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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 December 2004 (09.12.2004)

PCT

(10) International Publication Number
WO 2004/107700 A1

(51) International Patent Classification[7]: H04L 29/06, 12/22, 12/58

(21) International Application Number: PCT/CH2003/000341

(22) International Filing Date: 30 May 2003 (30.05.2003)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: **PRIVASPHERE GMBH** [CH/CH]; Weinmanngasse 114, CH-8700 Küsnacht (CH).

(72) Inventor; and
(75) Inventor/Applicant *(for US only)*: **HAUSER, Ralf** [CH/CH]; Bürglistrasse 23, CH-8002 Zürich (CH).

(74) Agent: **CLERC, Natalia**; Isler & Pedrazzini AG, Gotthardstrasse 53, Postfach 6940, CH-8023 Zürich (CH).

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

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

Published:
— *with international search 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.

(54) Title: SYSTEM AND METHOD FOR SECURE COMMUNICATION

(57) Abstract: A system for secure communication comprises a secure electronic messaging server which is accessible by the public and a database for storing data of subscribers. The data comprise subscriber-based instructions related to the handling of incoming mails, wherein the instructions comprise information related to an at least one security level chosen by the subscriber. The system comprises means to automatically handle in- and outgoing mails according to these instructions. This system allows transmitting messages privately and securely with minimum prerequisites for its users. The inventive system allows a secure transmitting of messages between a sender and a recipient, wherein only one of the two parties is a subscriber.

WO 2004/107700 A1

System and method for secure communication

Field of the invention

The invention relates to a system, a computer program product and a method for secure communication.

Background of the invention

Asymmetric encryption for end-to-end e-mail encryption is well known in the prior art. Pretty Good Privacy aka Phil's good program, called PGP, by PGP Corporation, Palo Alto California, and its open source follow-up effort GnuPG are one example. Similarly, the OSI standards family has come forth with x509 or S/MIME.

While the technical implementation of the content and attachment encryption and digital signatures are very similar in these technologies, there are many approaches for establishing a trust or security hierarchy with all its ramifications. The pure "web-of-trust" approach is pioneered by GPG/PGP.

The S/MIME family of standards puts significant effort in managing keys and trust by a cascade of central institutions, such as Certification Authorities (CAs) and Registration Authorities (RAs), and in revocation, which are all summarized under the term "public key infrastructure" (PKI).

To date, except for very specific business relationships or inside large corporations, PKIs have not met broad user and consumer acceptance. This is due to several reasons:
Difficulties with liabilities and governance of CAs;
Government institutions are reluctant to certify for data protection and privacy concerns and liability reasons as well;
Long and cumbersome processes are needed for obtaining a key and
the human factor has not sufficiently been considered such as convenience during usage of such systems etc.

One approach to exchange messages that only relied on certificates for HTTPS web access is the now defunct "UPS Secure Online Courier" from United Parcel Service Inc. Similar services have been offered by various other companies. While correct in their intent, these solutions had several points which could be improved, typically:
- non-subscribers cannot respond;
- downloads and installation of software are required;
- some solutions also resort to self-decrypting archives that are a serious problem for anti-virus filters;
- initial passwords are sent in e-mails upon registration;
- no out-of-band channel is supported for alternate sender/recipient verification;
- management of private keys is often delegated to a central service (e.g. crypto-proxy) - thus violating the rule that private keys are never shared;
- costly special purpose servers must be run centrally by the mail domain providing institution;
- lack of integration with the above state-of-the art end-to-end message security standards (due to proprietary approaches);

- possession of private keys by end users is a precondition for the system to work at all;
- it is necessary to obtain yet another e-mail address and traffic analysis is not prevented, so that an attacker can passively collect complete header information on each sent message.

US 2002/0007453 "Secure Electronic Mail System and Method" tries to find a better solution but fails to address many of these requirements because it requires the sender to have a private key and sender-side special purpose software instructions.

US 6,424,718, "Data communications system using public key cryptography in a web environment", stores multiple private keys on a server whereas the private keys are shared among users. It also requires the download of applet code to the client from the server.

U.S. Appl 20030007645, "Method and system for allowing a sender to send an encrypted message to a recipient from any data terminal", basically builds a tunnel to a server hosting recipient public keys and the sender's private key whereas the latter is a significant design flaw because private keys are not to be entrusted to anybody.

Furthermore, the needs of Private Messaging may possibly change in the future, as set out in the following:

Also within the hopefully reasonable checks and balances of civilized states, it is to be expected that on the long run, all professional messaging service infrastructures will eventually be required to allow law enforcement efficient access to the information flows from and to their

subscribers. Thus, both the content as well as the traffic information (header) of all information shared with a provider cannot be expected to remain solidly private. Furthermore, it is to be expected that with or without targeted, individual search warrants, the amount of traffic analysis and content observation conducted between the messaging service infrastructures (i.e. outside the control of the corresponding infrastructure providers) is going to increase.

Therefore, the question occurs, what individuals can do to improve or even maximize their privacy legally. The answer is, that they can encrypt end-to-end or use anonymous mixes, relays or similar systems.

If in the law, the balance is further shifted towards more means for effective law enforcement, most likely mixes or relays that are anonymous by cryptographically strong algorithms will be outlawed prior the individual being denied access to industry-strength cryptography as it is readily available today in open source form.

In most jurisdictions, an individual currently has the right to decently protect content end-to-end; possibly at the cost of needing to download the corresponding software unless it is already integrated in applications of the default install.

By choosing a responsible messaging service infrastructure provider, i.e. one that protects all connection links and interim storage with strong cryptography, the user will also be able to determine which jurisdiction will have access to the headers in their entirety and - if not encrypted end-to-end - also the content of a message.

Whatever one's mail provider promises in respect of privacy and security, super-encrypting end-to-end with a downloaded file-encryption software such as PGP/GPG provides optimum privacy under the expected legal circumstances, since the content will be hardly decipherable by anybody and the traffic pattern/header is only visible to insiders, i.e. the infrastructure providers of the sender and the recipients and their corresponding law enforcement agencies and traffic data is not fully visible to non-targeted mass-scanning/traffic analysis.

Under these circumstances, users may furthermore prefer to refine their own exposure profile towards the non-targeted mass-scanners. While it appears that shielding notification messages and other content uploads/downloads entirely from non-specific mass traffic analyzers is impossible, with a comparably simple download of a secure tunnelling software there are at least the following two scenarios imaginable:

a) end-to-end encryption of the content by the user to the recipient and using a secure messaging infrastructure provider in the jurisdiction of preference to conceal the counterpart header information;

b) tunnel to link into a messaging provider within the jurisdiction preferred for the traffic analysis data and have up- and downloads as relayed secure traffic (e.g. SSL) through that tunnel and subsequent peering tunnels between the front-end infrastructure provider and the interim-storage infrastructure provider.

When using the first scenario a higher security is achieved because there is not really a need to store the content with any trustworthy provider in plaintext. The second scenario has lower requirements on the end-user with regard to the required key management effort necessary to obtain an

acceptably secure messaging solution - no direct "key-trust" must be established between each sender and recipient, especially when taking the invention described next into consideration.

Summary of the invention

It is therefore an object of the invention to provide a system and a method for performing a secure and private electronic message exchange avoiding the above-mentioned shortcomings.

This is achieved by a system, a computer program product and a method for secure communication, i.e. for electronic private message transmission, according to the features of claim 1, claim 24 and claim 25 respectively.

The inventive system allows transmitting messages privately and securely with minimum prerequisites for its users. The inventive system allows a secure communication between users without the need for the users to have the same encrypting software or the same security level.

In preferred embodiments, the inventive system allows a secure transmitting of messages between a sender and a recipient, wherein only one of the two parties is a subscriber. The system even allows a secure reply by a non-subscribing recipient. It allows secure store & forward replies without having end-to-end message encryption facilities installed on the replier side. It is also able to translate between different encryption systems. The inventive system also allows to be run by having a recipient account at the inventive system with a public key that may

but needs not be globally known or that may but needs not to be certified by a globally known CA.

In one preferred embodiment, the inventive system provides an automatic encryption relay. In another embodiment, it allows to send mass-mails in a secure way. The inventive system furthermore allows communication between a variety of providers and allows the use of different service levels.

In a further preferred embodiment, the inventive system allows that subscribers are permanently reachable securely without either having to force the sender nor recipient to download software, exchange public keys and/or reveal their e-mail or name. The only thing needed is a state of the art Web Browser that supports strong encryption (such as SSL/https in Microsoft Internet Explorer, in Netscape Navigator, in Mozilla or in others). The inventive system just provides a URL uniquely identifying the recipient to the senders. This also functions as a "private business card and contact me" module. In a variant, it may also display the recipient's public keys, and other directory or recipient-chosen information such as advertising.

Subscribers can define their preferred trust management using multiple security levels. In a preferred embodiment, the inventive system comprises a trust management module that supports web-mail users to manage the out-of-band verification of their counterparts. This module increases the efficiency of the communication, since an out-of-band counterpart verification is not needed any more upon each exchange, but only when a trust-lowering event occurred with either party such as an authenticator-reset (e.g. a "forgot password event").

In another preferred embodiment, the inventive system allows a seamless upgrade to more security and to start digital rights management features. This can for example be achieved by uploading a public key. This has the advantage that it breaks the "negative network effects" of the state of the art, in that a secure channel can only be established if both parties have a public-private key pair that is compatible and whose initial trust is established via a CA or fingerprint out-of-band verification. A further advantage is, that the exposure of the content of the message is reduced, since the message body does not need to reside at the service system after being encrypted to the recipient. A third advantage is, that the inventive system allows publishing the public key by the service provider's directory, thus resulting in some implicit certification. The public key can also be used to facilitate login as client certificate. Furthermore, for example a password reset can be achieved without trust impact because the user's private key is a second trust-establishing secret. The sophistication of the authentication technology employed for account access can be set to different security levels either user-, employer- or service-provider-driven.

In a further embodiment, the inventive system offers a secure SMTP server that accepts messages from arbitrary domains. The only precondition is that the user identification is a subscriber's e-mail and the corresponding authenticator such as password or client certificate, etc. is valid. This SMTP server is not a typical message transfer agent (MTA) because it does not have to connect to another MTA. This server parses the mail received and enters it into the same database as if the message were submitted via the web-interface. Optionally, if public keys shall be used

to encrypt the messages to the recipients, those are attached with a particular naming convention allowing the system to determine which recipient relates to which attached key.

Furthermore, the inventive system gives the users the choice under which jurisdiction they want their messages stored in transit and from which jurisdiction they want to obtain notifications if at all. It allows federated operation of multiple private messaging server domains. The user is able to specify the server's service level. The multiple instances of the service are run according to different scenarios that determine the security of information flows as well as end-user convenience:

- in the "Island" scenario, there is no connection to the outside world. It maintains its own, closed customer base.
- in the "Complete Black Hole" scenario, the sender has to know that the recipient is subscriber to a black-hole instance of the service. In this case, the message can be delivered to there. Otherwise, the message remains at the sender's submission service server and the recipient is treated as a non-subscriber. The basic principle behind this scenario is, that there is no information outflow from the black hole.
- in the "Content Black Hole" scenario, the "no information outflow" principle is maintained for message content and subject, but this instance of the service exchanges information about its subscriber base with its peers and possibly also reports the message status back to the sender service.
- the "True Federation" scenario requires the addition of a directory service. Especially if run by the same provider, this directory service will also contain some

indication of network-geographic positioning in order to minimize the network load and distance when moving and storing large attachments.

Further preferred embodiments of the invention are described in the dependent claims.

Brief description of the drawings

The present invention will be more readily understood upon reading the following detailed description in conjunction with the drawings in which:

Figure 1 shows a generic encryption relay with a PrivaSphere relay alias for individuals;

Figure 2 shows a generic encryption relay without the need for individual relay addresses for entire recipient domains;

Figure 3 shows how a message can be encrypted to a recipient when the sender has the public key, but cannot encrypt the content due to lack of software or permission to use it;

Figure 4 shows how a message can be encrypted to a recipient when the sender has the public key, but cannot encrypt the content for the recipient due to lack of software for the recipient's specific key type ("key translation");

Figure 5 shows the most restrictive server instance cooperation scenario apart from pure island solutions: a server operation as "complete black hole";

Figure 6 shows a "content black hole" scenario and

Figure 7 shows an overall network landscape with a fed-

eration of at least approximately equal servers instead of a single hub server.

Description of a preferred embodiment

In this description, the inventive secure messaging system is called PrivaSphere. This inventive system starts from a protected Web-Mail service, preferably a SSL protected Web-Mail service. The invention is described in the following based on the use of such a SSL protected Web-Mail service. However, the scope of protection shall not be limited to this kind of service, since other services are or will be known in the art and may be useable too.

The inventive system comprises at least one electronic messaging server being accessible by the public and a database for storing data of its subscribers. In this database, at least the specific rules given by the subscriber and which are related to the deliverance of mails sent to the subscriber are stored. These rules comprise at least an instruction what to do with specific mails received by the system and which security level should be observed when handling this mail. The system's database can furthermore comprise instructions related to mails sent by the subscriber. As set forth in the following, the inventive system comprises even more features and can handle even more instructions of the subscriber automatically.

The inventive system protects messages to a non-subscriber with a message unlock code (MUC) that must be entered correctly with a limited amount of tries permitted to access a message. The sender transmits this MUC to the recipient

"out-of-band" (for example in a traditional paper letter, SMS, reading it to the recipient via the phone, etc.) in order to prevent eavesdroppers on the electronic "band" such as the Internet to be able to access the message as well.

A subscribed sender can sponsor message replies of non-subscribing recipients: For the matter of the message just received, such a recipient is sufficiently authenticated in a temporary way by providing the MUC. And therefore, the system will allow the recipient to also create a reply within such a session.

In the following, different preferred modules of the inventive system are described. They can be used, which is preferred, altogether in the same system or they can be used as stand alone solutions.

Permanent encryption relay module

Figures 1 and 2 show the permanent encryption relay module.

In the Individual Mail Encryption Relay Sub-module according to Figure 1, the recipient has a public and a private key and a corresponding de- and encrypting software and he is a subscriber of PrivaSphere. The sender can know, but does not have to know the public key of the recipient. He also does not have to be a subscriber of PrivaSphere or to have the same or any message de- and encrypting software.

The sender always sends the message to a PrivaSphere Server, wherein the message can already be encrypted or not. The PrivaSphere Server knows the public key and the real address of the recipient. When the message is not yet encrypted, it will now be encrypted with the public key of

the recipient and forwarded to the real recipient's address and therefore to the Mail Server of the Receiver Domain. Optionally, it can be signed digitally by PrivaSphere. The recipient retrieves the mail from his Mail Server at his convenience, he decrypts it and verifies the signature, if appropriate. In one embodiment, the recipient's mail user agent (MUA) such as Outlook can also directly connect to a PrivaSphere mail server - for example by means of an SSL-protected POP3 or IMAP protocol.

Such secure access to the PrivaSphere server is possible by all other modules and embodiments where the receiver is a subscriber too.

If the recipient subscriber wants to force incoming mail also from non-subscribing senders to be encrypted, the system has to issue a new e-mail address for the recipient ("relay address"). The recipient never needs to access an account under this new address because all messages will be forwarded to the subscribers existing mail account in encrypted form.

The recipients furthermore can avoid having to publish the system-generated e-mail addresses as their new mail address. They obtain an e-mail forwarding address by an institution of their choice (e.g. alumni address at a school). Typically, they would forward mail going to this address to their address at their current main mail hoster (e.g. ISP, employer, etc.). In the context of the invention, this forward points to the system generated relay e-mail address. With this approach, the system-generated e-mail remains "internal" between the recipient and the PrivaSphere system.

In the case of allowing also non-subscribers to use the encryption relay, PrivaSphere can withhold message delivery and only send summaries of the pending messages to the recipient. The recipient then can check off unwanted messages, such as spam mail or mails from senders named by the recipient. This helps to save bandwidth since unwanted messages do not have to been downloaded.

Such "relay addresses" and the forwarding mechanism described above can also be used in the core module to include messages from non-subscribers into the service databases without them noticing, irrespective of whether the recipient has a public key or not. In doing so, additional measures against spam will be included as well.

In the Corporation Mail Encryption Relay Sub-module according to Figure 2, we have the same constellation of recipient and sender. However, this time, PrivaSphere does not have to issue a relay address for the recipient, it only needs to know each recipient's public key. It is sufficient, that the Domain Name System (DNS) publishes a PrivaSphere server as the corporation's e-mail server. The PrivaSphere server furthermore is configured to relay all mail to a non-published server at the corporation (e.g. plaintextMail.corporation.com). This mail server can be configured to accept only messages from the PrivaSphere server - for example by means of a firewall - to prevent unencrypted mails and spam to reach the recipients.

A message sent by the sender to PrivaSphere is encrypted, where appropriate, and optionally signed digitally before it is forwarded to a ("plaintext") Mail Server of the Receiver Domain. If the corporate security policy allows sending internal mail in plaintext or all internal mail is

sent end-to-end encrypted by default, there is no need for internal mails to pass through the relay. In order to avoid that such internal messages leave the local area for PrivaSphere's encryption relay, the corporation's internal DNS should be set-up such that for example plainTextMail.corporation.com is internally seen as the domain's mail host and not a PrivaSphere server.

Messages relayed with this module can remain outside the trust management system described below because the senders potentially never directly contact a PrivaSphere server and thus cannot be authenticated unless the senders volunteer to authenticate their message e.g. by signing it. In this case, messages only need to be entered into the service databases if the billing is not on a flat-fee basis.

Mass-mail module

The sender is a subscriber to PrivaSphere and wants to send secure mass-mails, be they individualized (e.g. form letters) or not. The sender can for example be a bank, which wants to send confidential messages, such as monthly statements of the individual accounts, to its clients.

The sender can upload the message to PrivaSphere for example with a single file containing the addressing information for all recipients. In return, he gets a list with a MUC for each recipient unless the recipient is a subscriber and is trusted by the sender.

Trust management module

The trust management module enables the users of the system (i.e. the sender or recipient) to minimize the need for out-of-band verification of their counterparts (i.e. the

recipient or the sender), wherein trust management is established in a user-centred way. If sender and recipient are subscribers of PrivaSphere, PrivaSphere acts as a "hub" server being trusted by both parties as the intermediary. This intermediary controls the access to the content of the message and has established an authentication method with each subscriber allowing it to support transitive trust relationships. Under the assumption that either party takes due care of its authenticator - typically a shared secret (e.g. password), something they have (e.g. a hardware device or a private key), or something "they are" (e.g. when using biometric means for authentication) - it is not necessary to ask from a sender out-of-band counterpart verification upon each communication. Typically, the subscriber can choose that verification is only mandated upon major events causing the previous authenticator no longer to be usable - i.e. the password is forgotten or revealed, the private key lost or revealed, the (biological) fingerprint unavailable due to a skin disease or accident/amputation.

If it is acceptable to the sender to only refresh trust themselves under such exceptional conditions, the cost of trustworthy communication can be greatly reduced with the following pieces of logic:

i) Build Sender **Trust list** upon each message submission:
foreach recipient
    if recipient is a valid member of sender's trust-list
    then send message
    else
        prompt sender for out-of-band verification result
        if verification result = OK
        then
            send message
            enter the pair (recipient, timestamp) into
                sender's trust list
        else
            save message for recipient in queue for later
                sending upon verification acknowledgement by the sender (e.g. in case of a mail-based, i.e. store-and-forward message submission) OR

```
                                wipe it
                        end if
                end if
        end foreach

        ii) update every sender's trust-list upon each authenticator reset:
        foreach trust-list
                if the reset'ed subscriber is a member
                then
                        if recipient-addition-time-stamp < reset timestamp
                        then remove reset'ed subscriber from trust-list
                        end if
                end if
        end foreach
```

The performance can be optimized among others by indexing the lists by members or by implementing a lazy trust-list update. The second means, that all trust lists are not proactively walked upon each authenticator reset, but that on demand - i.e. upon checking the status of a recipient or sender when sending or receiving a message - the trust-list member-addition date is compared with an authenticator reset journal.

Essential for the correctness of this trust list management feature is the availability of reliable "last authenticator reset dates" per subscriber. The efficiency lies in the completeness of the trust list per participant and its up-to-date-ness and the access speed.

Also, the trust list cannot only be used upon sending a message, but also to indicate to a recipient her or his relationship with the sender and the other recipients of a message. Especially, the recipient can be warned if a sender previously was out-of-band verified, but had an authenticator reset since.

Not every forgot-password event has necessarily to be an authenticator reset. For example, there can be a combination of authenticators available to a server by the sub-

scriber uploading a public key. In the forgot password case, an encrypted challenge is sent to the subscriber and if it can be decrypted, this can avoid the above trust reset process. Essentially by uploading the public key, the subscriber has introduced a second authenticator beyond the initial password. Transitively, the presence of the private key can then be used by the subscriber to choose a new password instead of a forgotten one without the need for a basic trust-re-bootstrapping.

Seamless upgrade and flexible policy module

Based on the above described trust management module, it is also possible for a sender and a recipient being at a different level of security to communicate with each other.

Since the public-private key pair is at least initially only used "bilaterally" between the user and service system, there is no requirement for certification by a certification authority. The fundamental principle is that the security-level of each individual message is recipient-driven. The sender can for example set a policy not to send a message if the recipient is not capable of receiving at a certain security level, or alternatively, force the sender to a higher security level, if the sender account can operate at different security levels, but the sender currently defaults to use a lower security level. The system allows the recipient to determine that all message notifications and service messages are encrypted with the public key. Such message notifications can be among others the notification of an imminent expiration of a message or the notification that a recipient locked her-/himself out by entering wrong message unlock codes three times in a row. An ex-

ample for a service message is, that the information about the remaining account balance.

The inventive system can furthermore encrypt each message it sends to a deputy key. In addition, in case of vacation absences or similar events, a deputy cc message can be sent to the according recipient in encrypted form.

If the sender for example uses GPG upload as described in the PrivaSphere public key module below and the recipient S/MIME, effectively "translation between cryptosystems" is achieved.

The inventive system allows the sender to specify output controls in the sense of digital rights management (DRM). In an initial step the text and other presentational material is presented in a form that cannot be edited with effective tools anymore, e.g. by using a bitmap representation of text. The system also allows the sender to mandate secure viewers and other DRM parameters (e.g. expiration of reading right, forwarding restrictions, etc.) in its settings.

Additionally, the inventive system allows the subscriber to choose, if he wants to receive a notification about having mail or if he wants to retrieve the messages proactively without being triggered by the system. In the preferred embodiment, the subscriber has to set a flag, if he wants to receive messages only when logging in according to the "get principle". If the subscriber chooses to set this flag, he can once again minimize the exposure to e-mail traffic analysis.

"Private business card contact me" module

The secure "private business card contact me" module allows many individuals as well as for example institutions to be reachable through the inventions secure messaging service in a significantly more private way than before by virtue of a simple, unique URL at the PrivaSphere server. One reason to choose this module is that they cannot afford to run a secure website themselves. When following that URL the PrivaSphere brings up a web-mail form that resembles standard web mail or "contact us" forms a lot. Its usage is pre-paid by the recipient, which is the subscriber, with the following additional features:

i) if a public key is uploaded and the subscriber agrees to disclose it, the module can display the public key in order to allow for end-to-end content encryption, for example using GPG or S/MIME;

ii) if the subscriber wants to protect his privacy, e.g. to prevent spammers from reading the e-mail address and reselling this, his privacy can be selectively increased, i.e. either the e-mail address or the real-name or corporate affiliation/Logo or all of them can be hidden from the ("walk-by") sender;

iii) It is possible to configure "From" identification requirements for the sender in a flexible way. For example can a provided e-mail address be verified with a challenge until a contact-me-message is accepted, it can be just required to present, but unverified, or it can even be left optional altogether, thus allowing anonymous message submission.

Secure Mail Submission Server module

In a preferred embodiment, PrivaSphere is not exclusively

focused on web-mail-like access, but also contains a standard mail submission server. One standard used currently is the "simple mail transfer protocol" called SMTP. The mail submission server variant as proposed by this invention can only be reached in a private way, i.e. the connection to do so is protected by an adequate means for confidentiality such as SSL (on the standard port, only redirect or error messages are issued). A normal SMTP server accepts mail typically only from users of its own domain e.g. bluewin.ch. In this preferred embodiment, PrivaSphere accepts also other mails, as long as the sender is a subscriber and can prove this by proper use of an authenticator. PrivaSphere therefore accepts an arbitrary "From" field being the e-mail address identifying the subscriber account.

Lastly, unlike known MTAs, it does not just "store-and-forward" it to the destination MTA, but it inserts the message into a server's message and trust management system resulting in the representation in the system not unlike it were after a web-based upload. The big advantage for the subscriber is that unlike when using web-mail, the subscriber's machine needs not be network-connected at the very time of sending a message, but the regular mail client (MUA) can be used.

Figure 3 furthermore shows how a message can be encrypted to a recipient when the sender has the public key, but cannot encrypt the content due to lack of software or permission to use it. In this case, the sender needs to be subscriber of a PrivaSphere server. Such messages may not be included into the invention's trust management because non-subscribing recipients never authenticate with the PrivaSphere server to access their incoming messages. Alterna-

tively, if among multiple recipients there is also a subscriber or some of the recipients have no public key, using the trust management module for such a message is advisable.

As shown in Figure 3, by attaching a recipient public key, the system can be caused to encrypt the message for this recipient irrespective of whether the recipient is already a subscriber and whether that public key is already known to the server or not. Alternatively to attaching such a key, a globally unique key identification (ID) can be provided instead. Also, policy flags could be set to determine whether the public key corresponding to the key ID can only be used when already stored on the PrivaSphere server (e.g. by virtue of the recipient having uploaded it her- or himself) or whether the server can reach out to other key directories to retrieve it. A further policy input by the subscriber can be, that PrivaSphere has to determine which third-party key directories can be searched for a key under what condition such as "which certification authority needs to have certified it if any".

This above mentioned sender-provided public-key approach can equivalently be realized in the already described web-mail-based upload approach.

Non-subscribing recipients can reply on the sender's cost in a store-and-forward way if the message contains a one-time reply-to-mail address. In order for this to maintain security, the non-subscriber must add the PrivaSphere mail server as additional, private "mail account" in their MUA and to connect to it in a secure manner (such as SSL). In today's MUAs this means to add at least another SMTP server. An alternative is that the replier uses a Priva-

Sphere public key and the corresponding, specific PrivaSphere mail address in encrypted form as described in the following module.

Figure 4 shows how a message can be encrypted to a recipient when the sender has the public key, but cannot encrypt the content for the recipient due to lack of software for the recipient's specific key type. Again, the sender needs to be subscriber. Such messages typically are not part of the trust management because the recipient typically never authenticates with PrivaSphere.

PrivaSphere public key module

This module is used when the sender's MUA is not capable of protecting its connections such as securing SMTP upload with SSL or if the user is not allowed to add additional mail accounts i.e. SMTP servers. The sender, however, is assumed to be able to encrypt messages in his MUA to a sender-chosen public key. This module therefore requires the sender to encrypt the message to a particular address at the PrivaSphere service with a globally known and used public key for this address. The service then decrypts each message incoming to this particular PrivaSphere address as regular, but encrypted e-mail (for example as per rfc822 and the S/MIME or GPG extensions to it), parses the received information and enters it into the same database as if the message were submitted via a web-interface. The difference is that the sender needs to manually add further instructions to the message, how it shall be processed. For example, the first lines of the message contain the "To:", "Cc:", "Bcc:", and "Subject" fields that determine to whom the message shall be forwarded and how. Other ways to instruct the server what to do with such an incoming message

are also considered. Optionally, public keys can be attached like in the above mentioned Secure Mail Submission Server module.

This approach also allows for key translation, i.e. to handle recipient public keys that are incompatible with the encryption system of the sender. If it is a key translation among subscribers, the sender need not worry about the recipient key because this is recipient-determined and the message exchange is falling under the trust management module.

Messages will be billed according to the original from-address. In order to prevent misuse of this module at the subscriber's cost, multiple strategies can be applied:

i) Replies can have a unique message identifier only known to the server and the sender.

ii) The sender's mail-host network address can be used to limit the defrauding potential to other users from that domain, provided the mail-hosts of the other users implement good spam-protection such as preventing "open relays" and assuming network address spoofing to be hard.

iii) The message can be digitally signed by the sender and being verifiable with a key known to the server.

iv) A password can be included in the message body.

In all embodiments where the sender and not the system supplies the recipient public key, the trust management can be configured to impose an extra-exchange between the sender and the server to ensure the out-of-band verification of all the recipients who haven't undergone this procedure yet from the sender's perspective. In this case, messages to

not yet out-of-band-verified recipients will be held pending until the sender acknowledges them to be verified. Alternatively, the policy can be set in the sender's profile that all recipients addressed with the methods of this embodiment can be considered verified by default.

Service level dimensions module

A further module of the inventive system handles the presence of a multitude of servers. The subscriber can choose between at least the following key service level dimensions:

- jurisdiction where the content is stored;
- jurisdiction from which notifications are sent and from - and to which up-and-downloads occur;
- institutions running the service. E.g. a bank might offer more secure technologies both on the server side (for example the number of fire-walled hosting architecture segments, atomic shelters inside mountains, etc.), as well as on the client side (e.g. equipping their customers with security tokens, etc.)

Key scenarios of interaction among such servers are outlined next and reference is taken to figures 5 to 7:

Figure 5 shows the most restrictive server instance cooperation scenario apart from pure island solutions: the server operation as "complete black hole". The overall network landscape assumed is that there is a Hub as a default for the public and other service providers that create their own subscriber communities according to specific criteria. Institutions not wanting the outside world to know about their subscribers are likely to run a server according to the definitions of this scenario.

There is a separate protocol to allow peering servers to exchange user-set policies regarding analyzability of traffic versus choice of jurisdiction that potentially has read access to stored messages:

i) If the black hole's priority is to avoid connections from non-subscribers, then
   a. the upload will take place to the hub,
   b. the hub then relays the message to the black-hole server with leaving minimum traces locally, which is shown as per Flow 2 in Figure 5.

ii) If the black hole rather has the content protected from the hub server's law enforcement, then it instructs the hub server to issue the redirect, as shown per Flow 2a and 2b in Figure 5.

When the subscriber decides to give up the "no-download" premise, there is a third option where the sender installs a secure tunnelling software. This software then is used to connect the user to the Hub Server, that in turn relays an "inner" secure connection, i.e. between the user and the black hole (storage) server, through a second secure tunnel between the hub and the black hole to the black hole server, as indicated by Flow 0' in Figure 7. By relaying the secure upload connection, e.g. https from sender to black hole, in this way, two additional service characteristics can be achieved:

1) The content of the message is at most visible to the law enforcement of the black hole server.
2) The relation between the recipient and the black hole is only visible to the law enforcements of the hub and the black hole, but no other eavesdroppers on the communication path thereto.

An interim approach is that even without the sender and re-

cipient being able to tunnel, also the recipient only accesses the Hub server. This implies that the hub becomes a 2nd order trusted man-in the middle for the recipient. This allows to shield the recipient from arbitrary traffic analyzers, but at the risk of the law enforcement of the Hub server also being able to access contents on the black hole server by posing requests as being genuinely from the recipient. The hub server operators could hardly withstand requests to do so from their own law enforcement agencies or laws could easily be created forcing them to do so.

Figure 6 shows a "content black hole" scenario. Messages take the same route as before, but user convenience can be improved by the servers being able to offer their subscribers the choice of service instances a recipient might be subscribed to. This scenario increases end-user convenience. The sender no longer has to know the black-hole server their recipients are subscribers with, but the hub and other servers taking in directory information will be able to tell whether and where the recipient is subscriber as shown in Flow 1' of Figure 6. If he is subscriber at more than one server, they can even provide a choice of servers through which the recipient can be reached. A precondition for this higher user-convenience is that the servers exchange information about their subscribers in a timely manner as shown in Flow 0 in Figure 6.

Such a "public" subscriber directory record shared needs to contain at least an e-mail address and a server or hoster ID. The default will also support a distributed version of the trust list management. Therefore, the GMT timestamp of last authenticator reset is included as well and will be broadcasted upon an authenticator reset event. Furthermore,

to facilitate end-to-end content super-encryption and provide generic public key directory functions, the servers should also exchange the subscribers' public key(s) provide the owner consents.

If the overall system gets to the level of sophistication where content- versus traffic-analysis-secrecy gets traded off, subscriber servers need to convey to the relevant peers only the user preference for traffic analysis protection or content protection. The servers itself specify whether they prefer to be contacted by peers or by redirected uploads in case the users have no corresponding preferences set. This information obviously needs not generally to be shared because each sharing exposes the information to possibly yet another jurisdiction's law enforcement.

PrivaSphere lets the involved parties take yet another service level decision: The subscribers can decide how to handle message status information. It can be both user- and server-determined whether this will be shared after the initial message submission to the black hole. A likely design principle is that each subscriber reads about the status of messages of her or his interest at the own subscription server. A message-storage-location-centred approach is conceivable but poses significant authentication challenges and as per these very scenarios, messages can be stored in multiple locations under very different service levels.

Figure 7 shows the overall network landscape, when there is no more single "hub" server operator, but possibly a federation of servers equal in most respects. In order to keep

the overall system efficient, there will be a central directory service that also maintains the core of the trust-list management function and possibly other services such as financial clearing among service providers.

If the network for private message transferring servers is to scale further, the notion of a "hub-server" will be divided into a directory server and possibly multiple public message holding servers - thus the latter no longer the hub of a hub-and-spoke system. The directory server will continue to be the hub for directory information and is essential to maintaining a distributed trust list that is reliable. It furthermore can handle the clearing/accounting of peer server financial transfers etc. Some such servers may differentiate themselves as described by the technology they are employing be it at their own data center or towards the subscribers including equipping them with it. This may lead to policy situations that will be part the shared directory entries such as:

"Don't even try to submit messages to me unless:

- you are authenticating yourself to the system with a special purpose crypto-device of type X or
- every link is at least encrypted with 512 bit (symmetric) key strength or
- your attachments/content is certifiably virus scanned by method xyz or
- ..."

Scalability can further be facilitated if multiple physical servers constituting one subscriber base use geo-proximity redirects to load balance message content and potentially large attachments. Such an architecture adds yet an extra layer of distributed authentication and user management.

The scenarios so far have always assumed a single server peering relation. Surely, multiple hop server peering relations can be conceived as well also across multiple jurisdictions.

5

Claims

1. A system for secure communication, the system comprising a secure, electronic messaging server being accessible by the public and a database for storing data of a subscriber, the data comprising at least subscriber-based instructions related to the handling of incoming mails, characterized in that the instructions comprise information related to an at least one security level chosen by the subscriber and that the system comprises means to automatically handle in- and outgoing mails according to the subscriber-based instructions.

2. The system according to claim 1, wherein the system enables the subscriber to have more than one security level.

3. The system according to one of claims 1 or 2, wherein the database comprises an email address for forwarding incoming mails, for which the subscriber is the intended recipient.

4. The system according to one of claims 1 to 3, wherein the data comprises at least one public key of the subscriber.

5. The system according to one of claims 1 to 4, wherein, when a mail sent by a sender is received by the system, the system comprises means for automatically encrypting the mail by using a recipient's public key before sending it, wherein the encryption is made according to the chosen security level of the subscriber, the subscriber being either the recipient or the sender of the mail.

6. The system according to claim 5, wherein, upon request of a domain related to the system, the system sends the mails to all recipients of this domain in encrypted form.

7. The system according to claim 6, wherein the system sends the mails without issuing a relay alias email address per recipient to a non-published mail server of that domain.

8. The system according to one of claims 1 to 7, wherein the system comprises means for digitally signing mails when forwarding them to the subscriber.

9. The system according to one of claims 5 to 8, wherein the system comprises means for encrypting mails when forwarding them to the recipient by using a public key provided by the sender.

10. The system according to one of claims 5 to 9, wherein the system comprises means for receiving an encrypted mail, the mail having been encrypted by the sender using a first encryption system, the system further comprising means for encrypting the message with a second encryption system and for forwarding the encrypted mail, which is encrypted by the second encryption system to the recipient.

11. The system according to one of claims 1 to 10, wherein the system sends upon request mails to a black hole server until retrieval or forwarding the mail to a recipient's mail server of recipient's domain.

12. The system according to one of claims 1 to 11, wherein the system provides a sender of a mail with a selection of servers being useable as black hole servers.

13. The system according to one of claims 1 to 12, wherein the system comprises a "contact me and business card" module, which provides a secure web-mail form resembling standard contact forms of web-sites .

14. The system according to one of claims 1 to 13, wherein the system comprises a mass-mail module, enabling a subscriber to upload a message to be forwarded as mass-mail or form mail to the system and providing the subscriber with a list of message unlock codes for each not yet out-of-band verified recipient.

15. The system according to one of claims 1 to 14, wherein the system comprises a trust management module enabling electronic messaging users to manage out-of-band verification of their counterparts.

16. The system according to claim 15, wherein the trust management module forms pairs consisting of said user and said counterpart and wherein said module records for each pair their initial trust-establishing out-of-band verification.

17. The system according to claim 16, wherein the trust management module monitors on the subscriber's behalf all trust relationships in between pairs which comprise the subscriber as the user or the counterpart and wherein the module automatically warns the subscriber upon trust-destroying actions by one of the counterparts of said pairs, the trust-destroying action being

in particular an authenticator reset.

18. The system according to one of claims 1 to 17, wherein the system comprises a secure mail server which accepts incoming electronic messages sent by the subscriber from arbitrary domains and secures its connections.

19. The system according to one of claims 1 to 18, wherein the data comprises a jurisdiction concerning the way of storing the subscriber's mails in transit and/or instructions concerning notifications.

20. The system according to one of claims 1 to 19, wherein the system comprises a service level dimensions module handling the presence of multiple of servers, the module comprising data concerning network-geographic positioning.

21. The system according to one of claims 1 to 20, wherein the system is based on a SSL protected Web-Mail service.

22. The system according to one of claims 1 to 21, wherein the system acts as a trusted server for a sender and a recipient both being subscribers.

23. The system according to one of claims 1 to 22, wherein the system comprises a security level upgrade module, that is recipient-driven.

24. A computer program product to be used in a system for secure communication, the system comprising a secure, electronic messaging server being accessible by the public, the computer program product providing means

for storing data of a subscriber in a database, the data comprising at least subscriber-based instructions related to the handling of incoming mails, characterized in that the instructions comprise information related to an at least one security level chosen by the subscriber and that the computer program product comprises means to automatically handle in- and outgoing mails according to the subscriber-based instructions.

25. A method for secure communication using a secure electronic messaging server which is accessible by the public and which has a database for storing data of a subscriber, the method comprising the steps of storing data comprising at least subscriber-based instructions related to the handling of incoming mails, these instructions comprising information related to the subscriber's chosen at least one security level and automatically handling in- and outgoing mails according to the subscriber-based instructions.

26. The method according to claim 25, wherein incoming mails are encrypted according to the subscriber's chosen security level and forwarded to a predetermined server.

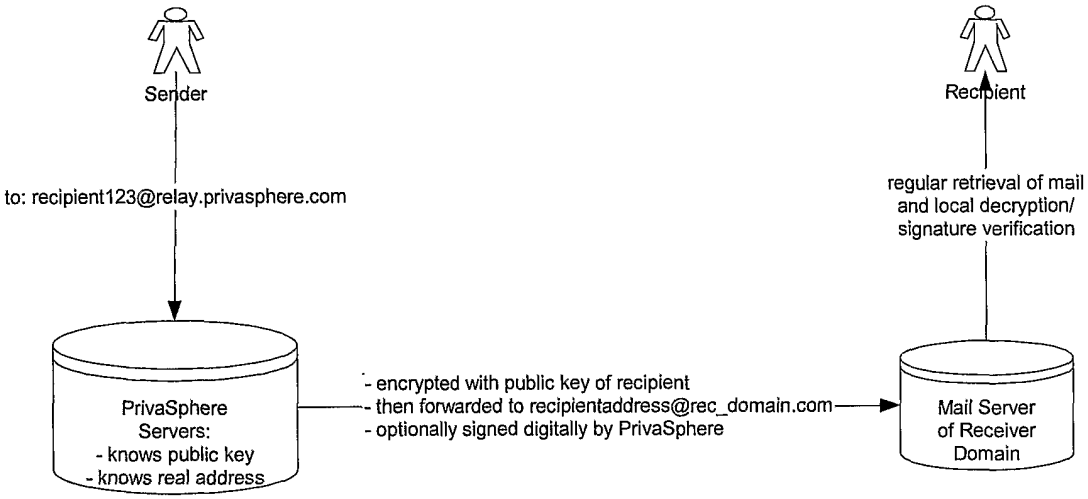
Sender
to: recipient123@relay.privasphere.com
PrivaSphere Servers:
- knows public key
- knows real address
- encrypted with public key of recipient
- then forwarded to recipientaddress@rec_domain.com
- optionally signed digitally by PrivaSphere
Mail Server of Receiver Domain
Recipient
regular retrieval of mail and local decryption/ signature verification

Fig. 1

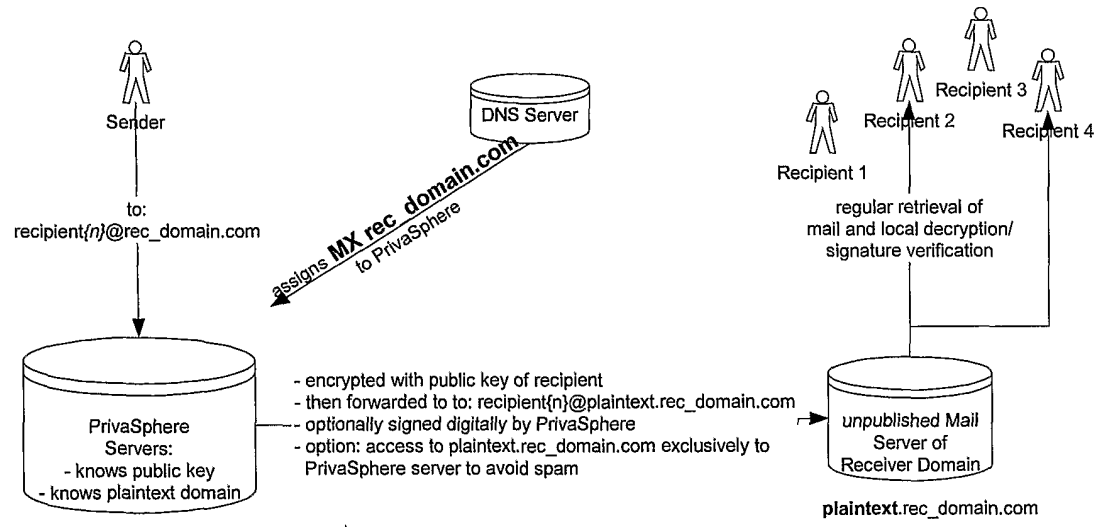
Sender
to: recipient{n}@rec_domain.com
DNS Server
assigns MX rec_domain.com to PrivaSphere
PrivaSphere Servers:
- knows public key
- knows plaintext domain
- encrypted with public key of recipient
- then forwarded to to: recipient{n}@plaintext.rec_domain.com
- optionally signed digitally by PrivaSphere
- option: access to plaintext.rec_domain.com exclusively to PrivaSphere server to avoid spam
unpublished Mail Server of Receiver Domain
plaintext.rec_domain.com
Recipient 1
Recipient 2
Recipient 3
Recipient 4
regular retrieval of mail and local decryption/ signature verification

Fig. 2

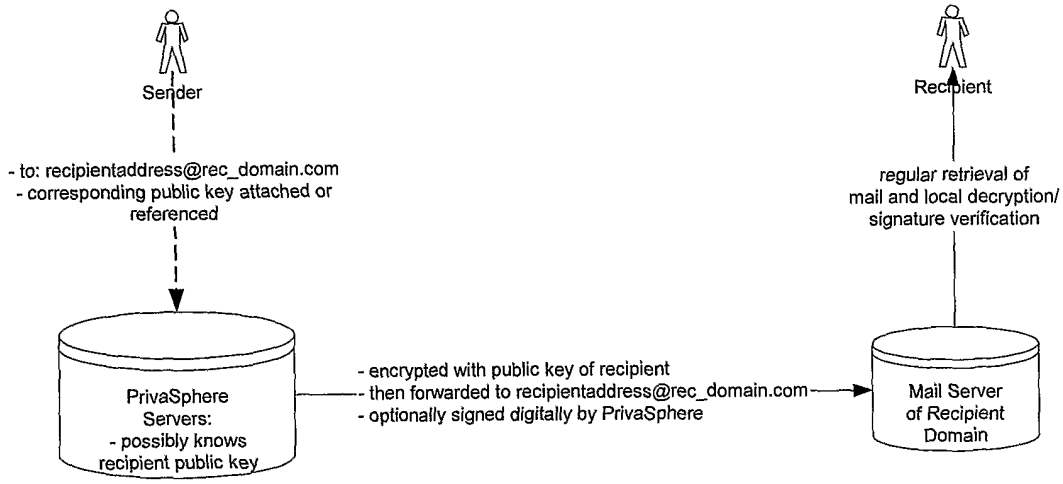
Sender
- to: recipientaddress@rec_domain.com
- corresponding public key attached or referenced
PrivaSphere Servers:
- possibly knows recipient public key
- encrypted with public key of recipient
- then forwarded to recipientaddress@rec_domain.com
- optionally signed digitally by PrivaSphere
Mail Server of Recipient Domain
Recipient
regular retrieval of mail and local decryption/ signature verification

Fig. 3

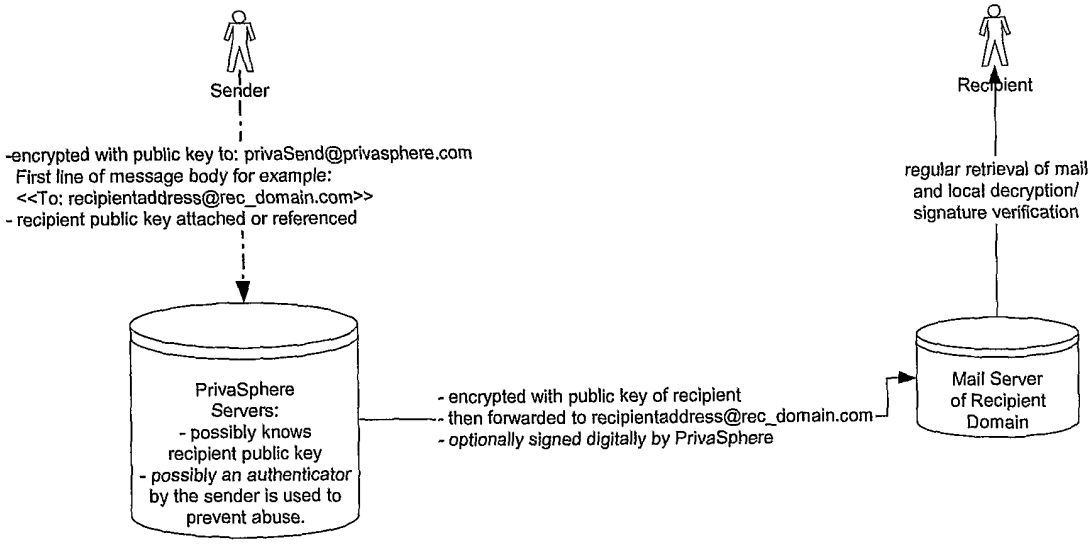
Sender
-encrypted with public key to: privaSend@privasphere.com
First line of message body for example:
<<To: recipientaddress@rec_domain.com>>
- recipient public key attached or referenced
PrivaSphere Servers:
- possibly knows recipient public key
- possibly an authenticator by the sender is used to prevent abuse.
- encrypted with public key of recipient
- then forwarded to recipientaddress@rec_domain.com
- optionally signed digitally by PrivaSphere
Mail Server of Recipient Domain
Recipient
regular retrieval of mail and local decryption/ signature verification

Fig. 4

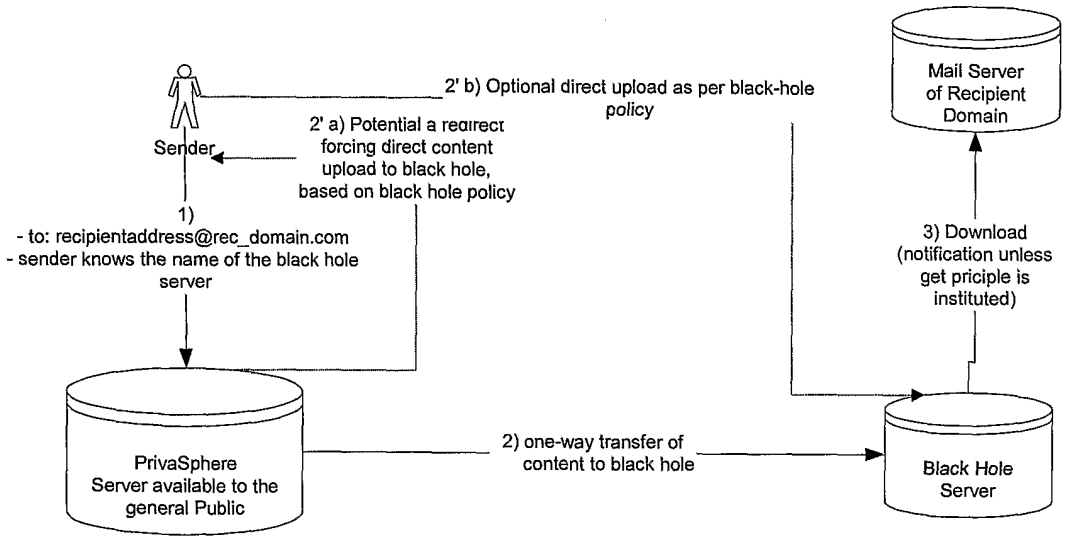
Mail Server of Recipient Domain
2' b) Optional direct upload as per black-hole policy
Sender
2' a) Potential a redirect forcing direct content upload to black hole, based on black hole policy
1)
- to: recipientaddress@rec_domain.com
- sender knows the name of the black hole server
3) Download (notification unless get priciple is instituted)
PrivaSphere Server available to the general Public
2) one-way transfer of content to black hole
Black Hole Server

Fig. 5

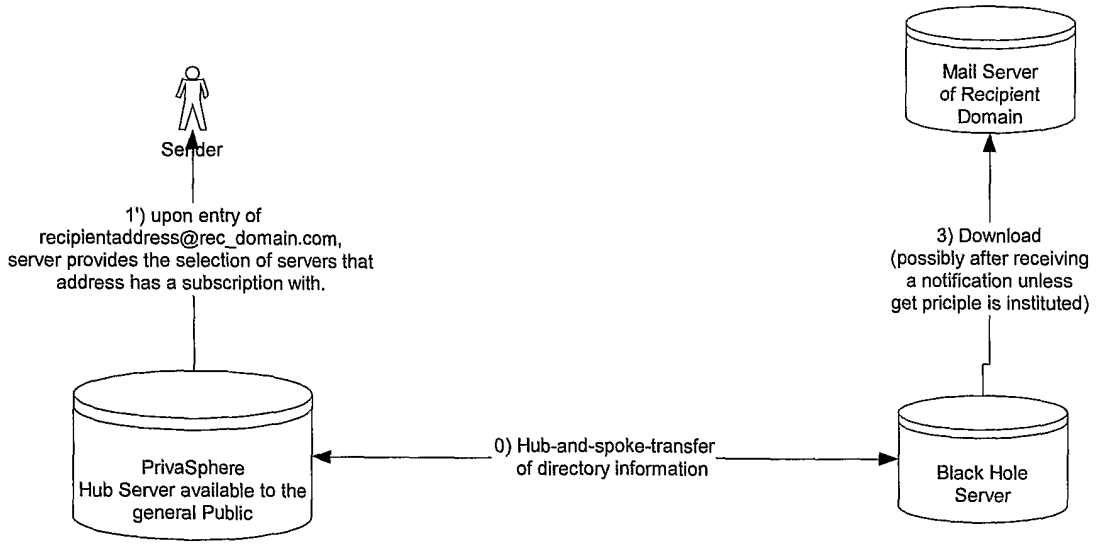
Mail Server of Recipient Domain
Sender
1') upon entry of recipientaddress@rec_domain.com, server provides the selection of servers that address has a subscription with.
3) Download (possibly after receiving a notification unless get priciple is instituted)
PrivaSphere Hub Server available to the general Public
0) Hub-and-spoke-transfer of directory information
Black Hole Server

Fig. 6

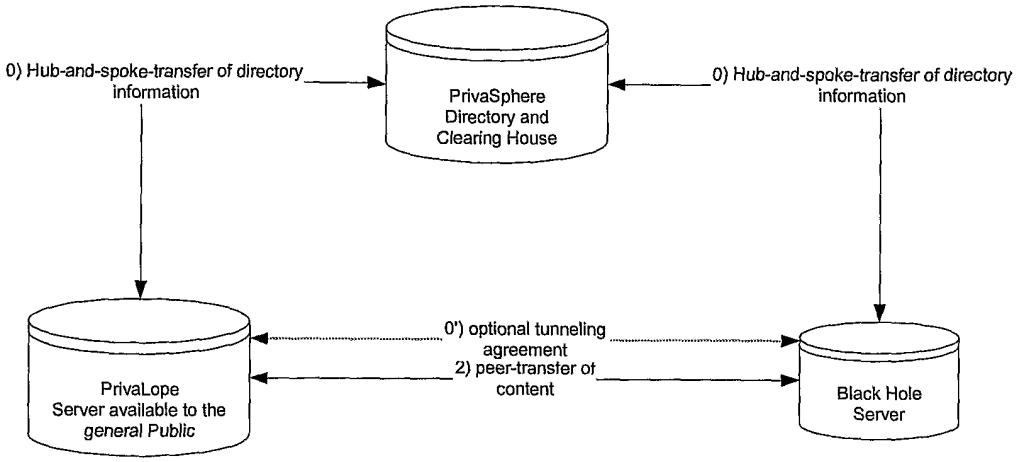
0) Hub-and-spoke-transfer of directory information
PrivaSphere Directory and Clearing House
0) Hub-and-spoke-transfer of directory information
0') optional tunneling agreement
2) peer-transfer of content
PrivaLope Server available to the general Public
Black Hole Server

Fig. 7

INTERNATIONAL SEARCH REPORT

International Application No
PCT/CH 03/00341

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/06 H04L12/22 H04L12/58

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data, PAJ

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 01 97089 A (COOK DAVID P ;ZIXIT CORP (US)) 20 December 2001 (2001-12-20)	1-6, 8-11,18, 19,21-26
A	page 2, line 23 -page 3, line 19 page 5, line 16-24 page 10, line 1-7 page 12, line 9-27 page 14, line 16-29 page 20, line 13-29	7,12-17, 20

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

° Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 February 2004	04/03/2004

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL – 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Lázaro, M.L.

INTERNATIONAL SEARCH REPORT

International Application No
PCT/CH 03/00341

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication,where appropriate, of the relevant passages	Relevant to claim No.
X	US 6 324 569 B1 (OGILVIE GENIE L ET AL) 27 November 2001 (2001-11-27)	1-5,8,9, 22,24-26
A	column 2, line 31-52 column 3, line 33-48 column 5, line 59 -column 7, line 16 column 11, line 19-53 column 12, line 5-8 column 16, line 8-35 column 18, line 36-39	6,7, 10-21,23
X	US 2003/009698 A1 (RUMPELEIN JOHN ET AL) 9 January 2003 (2003-01-09)	1,3,7, 11,15, 18,22, 24,25
A	paragraphs '0044!,'0045!,'0053!,'0121!	2,4-6, 8-10, 12-14, 16,17, 19-21, 23,26
X	WO 02 33872 A (AUSTAD MELISSA ANNE ;KENNEDY JOHN C (US)) 25 April 2002 (2002-04-25)	1-3,15, 18,21-26
A	page 2, line 11 -page 3, line 4 page 3, line 19 -page 4, line 28 page 5, line 3-23 page 6, line 12-17 page 7, line 11 -page 9, line 17	4-13,16, 17,19,20
X	EP 1 003 308 A (LUCENT TECHNOLOGIES INC) 24 May 2000 (2000-05-24)	1-3,9, 18,19, 22-25
	paragraphs '0005!,'0006!,'0013!-'0015!	
A	US 6 161 181 A (FRIEDMAN THOMAS JAY ET AL) 12 December 2000 (2000-12-12) column 2, line 53-67 column 4, line 1-16 column 8, line 63 -column 9, line 19 column 10, line 16 -column 11, line 35 column 15, line 16-45	1-26

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/CH 03/00341

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0197089 A	20-12-2001	AU 6697101 A	24-12-2001
		EP 1311984 A1	21-05-2003
		WO 0197089 A1	20-12-2001
		US 2004025057 A1	05-02-2004
US 6324569 B1	27-11-2001	AU 7106200 A	24-04-2001
		WO 0122243 A1	29-03-2001
		US 2002026487 A1	28-02-2002
		EP 1116126 A1	18-07-2001
		WO 0017768 A1	30-03-2000
US 2003009698 A1	09-01-2003	NONE	
WO 0233872 A	25-04-2002	AU 1664202 A	29-04-2002
		WO 0233872 A2	25-04-2002
EP 1003308 A	24-05-2000	DE 69910952 D1	09-10-2003
		EP 1003308 A1	24-05-2000
		JP 2000201170 A	18-07-2000
		KR 2000035055 A	26-06-2000
US 6161181 A	12-12-2000	NONE	