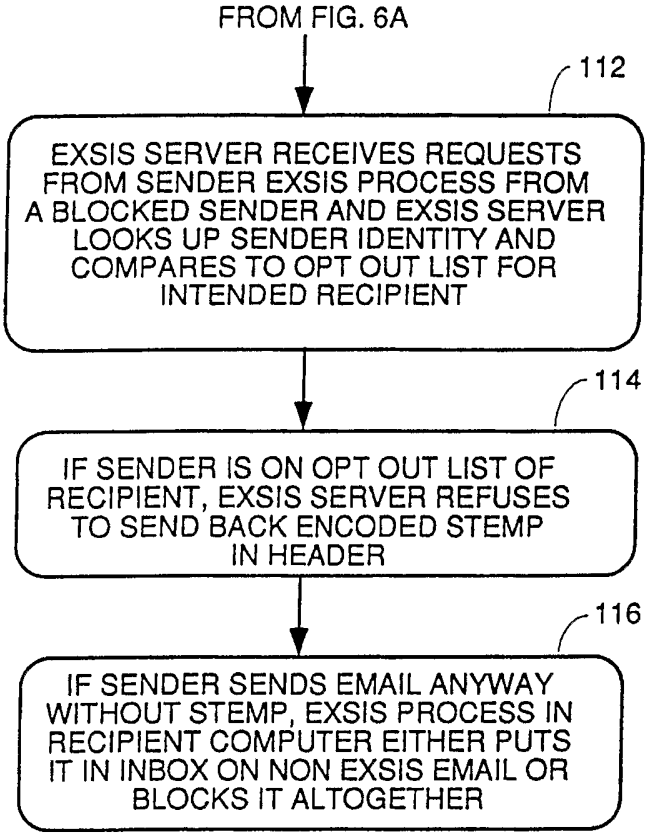

FROM FIG. 6A
112
EXSIS SERVER RECEIVES REQUESTS FROM SENDER EXSIS PROCESS FROM A BLOCKED SENDER AND EXSIS SERVER LOOKS UP SENDER IDENTITY AND COMPARES TO OPT OUT LIST FOR INTENDED RECIPIENT
114
IF SENDER IS ON OPT OUT LIST OF RECIPIENT, EXSIS SERVER REFUSES TO SEND BACK ENCODED STEMP IN HEADER
116
IF SENDER SENDS EMAIL ANYWAY WITHOUT STEMP, EXSIS PROCESS IN RECIPIENT COMPUTER EITHER PUTS IT IN INBOX ON NON EXSIS EMAIL OR BLOCKS IT ALTOGETHER

FIG. 6B

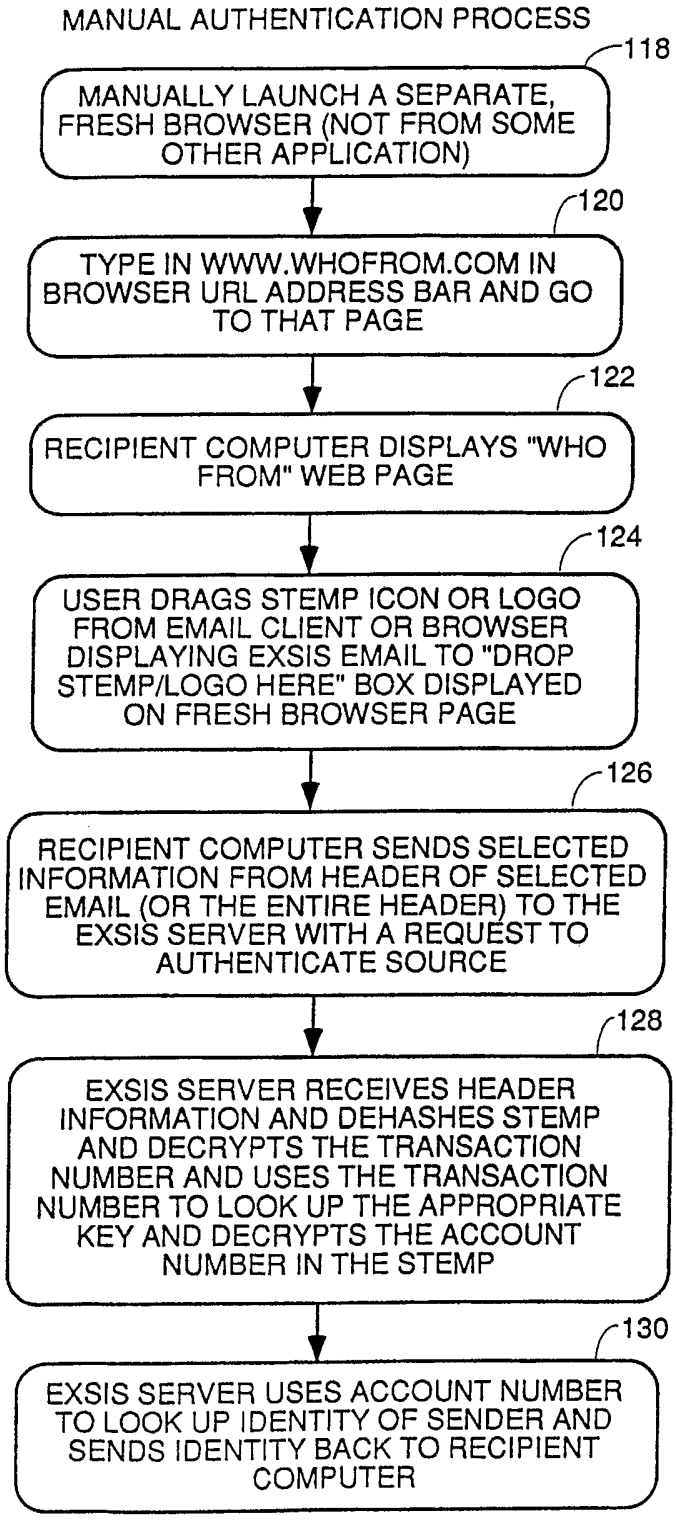

MANUAL AUTHENTICATION PROCESS
118
MANUALLY LAUNCH A SEPARATE, FRESH BROWSER (NOT FROM SOME OTHER APPLICATION)
120
TYPE IN WWW.WHOFROM.COM IN BROWSER URL ADDRESS BAR AND GO TO THAT PAGE
122
RECIPIENT COMPUTER DISPLAYS "WHO FROM" WEB PAGE
124
USER DRAGS STEMP ICON OR LOGO FROM EMAIL CLIENT OR BROWSER DISPLAYING EXSIS EMAIL TO "DROP STEMP/LOGO HERE" BOX DISPLAYED ON FRESH BROWSER PAGE
126
RECIPIENT COMPUTER SENDS SELECTED INFORMATION FROM HEADER OF SELECTED EMAIL (OR THE ENTIRE HEADER) TO THE EXSIS SERVER WITH A REQUEST TO AUTHENTICATE SOURCE
128
EXSIS SERVER RECEIVES HEADER INFORMATION AND DEHASHES STEMP AND DECRYPTS THE TRANSACTION NUMBER AND USES THE TRANSACTION NUMBER TO LOOK UP THE APPROPRIATE KEY AND DECRYPTS THE ACCOUNT NUMBER IN THE STEMP
130
EXSIS SERVER USES ACCOUNT NUMBER TO LOOK UP IDENTITY OF SENDER AND SENDS IDENTITY BACK TO RECIPIENT COMPUTER

FIG. 7

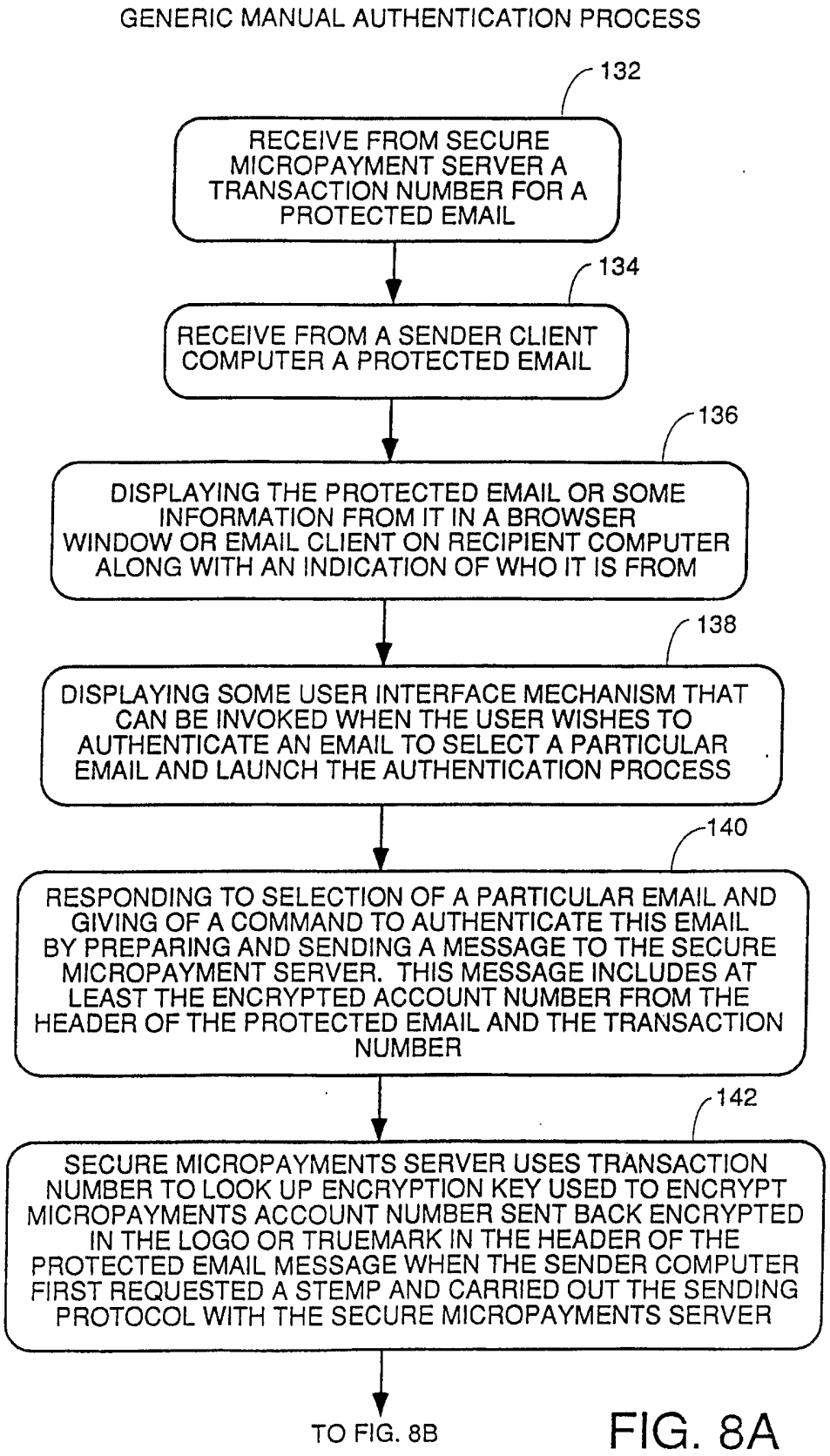

GENERIC MANUAL AUTHENTICATION PROCESS
132
RECEIVE FROM SECURE MICROPAYMENT SERVER A TRANSACTION NUMBER FOR A PROTECTED EMAIL
134
RECEIVE FROM A SENDER CLIENT COMPUTER A PROTECTED EMAIL
136
DISPLAYING THE PROTECTED EMAIL OR SOME INFORMATION FROM IT IN A BROWSER WINDOW OR EMAIL CLIENT ON RECIPIENT COMPUTER ALONG WITH AN INDICATION OF WHO IT IS FROM
138
DISPLAYING SOME USER INTERFACE MECHANISM THAT CAN BE INVOKED WHEN THE USER WISHES TO AUTHENTICATE AN EMAIL TO SELECT A PARTICULAR EMAIL AND LAUNCH THE AUTHENTICATION PROCESS
140
RESPONDING TO SELECTION OF A PARTICULAR EMAIL AND GIVING OF A COMMAND TO AUTHENTICATE THIS EMAIL BY PREPARING AND SENDING A MESSAGE TO THE SECURE MICROPAYMENT SERVER. THIS MESSAGE INCLUDES AT LEAST THE ENCRYPTED ACCOUNT NUMBER FROM THE HEADER OF THE PROTECTED EMAIL AND THE TRANSACTION NUMBER
142
SECURE MICROPAYMENTS SERVER USES TRANSACTION NUMBER TO LOOK UP ENCRYPTION KEY USED TO ENCRYPT MICROPAYMENTS ACCOUNT NUMBER SENT BACK ENCRYPTED IN THE LOGO OR TRUEMARK IN THE HEADER OF THE PROTECTED EMAIL MESSAGE WHEN THE SENDER COMPUTER FIRST REQUESTED A STEMP AND CARRIED OUT THE SENDING PROTOCOL WITH THE SECURE MICROPAYMENTS SERVER
TO FIG. 8B

FIG. 8A

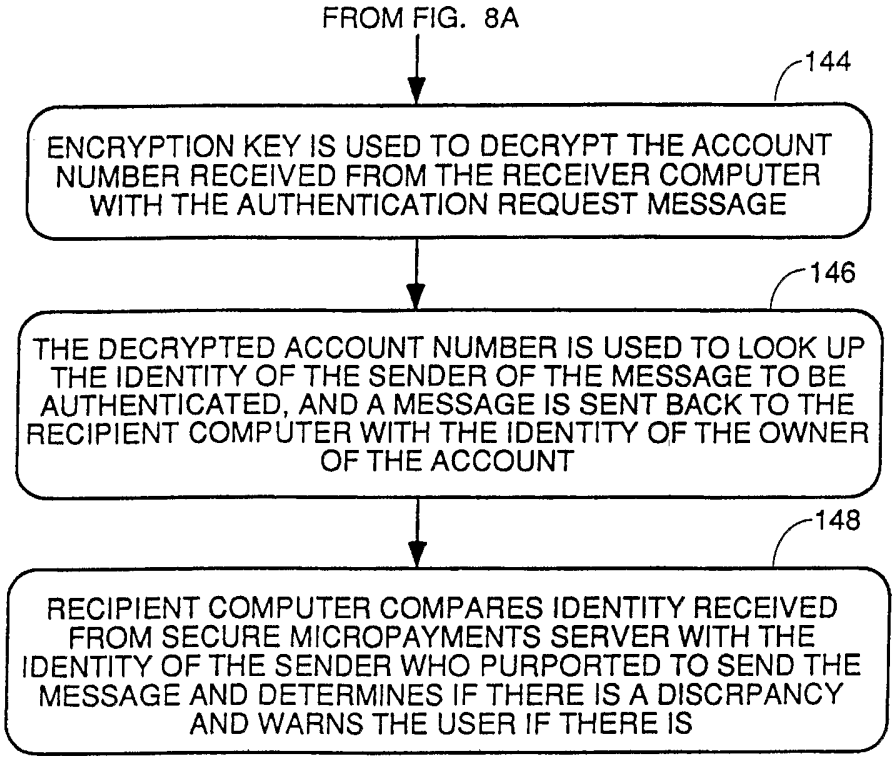

FROM FIG. 8A
144
ENCRYPTION KEY IS USED TO DECRYPT THE ACCOUNT NUMBER RECEIVED FROM THE RECEIVER COMPUTER WITH THE AUTHENTICATION REQUEST MESSAGE
146
THE DECRYPTED ACCOUNT NUMBER IS USED TO LOOK UP THE IDENTITY OF THE SENDER OF THE MESSAGE TO BE AUTHENTICATED, AND A MESSAGE IS SENT BACK TO THE RECIPIENT COMPUTER WITH THE IDENTITY OF THE OWNER OF THE ACCOUNT
148
RECIPIENT COMPUTER COMPARES IDENTITY RECEIVED FROM SECURE MICROPAYMENTS SERVER WITH THE IDENTITY OF THE SENDER WHO PURPORTED TO SEND THE MESSAGE AND DETERMINES IF THERE IS A DISCRPANCY AND WARNS THE USER IF THERE IS

FIG. 8B

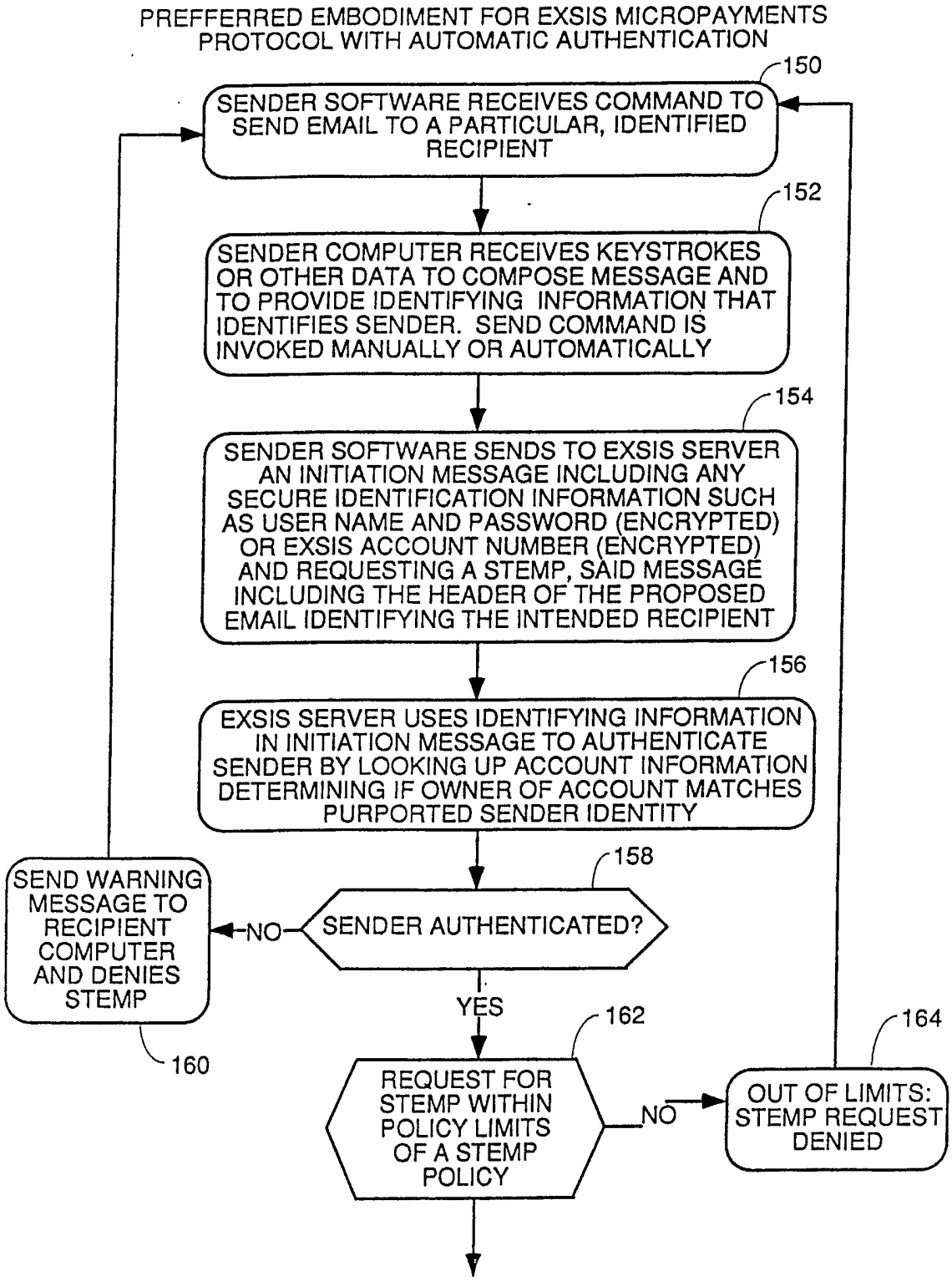

PREFFERRED EMBODIMENT FOR EXSIS MICROPAYMENTS PROTOCOL WITH AUTOMATIC AUTHENTICATION
150
SENDER SOFTWARE RECEIVES COMMAND TO SEND EMAIL TO A PARTICULAR, IDENTIFIED RECIPIENT
152
SENDER COMPUTER RECEIVES KEYSTROKES OR OTHER DATA TO COMPOSE MESSAGE AND TO PROVIDE IDENTIFYING INFORMATION THAT IDENTIFIES SENDER. SEND COMMAND IS INVOKED MANUALLY OR AUTOMATICALLY
154
SENDER SOFTWARE SENDS TO EXSIS SERVER AN INITIATION MESSAGE INCLUDING ANY SECURE IDENTIFICATION INFORMATION SUCH AS USER NAME AND PASSWORD (ENCRYPTED) OR EXSIS ACCOUNT NUMBER (ENCRYPTED) AND REQUESTING A STEMP, SAID MESSAGE INCLUDING THE HEADER OF THE PROPOSED EMAIL IDENTIFYING THE INTENDED RECIPIENT
156
EXSIS SERVER USES IDENTIFYING INFORMATION IN INITIATION MESSAGE TO AUTHENTICATE SENDER BY LOOKING UP ACCOUNT INFORMATION DETERMINING IF OWNER OF ACCOUNT MATCHES PURPORTED SENDER IDENTITY
158
SENDER AUTHENTICATED?
NO
SEND WARNING MESSAGE TO RECIPIENT COMPUTER AND DENIES STEMP
160
YES
162
REQUEST FOR STEMP WITHIN POLICY LIMITS OF A STEMP POLICY
NO
164
OUT OF LIMITS: STEMP REQUEST DENIED

FIG. 9A

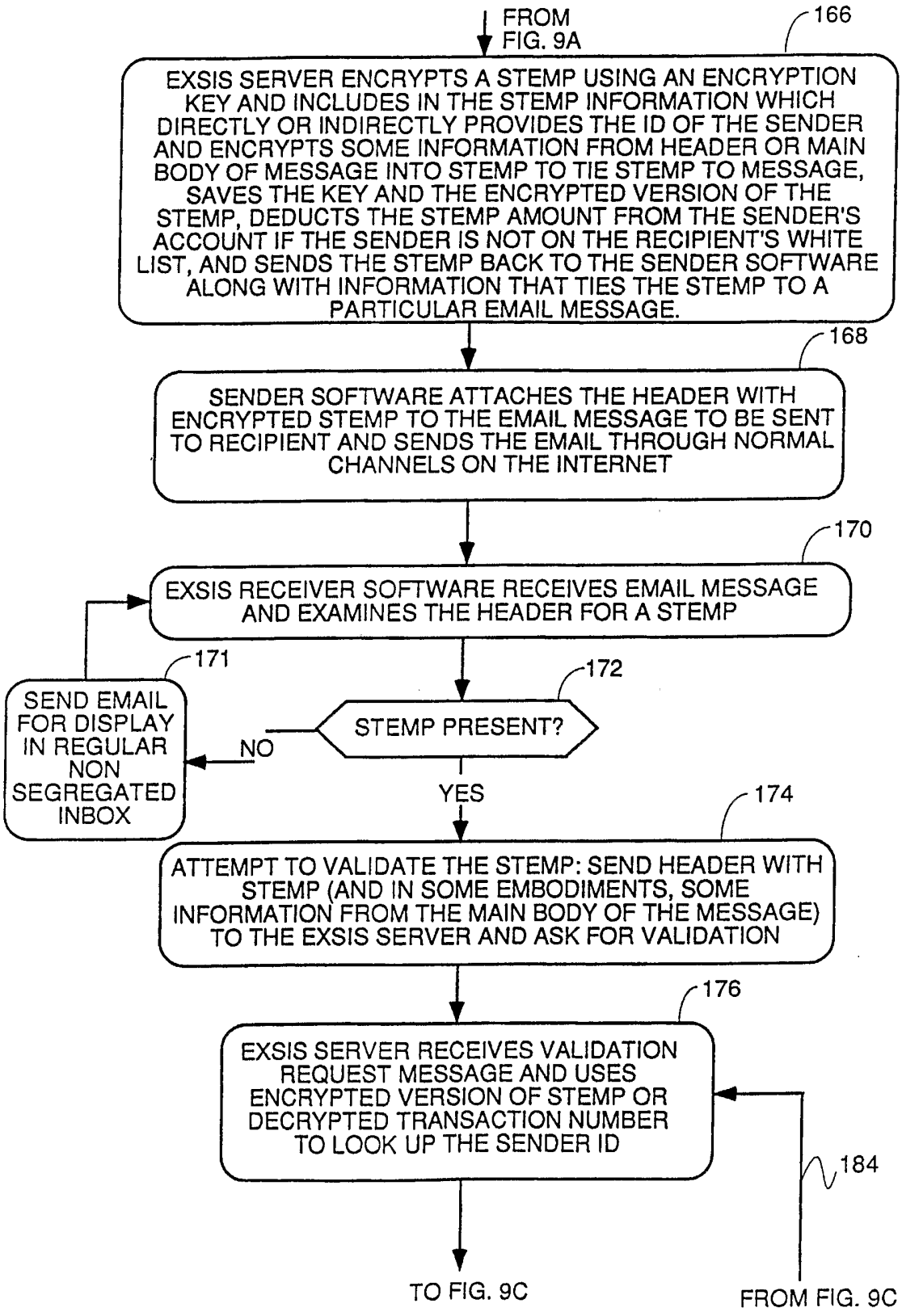

FROM FIG. 9A
166
EXSIS SERVER ENCRYPTS A STEMP USING AN ENCRYPTION KEY AND INCLUDES IN THE STEMP INFORMATION WHICH DIRECTLY OR INDIRECTLY PROVIDES THE ID OF THE SENDER AND ENCRYPTS SOME INFORMATION FROM HEADER OR MAIN BODY OF MESSAGE INTO STEMP TO TIE STEMP TO MESSAGE, SAVES THE KEY AND THE ENCRYPTED VERSION OF THE STEMP, DEDUCTS THE STEMP AMOUNT FROM THE SENDER'S ACCOUNT IF THE SENDER IS NOT ON THE RECIPIENT'S WHITE LIST, AND SENDS THE STEMP BACK TO THE SENDER SOFTWARE ALONG WITH INFORMATION THAT TIES THE STEMP TO A PARTICULAR EMAIL MESSAGE.
168
SENDER SOFTWARE ATTACHES THE HEADER WITH ENCRYPTED STEMP TO THE EMAIL MESSAGE TO BE SENT TO RECIPIENT AND SENDS THE EMAIL THROUGH NORMAL CHANNELS ON THE INTERNET
170
EXSIS RECEIVER SOFTWARE RECEIVES EMAIL MESSAGE AND EXAMINES THE HEADER FOR A STEMP
171
SEND EMAIL FOR DISPLAY IN REGULAR NON SEGREGATED INBOX
172
STEMP PRESENT?
NO
YES
174
ATTEMPT TO VALIDATE THE STEMP: SEND HEADER WITH STEMP (AND IN SOME EMBODIMENTS, SOME INFORMATION FROM THE MAIN BODY OF THE MESSAGE) TO THE EXSIS SERVER AND ASK FOR VALIDATION
176
EXSIS SERVER RECEIVES VALIDATION REQUEST MESSAGE AND USES ENCRYPTED VERSION OF STEMP OR DECRYPTED TRANSACTION NUMBER TO LOOK UP THE SENDER ID
184
TO FIG. 9C
FROM FIG. 9C

FIG. 9B

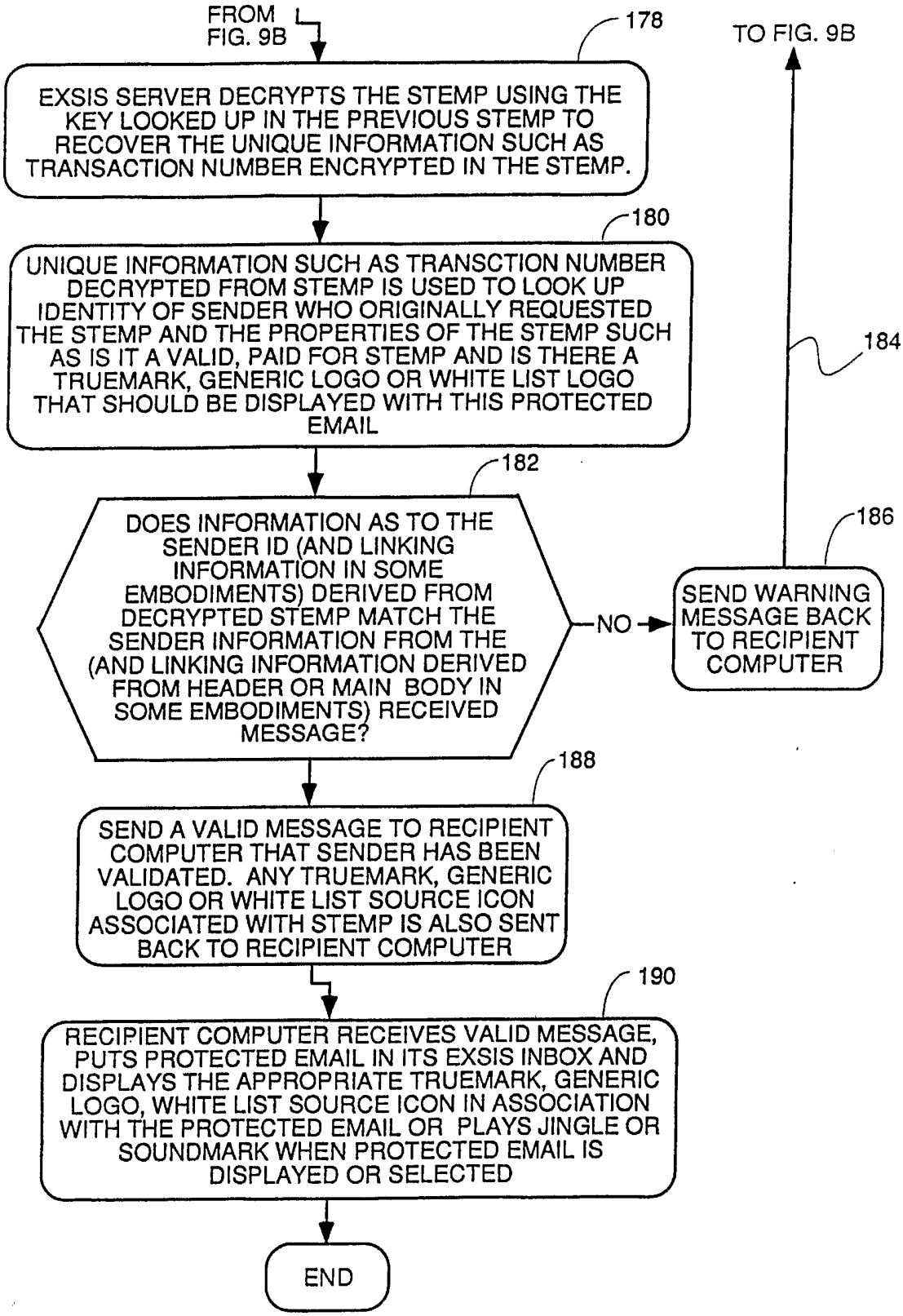

FROM FIG. 9B
178
EXSIS SERVER DECRYPTS THE STEMP USING THE KEY LOOKED UP IN THE PREVIOUS STEMP TO RECOVER THE UNIQUE INFORMATION SUCH AS TRANSACTION NUMBER ENCRYPTED IN THE STEMP.
180
UNIQUE INFORMATION SUCH AS TRANSCTION NUMBER DECRYPTED FROM STEMP IS USED TO LOOK UP IDENTITY OF SENDER WHO ORIGINALLY REQUESTED THE STEMP AND THE PROPERTIES OF THE STEMP SUCH AS IS IT A VALID, PAID FOR STEMP AND IS THERE A TRUEMARK, GENERIC LOGO OR WHITE LIST LOGO THAT SHOULD BE DISPLAYED WITH THIS PROTECTED EMAIL
182
DOES INFORMATION AS TO THE SENDER ID (AND LINKING INFORMATION IN SOME EMBODIMENTS) DERIVED FROM DECRYPTED STEMP MATCH THE SENDER INFORMATION FROM THE (AND LINKING INFORMATION DERIVED FROM HEADER OR MAIN BODY IN SOME EMBODIMENTS) RECEIVED MESSAGE?
NO
SEND WARNING MESSAGE BACK TO RECIPIENT COMPUTER
186
184
TO FIG. 9B
188
SEND A VALID MESSAGE TO RECIPIENT COMPUTER THAT SENDER HAS BEEN VALIDATED. ANY TRUEMARK, GENERIC LOGO OR WHITE LIST SOURCE ICON ASSOCIATED WITH STEMP IS ALSO SENT BACK TO RECIPIENT COMPUTER
190
RECIPIENT COMPUTER RECEIVES VALID MESSAGE, PUTS PROTECTED EMAIL IN ITS EXSIS INBOX AND DISPLAYS THE APPROPRIATE TRUEMARK, GENERIC LOGO, WHITE LIST SOURCE ICON IN ASSOCIATION WITH THE PROTECTED EMAIL OR PLAYS JINGLE OR SOUNDMARK WHEN PROTECTED EMAIL IS DISPLAYED OR SELECTED
END

FIG. 9C

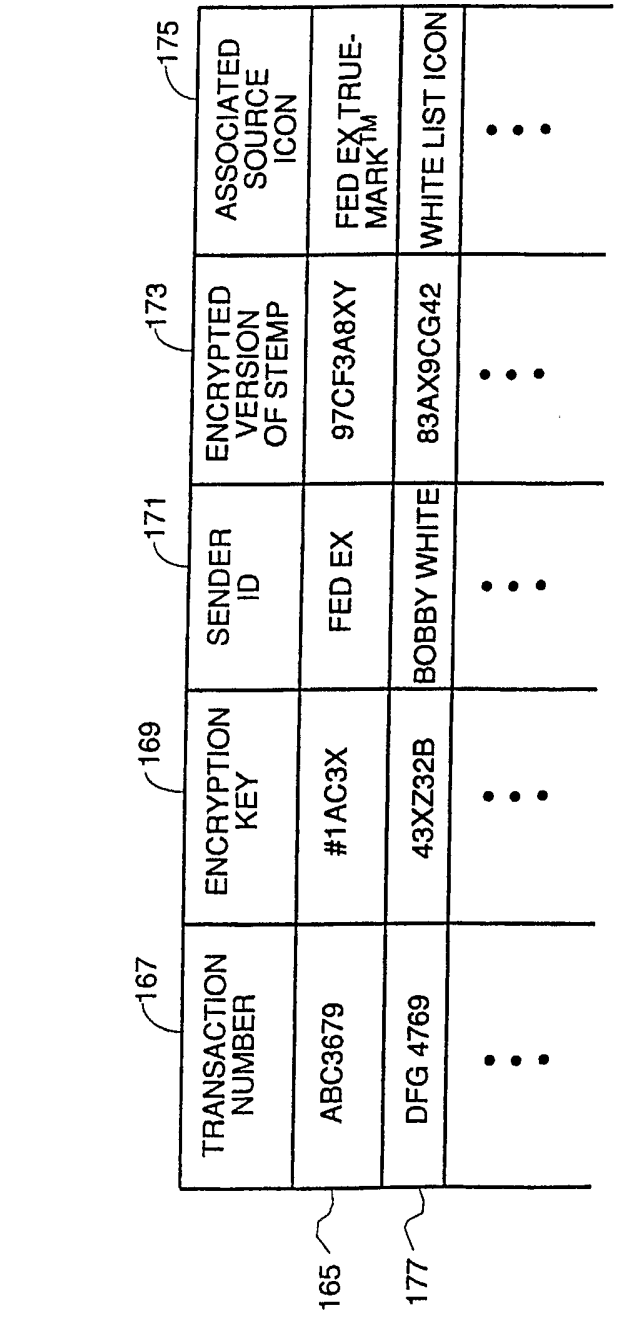

TRANSACTION NUMBER
ENCRYPTION KEY
SENDER ID
ENCRYPTED VERSION OF STEMP
ASSOCIATED SOURCE ICON
ABC3679
#1AC3X
FED EX
97CF3A8XY
FED EX TRUE-MARK™
DFG 4769
43XZ32B
BOBBY WHITE
83AX9CG42
WHITE LIST ICON
167
169
171
173
175
165
177

FIG. 10

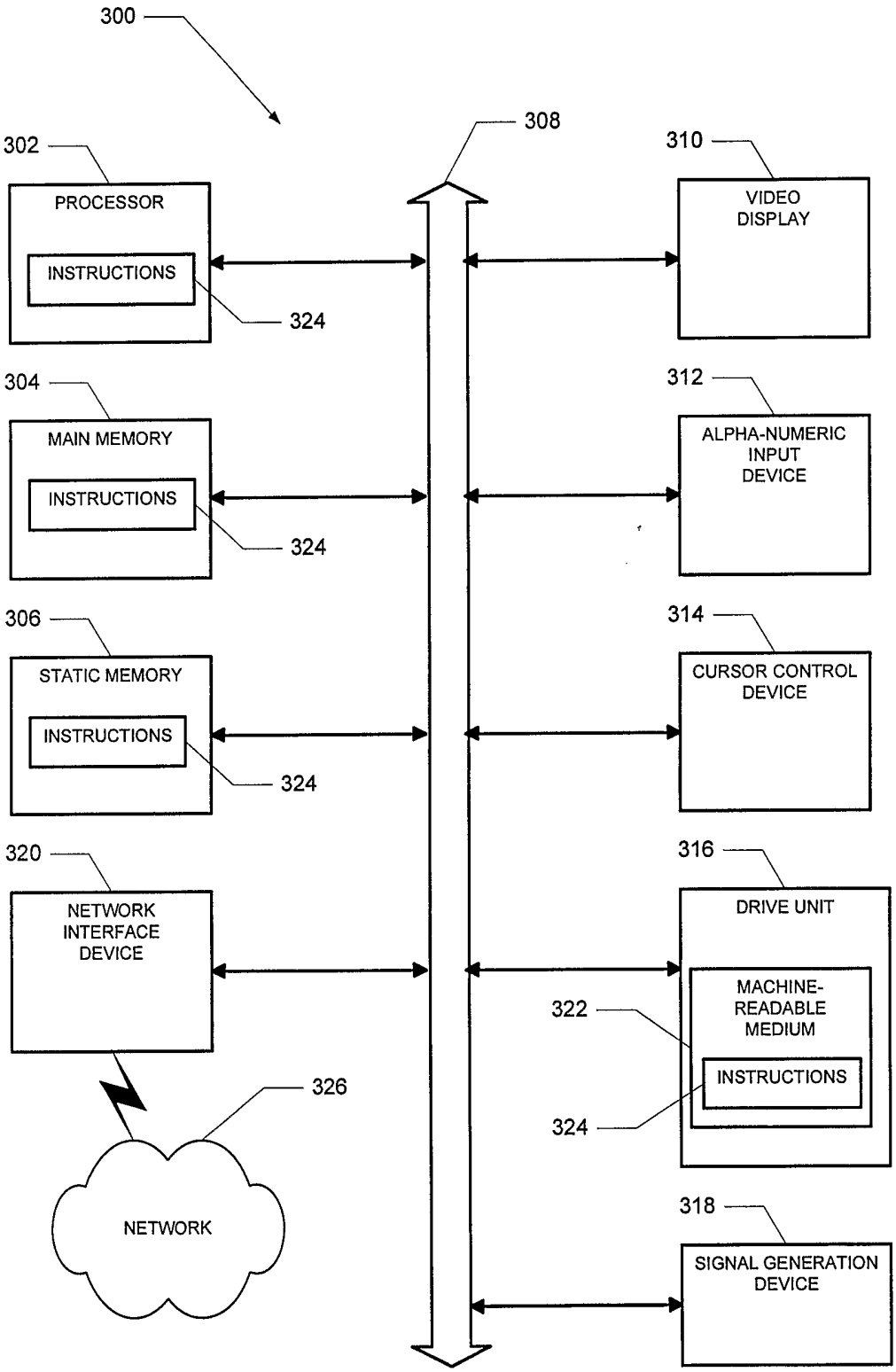

300
302
PROCESSOR
INSTRUCTIONS
324
304
MAIN MEMORY
INSTRUCTIONS
324
306
STATIC MEMORY
INSTRUCTIONS
324
320
NETWORK INTERFACE DEVICE
326
NETWORK
308
310
VIDEO DISPLAY
312
ALPHA-NUMERIC INPUT DEVICE
314
CURSOR CONTROL DEVICE
316
DRIVE UNIT
MACHINE-READABLE MEDIUM
INSTRUCTIONS
322
324
318
SIGNAL GENERATION DEVICE

FIG. 11