(12) UK Patent Application (19) GB (11) 2531713 (13) A

(43) Date of A Publication 04.05.2016

(21) Application No: 1418961.7

(22) Date of Filing: 24.10.2014

(71) Applicant(s):
Clearswift Limited
(Incorporated in the United Kingdom)
1310 Waterside, Arlington Business Park, Theale, READING, Berkshire, RG7 4SA, United Kingdom

(72) Inventor(s):
Guy Bunker

(74) Agent and/or Address for Service:
Haseltine Lake LLP
Redcliff Quay, 120 Redcliff Street, BRISTOL, BS1 6HU, United Kingdom

(51) INT CL:
G06F 21/62 (2013.01)

(56) Documents Cited:
GB 2518433 A **WO 2011/059510 A1**
US 20090222527 A1

(58) Field of Search:
INT CL **G06F, G06K**
Other: **WPI, EPODOC, TXTE, Internet**

(54) Title of the Invention: **Automatic consistent redaction of text**
Abstract Title: **Automatic redaction of text by identifying a token**

(57) A template is defined which has: a predefined structure, a starting iteration value, and a mechanism for modifying the starting iteration value of the template to produce subsequent iteration values. The method comprises: identifying 100 a first token within the file as being an example of one type of potentially sensitive information; generating a replacement token 106 by using a first iteration value of the corresponding template; and replacing a first token with a replacement token 108. If a subsequent token is identified as another example of the same type of potentially sensitive information, a second replacement token is generated by using a subsequent iteration value of the corresponding template, and replacing said subsequent token with said second replacement token.

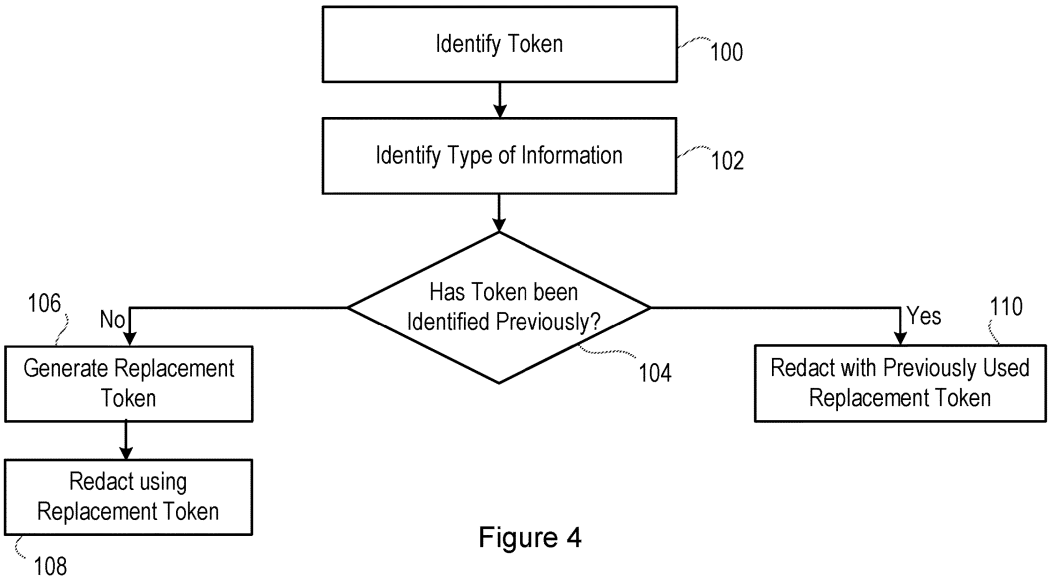

Figure 4

GB 2531713 A

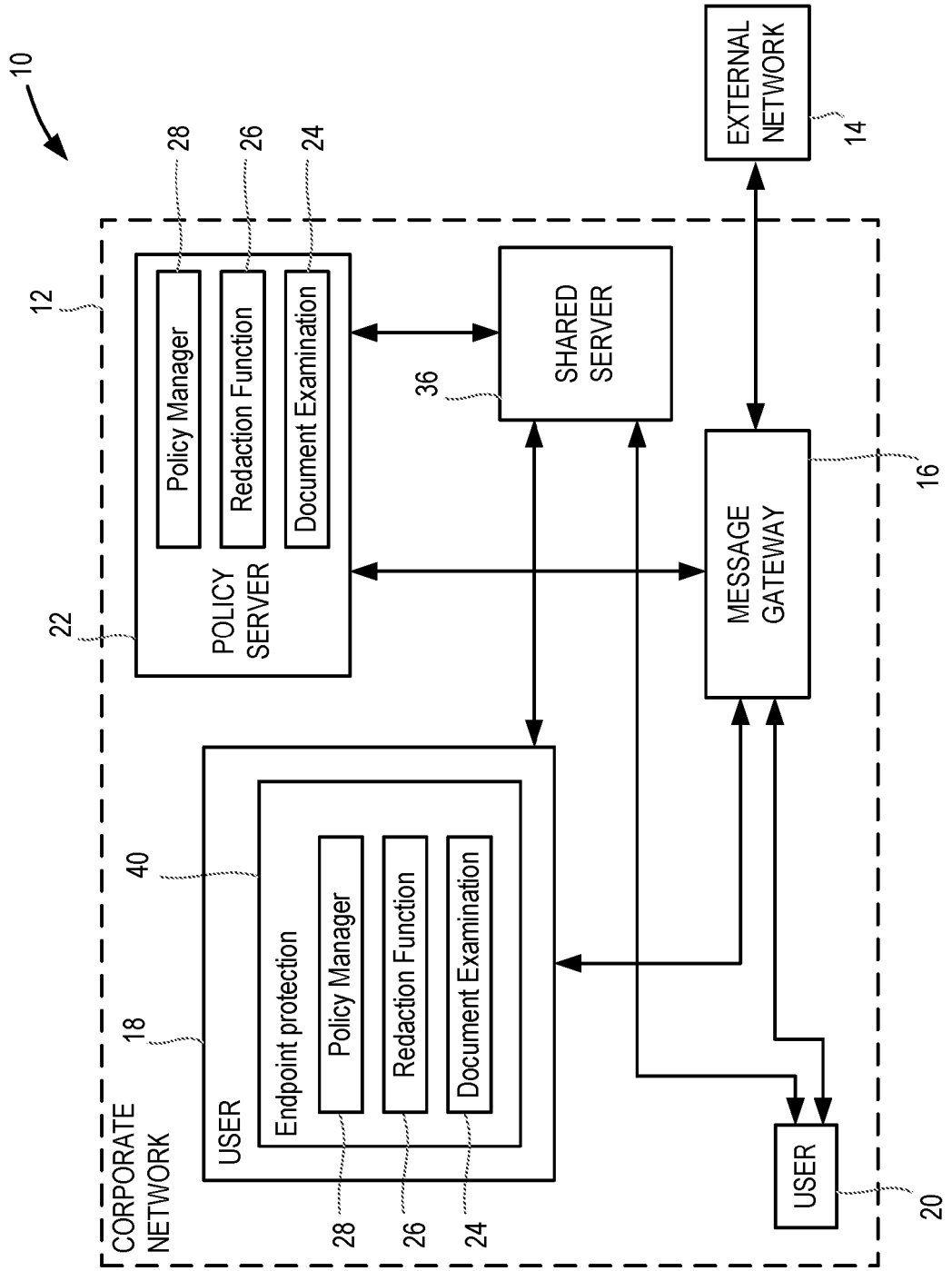
10
12
CORPORATE NETWORK
22
POLICY SERVER
Policy Manager
Redaction Function
Document Examination
28
26
24
36
SHARED SERVER
EXTERNAL NETWORK
14
MESSAGE GATEWAY
16
18
USER
40
Endpoint protection
Policy Manager
Redaction Function
Document Examination
28
26
24
USER
20

Figure 1

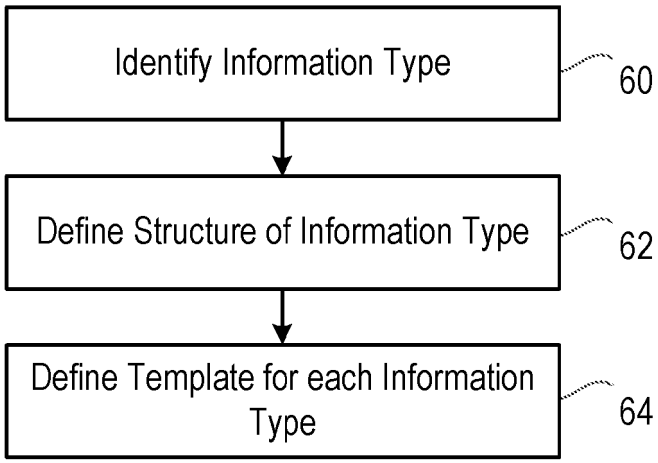
Identify Information Type
60
Define Structure of Information Type
62
Define Template for each Information Type
64

Figure 2

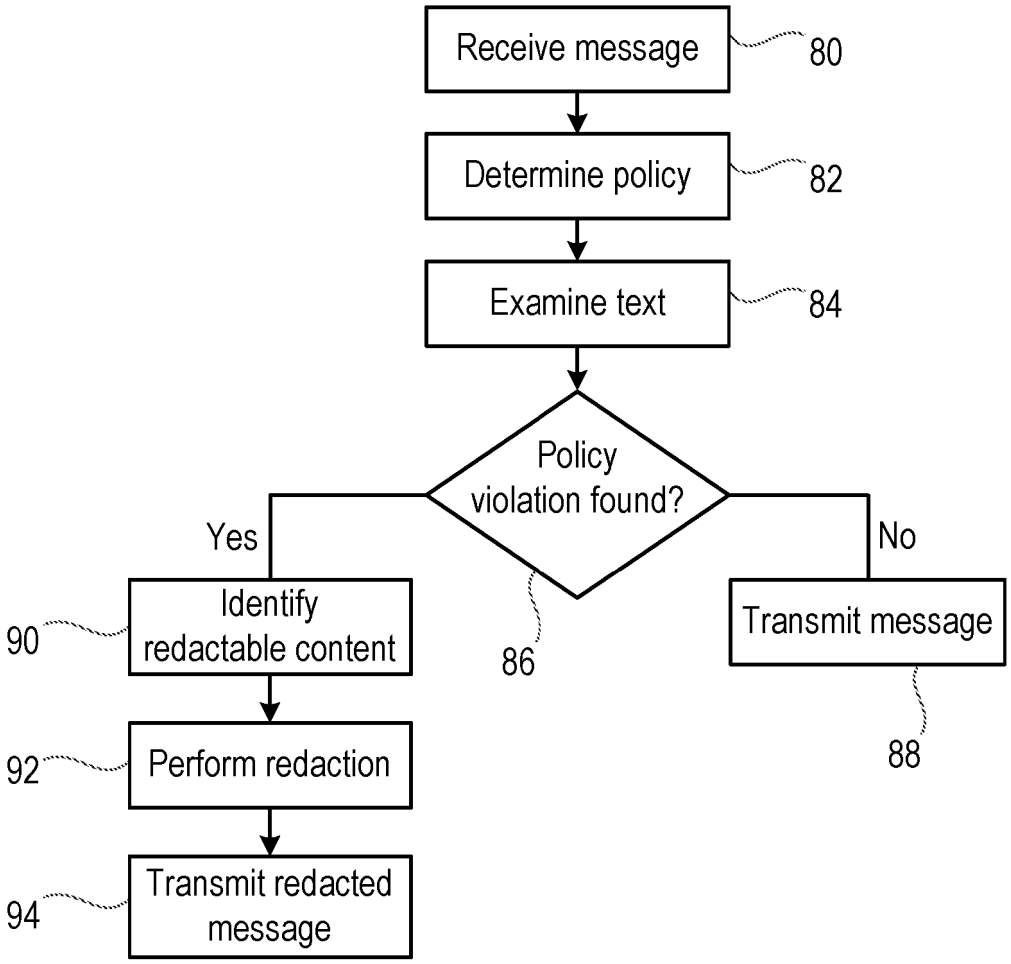
Receive message
80
Determine policy
82
Examine text
84
Policy violation found?
86
Yes
No
Identify redactable content
90
Perform redaction
92
Transmit redacted message
94
Transmit message
88

Figure 3

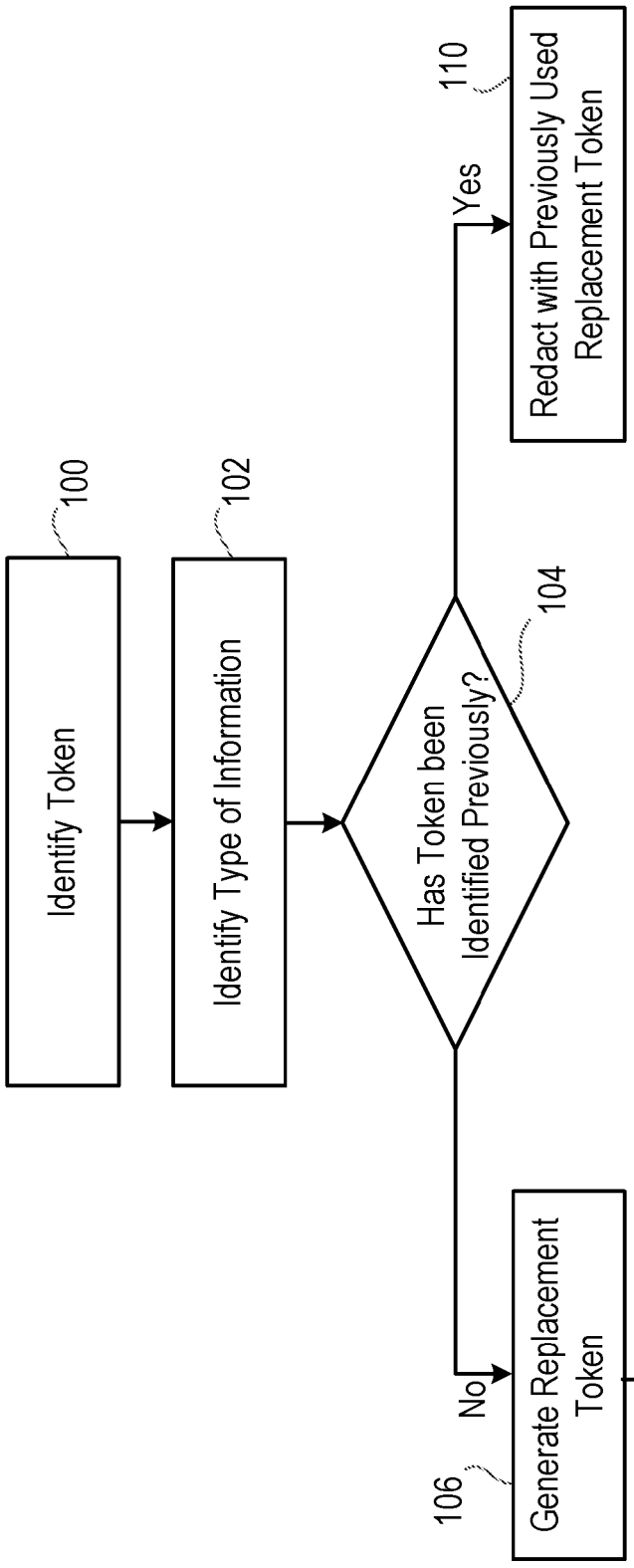
Identify Token
100
Identify Type of Information
102
Has Token been Identified Previously?
104
Yes
Redact with Previously Used Replacement Token
110
No
Generate Replacement Token
106
Redact using Replacement Token
108

Figure 4

AUTOMATIC CONSISTENT REDACTION OF TEXT

This invention relates to data handling, and in particular to a method and system for applying policies to such data, and for mitigating the effects of policy violations in textual content.

In electronic mail systems, it is common to apply policies to messages that are sent. That is, a system administrator is able to set various rules, and a policy manager in the system tests whether a message complies with those rules. If the message complies with the rules, then the message is sent to the intended destination. However, if the message does not comply with the rules, the policy can determine the action that is to be taken. Similarly, such policies can also be applied to other information transfers, such as file transfers, uploading and downloading information to and from the Internet, and retrieving web pages.

For example, the action that is taken in the event of a policy violation might be discarding the message, quarantining the message and sending a warning to the sender and/or intended recipient of the message, or the like.

It is also known that, in the case of textual documents, redaction can be applied to the document, so that sensitive content is removed. For example, when a document contains personal information, such as customer names, credit card numbers, or the like, that information can be removed (either manually or automatically) before the document is released.

In some situations, it is desirable to remove sensitive information from a document, while keeping the document in a form that allows it to be understood to a limited extent.

According to a first aspect of the present invention, there is provided a method of automatic redaction of a file comprising:

defining at least one type of potentially sensitive information which may exist in said file;

defining a respective template corresponding to each type of sensitive information for generating replacement tokens, wherein each template has:

a predefined structure,

a starting iteration value, and

a mechanism for modifying the starting iteration value of the template to produce subsequent iteration values;

identifying a first token within the file as being an example of one type of potentially sensitive information;

generating a replacement token by using a first iteration value of the corresponding template; and

replacing said first token with said replacement token; and,

if a subsequent token is identified as another example of the same type of potentially sensitive information;

generating a second replacement token by using a subsequent iteration value of the corresponding template, and

replacing said subsequent token with said second replacement token; wherein

if a token is identified multiple times within said file, the token is replaced with the same replacement token on each occasion that it is identified.

The method may comprise using the iteration values as replacement tokens, or may comprise using the iteration values to retrieve respective stored replacement tokens.

The mechanism for modifying the starting iteration value of the template to produce subsequent iteration values may comprise incrementing a numerical value, or may comprise using a pseudorandom method.

For a type of potentially sensitive information comprising structured information, the predefined structure of the respective template may match the structure of the structured information.

The type of potentially sensitive information may comprise system information, or may comprise personal information.

The method of redaction may be performed as part of a policy enforcement system, in which it is determined whether a file may be transmitted from one device to another. For example, the policy may determine whether a particular file may be uploaded to a remote device (such as a web server), downloaded from a remote device (such as a web server), or transmitted as an email between two devices. If it is found that the file may not be transmitted in accordance with the policy, the method of redaction may be performed, such that the file may then be transmitted.

According to a second aspect of the invention, there is provided a computer program product, comprising instructions for causing a programmed device to operate in accordance with the method of the first aspect. The computer program product may be for use in association with a network policy server, or may be for use in association with an endpoint protection device.

This has the advantage that redaction can be performed in a fully automated manner in order to remove the sensitive information, but that the document still has some meaning because the sensitive information is replaced on a consistent basis by alternative information.

According to a second aspect of the present invention, there is provided a computer program product, comprising instructions for performing the method of the first aspect.

For a better understanding of the present invention, and to show how it may be put into effect reference will now be made, by way of example only, to the accompanying drawings, in which:-

Figure 1 is a schematic diagram of a computer network in accordance with an aspect of the present invention;

Figure 2 is a flow chart illustrating a method in accordance with an aspect of the invention;

Figure 3 is a flow chart illustrating a further method in accordance with an aspect of the invention; and

Figure 4 is a flow chart illustrating a further method in accordance with an aspect of the invention.

Figure 1 shows a part of a computer network 10. Specifically, Figure 1 shows a part of a corporate network 12, having a connection to an external network 14. In one embodiment, the corporate network 12 may for example be a local area network (LAN)

within an organisation, but it will be appreciated that the methods described herein could be applied in other situations. For example, the method described herein could be implemented in a non-corporate network, such as within a service provider's network, or in secured wireless communications such as naval ship to shore. Similarly, the external network 14 could for example be the internet, but it will be appreciated that the methods described herein could be applied in other situations, for example in a cross-domain scenario, where there are two local area networks of different security levels (for example "secret" and "top secret") and email needs to pass between the networks in a controlled manner.

In the illustrated network, the corporate network 12 includes a message gateway 16, through which all electronic mail messages are passed. Figure 1 also shows users 18, 20 on the corporate network 12. Of course, there will be many more than two users in a typical network, but it is sufficient to show two such users to illustrate the operation of the method. The users 18, 20 may be connected to the corporate network through wireless connections, Ethernet connections, or any other suitable wired connection.

The users 18, 20 are able to send and receive electronic mail messages to and from each other, and to and from other users on the corporate network 12 that are not shown in Figure 1, and to and from other users on the external network 14. All such messages are passed through the message gateway 16.

Although only one such message gateway is shown in this example, it will be appreciated that typical corporate networks may have more complex structures. For example, there may be one message gateway for handling internal mail messages between users on the network, and a separate message gateway for handling external mail messages between a user on the network and a user on the external network. However, the illustrated architecture is sufficient for an explanation of the present invention.

Figure 1 also shows a policy server 22, connected to the message gateway 16. As will be understood, the policy server applies message policies to messages passing through the message gateway 16. In an architecture with multiple message gateways, there may be a policy server associated with each gateway, or there may be a single policy server associated with multiple message gateways.

As described in more detail below, the policy server 22 includes at least a document examination block 24, a redaction function 26, and a policy manager 28. In general terms, the purpose of the policy server 22 is to enforce policies that are set by, for example, a system administrator of the corporate network 12. For example, such policies may prohibit the sending of certain messages between certain users, or at least place conditions on the sending of such messages.

The network 12 may also include a shared server 36, such that a user can upload a file to the shared server, for later download by another user. The policy server 22 is also able to enforce policies relating to such file transfers. For example, such policies may prohibit the storage of certain files on a removable storage device, or may prohibit the transfer of certain files from such a device within a document management system, or may at least place conditions on such activities, with the files being identified based on their textual content.

Figure 1 also shows one user device 18 being provided with an endpoint protection product 40, of a type which is intended for deployment on a desktop or laptop computer, or the like. The endpoint protection product 40 is shown in Figure 1 as including similar functions to those included in the policy server 22, namely a document examination block 24, a redaction function 26, and a policy manager 28. In general terms, one purpose of the endpoint protection product 40 is to enforce policies relating to the transfer of information between the user 18 to and from removable storage devices such as optical storage discs (for example, CDs, DVDs, etc) and memory sticks. For example, such policies may prohibit the storage of certain files on a removable storage device, or may prohibit the transfer of certain files from such a device, or may at least place conditions on such activities, with the files being identified based on their textual content.

When a user device is provided with an endpoint protection product 40, this will operate whether the user device is operating within a network as shown in Figure 1, or operating outside such a network.

In the case of the policy server 22, the policies may for example relate to messages that contain specified file types as attachments, or that exceed a specified size. In this illustrated example, the policies relate to the information content of a message. More specifically, the policies may relate equally to the information content of the body of an

email message, to the information content of an attachment to an email message, and/or to the information content of the metadata of an email message such as the subject. Furthermore, policies may relate equally to different aspects of a structured format used within the email body or attachment including but not limited to the main body text, page headers and footers, footnotes, endnotes, annotations, textboxes and metadata.

In the case of the endpoint protection product 40, the policies may relate to the textual content of any file that the user seeks to transfer.

It will be appreciated that the method described herein can be performed in an endpoint protection device on a user device, even when that user device is not operating within a corporate network. Conversely, the method described herein can be performed in a policy server, even when the individual user devices do not include endpoint protection devices.

Where, as shown in Figure 1, the method is performed in an endpoint protection device on a user device, and that user device is operating within a corporate network in which the method described herein is also being performed in a policy server, the same policies may be applied in the endpoint protection device and in the policy server.

Figure 2 is a flow chart, illustrating an aspect of a process performed by software running on the policy server 22, in order to implement policies related to the content of electronic mail messages. The same process is performed by the software of the endpoint protection product 40. More generally, the method can be implemented by a computer program product, provided on any transitory or non-transitory medium, containing instructions for causing a programmed device to perform the method described herein.

Although the invention is described herein with reference to a specific example in which the process is applied in order to implement policies related to the content of electronic mail messages, the same or similar techniques can be used to implement policies relating to the content of web traffic, such as Internet based uploads and downloads, traffic to and from websites, including 'cloud collaboration' and 'social networking' sites. The same or similar techniques can be used more generally to implement policies that control any disclosure of information. For example, policies can be used to control the

transfer of information using file transfer methods, or instant messaging, and can also be used to control the transfer of information in document management and publishing systems. The term "file" is used herein to refer to any of these ways in which information can be stored and transferred, as well as the more traditional "file".

One specific example of a situation in which it is desirable to control any disclosure of information is when a message or a file contains system information, that is, information relating to an aspect of the network 12 itself. This system information includes, but is not limited to, usernames, system names (for example machine names), and system IP addresses. This information may for example be found in system log files stored within the network.

This system information can be used by cyber-attackers to compromise the security of the network, and so it is essential that the system information should not become available outside the organisation.

Another specific example is PII (Personally identifiable information) such as names, social security numbers, healthcare identifiers etc. This information can be used by cyber-attackers to compromise the identity of an individual (identity theft) and so it is essential that this information is secured and not available outside the organization, unless properly secured.

In both of these cases, there are also often requirements to anonymise the information but to do it in a way which retains some use for the recipient.

However, it is not suitable simply to block the sending of messages, or the transfer of files, that contain system information, because there exist situations in which it may be necessary for reports that contain system information to be sent to recipients who might be outside the organisation. For example, in the event of a fault in the network 12, it may be necessary to share the system log files with an external hardware or software supplier, in order to be able to diagnose and resolve the fault.

In that case, simply sending the system log files outside the organisation might provide an unacceptable security risk, while simply redacting the system information might make it impossible to investigate the operation of the system.

Thus, the system can be configured so that specific types of system information in a document, or of other information as described in more detail below, can be redacted in a way that maintains the intelligibility of the document.

In step 60 of the process illustrated in Figure 2, the type or types of relevant, potentially sensitive, information are identified. These types of information may be in the form of structured data elements, having a well-defined format, or may be in the form of unstructured data elements, which do not have such a well-defined format. Credit card numbers or IP addresses are examples of structured data elements, while personal names and addresses are examples of structured data elements.

For example, in the case where it is considered that it may be desirable to share system log files with an external hardware or software supplier, as described above, it may be noted that the system log files contain system information that should not become known to third parties, such as the network usernames and passwords, logon credentials, system directories, filenames, system names (for example machine names), and system IP addresses, amongst other things. These are thus identified as the types of potentially sensitive information that are applicable in this case.

It will be appreciated that this step of identifying the types of potentially sensitive information may be performed at the start of each process of examining a set of specific files. Alternatively, a library of types of potentially sensitive information may be generated in advance and used whenever applicable.

In step 62, each of these identified types of information is considered in turn, and, in the case of the structured data elements, the structure of the information is defined. For example, in the case of system IP addresses, it is known that IP addresses, at least in Internet Protocol Version 4 (IPv4), are made up of four 8-bit binary numbers. These are usually stored in a more easily readable form as four decimal numbers, each in the range from 0 to 255, separated by dots, for example 99.100.101.102.

The process then passes to step 64, in which a template is defined for each type of information. As described below, the template is used for generating replacement information when an example of the potentially sensitive information is identified in a file. In each case, the template has a suitable structure.

When the type of potentially sensitive information is in the form of structured data elements, the template has a corresponding structure. Thus, for example, in the case of system IP addresses that are stored in the more easily readable form described above, the template may be of the form [. . .], that is having four blank spaces that can contain three-digit decimal numbers, separated by dots.

When the type of potentially sensitive information appears in the form of unstructured data elements, the template has a suitable structure. For example, in the case of personal names, the template may have a structure of the form [Name], that is having the word “Name” and a space that can contain, for example, a three-digit number, or any other length number that is sufficient to provide the required number of different values..

The templates also include a starting iteration value, and a mechanism for modifying the starting iteration value to generate subsequent iteration values, so that all of the iteration values can be used as replacements when the potentially sensitive information appears in a file under consideration.

For example, in the case of the IP address mentioned above, where the template is of the form [. . .], having four blank spaces that can contain three-digit decimal numbers, separated by dots, the starting iteration value may be [0.0.0.0] with the simple iteration rule that the number is regarded as a twelve-digit number that is incremented by one each time, with each of the four three-digit numbers having a maximum value of 255. (Equivalently, the IP address can be regarded as a 32-bit binary number that is incremented by one each time and then converted to decimal.)

The result is that a sequence of iteration values [0.0.0.0], [0.0.0.1], [0.0.0.2], …, [0.0.0.255], [0.0.1.0] [0.0.1.1], …, [0.0.1.255], [0.0.2.0], [0.0.2.1] can be generated by following the iteration rule.

In the case of names, where the template has a structure of the form [Name], that is having the word “Name” and a space that can contain a three-digit number, the starting iteration value could be [Name 001] with the rule being to increment the number by 1 on each occasion. Alternatively, the structure could be of the form [Name], that is having the word “Name” and a space that can contain four letters, in which case the

starting value could be [Name aaaa], and a rule could define how the characters are incremented.

As a further alternative, the iteration values that are generated may not be used as replacement tokens directly, but may be used to produce replacement tokens. For example, the structure could be of the form [Name], where the space can be filled by a name chosen from a stored list. That is, the starting iteration value could be [Name {name_list_1}], where {name_list_1} causes the first name of the stored list to be retrieved, and the rule is that the number is incremented each time so that a subsequent name of the stored list is retrieved. Thus, the iteration value is used to obtain the replacement token.

In either case, different starting iteration values could be chosen and/or more complex iteration rules could be applied, so that the iteration values that are generated are less distinctive. For example, a pseudorandom method could be used to produce subsequent iteration values.

In a similar manner, the potentially sensitive information may be in the form of images. For example, the presence of company logos or the like in a document may provide confidential information to a reader. In that case, the template may have a structure of the form [Image], that is having the word “Image” and a space that can contain an image chosen from a stored list. That is, the starting iteration value could be [Image {image_list_1}], where {image_list_1} causes the first image of the stored list to be retrieved, and the rule is that the number is incremented each time so that a subsequent image of the stored list is retrieved.

Having performed the method shown in Figure 2, it is now possible to deploy the policy server to prevent the unwanted transmission of sensitive information.

Figure 3 shows an aspect of a process performed by software running on the policy server 22, in order to implement policies related to the content of electronic mail messages. The same process is performed by the software of the endpoint protection product 40. More generally, the method can be implemented by a computer program product, provided on any transitory or non-transitory medium, containing instructions for causing a programmed device to perform the method described herein.

In step 80, a message is received, having some textual content, either in the body of the message, and/or in an attachment to the message (including in structural constructs such as page headers and footers, footnotes and endnotes of the message or its attachment), and/or in the message metadata.

In step 82, it is determined which policy or policies apply to the message. For example, the policy manager may have been configured such that messages sent between any member of a first group of users and any member of a second group of users may not contain content of a certain type, while messages sent between any member of a third group of users and any member of a fourth group of users may not contain content of a different type. Purely as an example, a first policy may specify that messages sent from members of a company's finance team to members of the company's marketing team may not contain any payment card numbers (i.e. sixteen digit numbers, especially when divided into four blocks of four digits); a second policy may specify that messages sent from members of the company's engineering team to recipients outside the company may not refer to the name of a secret internal project; and a third policy may specify that messages sent from any user must not contain profanity.

It is mentioned here for the sake of completeness that similar tests are applied in the case where the received text forms part of some content that is being downloaded from a website, in the use of a web browser program for example, or forms part of some content that is being uploaded to a website. The policy or policies that apply to the text will then typically be based on the user who is requesting the transfer, possibly amongst other factors. More generally, in this example the text may be received as part of an upload to, or a download from, any external network.

More generally, it is known that policies may attempt to deal with issues such as: controlling offensive material; controlling the disclosure of intellectual property; and controlling the disclosure of sensitive information including Personal Identifiable Information (PII), Payment Card Information (PCI) and Corporate Infrastructure Information (CII) such as usernames, IP addresses, machine names and URLs.

Thus, in step 82, it is determined, for example based on the identities of the sender and recipient (but potentially also based on other information) which policies apply to the received message.

In step 84, the relevant textual content is examined, to determine whether it complies with the applicable policies.

First, the relevant text is identified. As mentioned above, the policy may for example be set such that the text in the body of the message is examined, that the text in any attachment to the message is examined and/or the text within the message metadata is examined. This may involve the identification of the format of any attachments and performing any decomposition such as extracting files from within an archive and continuing this identification/decomposition process in a recursive manner. The identification of the format and examination of structured formats for the presence of aspects such as page headers and footers are used to identify text that is relevant to the policy. For example, a policy may specify that specific text should not appear in the page footer of a document and the relevant text could be found in the page footer of a word processing document which is within a ZIP archive that has been attached to an email message.

Having identified the relevant text from the message, in step 84 the relevant textual content is examined in a process of lexical analysis to determine whether the information is acceptable, that is, conforms to a policy.

For example, this may be done by tokenising the text (that is, dividing the text into smaller components, such as words), and then searching the tokens for specific tokens or combinations of tokens. Combinations could be simple sequences that form a phrase, or token sequences that are combined with logical operations such as “AND” and “OR” and positional/proximity operations such as “BEFORE”, “AFTER” and “NEAR”. This search construct is known as an expression.

Using a technique known as Text Entity Extraction it is also possible to identify higher order information within the textual content; for example, names, dates, Credit Card Numbers, National Insurance Numbers and Social Security Numbers; by examining the tokens. Text Entities such as these can also be used in place of tokens within the expressions.

Similarly, regular expressions can take the place of tokens within a search.

When dealing with sensitive information (such as Personal Identifiable Information, Payment Card Information, or Corporate Infrastructure Information as discussed above), it may be that any match of a Social Security Number, Credit Card Number or IP address is all that is needed to determinate that the policy has been violated.

When dealing with offensive material, the presence of a single token or combination of tokens might not be enough for the text as a whole to be considered unacceptable, but a combination of tokens repeated enough times, or the presence of certain tokens in the presence of other tokens might be enough for the text to be considered unacceptable.

As an example, a policy may be defined in terms of an expression list that consists of a set of entries, each of which consists of an expression with an associated weighting. The weighting can have a positive integer value or a special violation value. A threshold value is also set.

An initial score is set to zero, and the textual content is tokenised and any Text Entities are identified. The tokens and Text Entities are then searched to determine, for each expression in the expression list, whether it matches the textual content. When a match is found, the weighting for the relevant expression is added to the score. If the weighting is the special violation value, then the score is set to the threshold value of the expression list.

In step 86 of the process shown in Figure 3, after all of the expressions have been used as the basis for the search, the final score is examined. If the score is greater than or equal to the threshold value, then it is determined that the policy has been violated. In one example, the policy is set such that the presence of any item of potentially sensitive information means that the policy has been violated.

If it is found that the policy has not been violated, then the process passes to step 88, and the message is transmitted as intended by the sender.

However, if it is found in step 86 that transmitting the message would violate the policy, the process passes to step 90. The intention here is to mitigate the policy violation, such that the message can still be sent.

Thus, in step 90, the redactable text in the message is identified. Then, the process passes to step 92, in which an attempt is made to mitigate every match of every expression in the expression list. A match is mitigated by redacting each of the tokens and Text Entities that form the match. When a Text Entity is formed from a number of tokens, it can be redacted by redacting the constituent tokens.

The expression list may contain, for some or all of its entries, instructions to indicate what form of redaction should take place in order to mitigate any violation caused by the use of the relevant expression.

The form that the redaction should take is dependent upon the nature of the text and the context in which it is being used; it is therefore advantageous that the redaction process can be controlled via the policy. For example, when a text contains a credit card number and an associated expiry date, it may be appropriate in some business contexts that the credit card number is redacted but the associated expiry date is not. One way in which this can be accomplished is to embed a unary operator within the expression, which marks the following sub-expression such that any matches to that sub-expression will be redacted but matches to other sub-expressions will not. For example, an expression of the form ".REDACT. .TextEntity=CreditCardNumber. .NEAR. .TextEntity=Date." would result in any credit card numbers being redacted but any dates near them would not be redacted. Alternatively, an expression of the form ".REDACT. (.TextEntity=CreditCardNumber. .NEAR. .TextEntity=Date.)" would result in both the credit card numbers and any dates near to them being redacted.

In some cases, the redaction may wholly or partly replace the potentially sensitive information with non-specific characters. For example, every IP address may be replaced with a series of asterisks ***.***.***.***.

In certain methods, however, where the potentially sensitive information is of a type that has been previously identified as described with reference to Figure 2, the step of performing the redaction is performed as shown in Figure 4.

Specifically, in step 100, the potentially sensitive information is identified, by identifying a relevant token in the file. In step 102, the type of information is identified, from amongst the various types of potentially sensitive information identified in step 60 of the

process shown in Figure 2. In the case of structured information, such as credit card numbers, IP addresses and the like, these may be identified by the presence of the relevant structure in the data. In the case of unstructured information such as names and addresses, these can be identified using named-entity recognition techniques.

In step 104, it is determined whether that specific item of potentially sensitive information has been detected previously within the same file, or document. If not, the process passes to step 106. In step 106, the replacement token is generated.

The replacement token is generated using the predefined template that corresponds to that type of potentially sensitive information. Thus, starting from the structure of the template, the relevant mechanism or rule is followed to determine the next previously unused iteration value.

As described above, the subsequent iteration value can be used directly as the replacement token, or can be used to link to a previously stored replacement token.

Having generated the replacement token, the process passes to step 108, in which the item of potentially sensitive information in the file under examination is replaced by the replacement token generated in step 106.

If it is determined in step 104 that the specific item of potentially sensitive information has previously been detected within the same file, or document, the process passes to step 110.

In that event, the item of potentially sensitive information is replaced by the same replacement token that was generated when that same item of potentially sensitive information was first identified in the file. Thus, in this situation, the item of information is redacted by replacing it with the same item of replacement information that was used to replace that item of information on its previous occurrence or occurrences in the current document.

Thus, if a particular identical token is identified multiple times within the text file, the identified token is replaced consistently with the same replacement token on each occasion. If a second token of the same type (for example, with the same predefined

structure) is identified, it is replaced by a second replacement token on each occasion that it is identified.

The effect of this is that the intelligibility of the document is maintained, because repeated references to the same piece of system information are replaced consistently by repeated uses of the same piece of replacement information.

For example, a text reading:
"Dear Mr Andrews,
Your credit card numbers 5674 2354 1278 9081 and 3256 2145 6783 7857 have been blocked due to fraudulent activity we have detected. In particular, credit card number 5674 2354 1278 9081 was used simultaneously in Japan and Brazil."

could be replaced by a redacted text reading:
"Dear Name-0001,
Your credit card numbers 0000 0000 0000 0001 and 0000 0000 0000 0002 have been block due to fraudulent activity we have detected. In particular, credit card number 0000 0000 0000 0001 was used simultaneously in Japan and Brazil."

Once the possible redactions have been performed, the process passes to step 94 of the process shown in Figure 3, and the message is transmitted to the recipient intended by the sender, with the text resulting from the redaction.

In an alternative embodiment, the text resulting from the redaction in step 92 is re-examined, for example with the re-examination taking the same form as the examination of the text performed in step 84 described above, in order to check that the policy would not be violated by the transmission of the redacted message. If it is then found that transmitting even the redacted message would violate the policy, a disposal is performed in accordance with the policy. For example, the policy may state that the message should simply be discarded, or may state that the message may not be transmitted but that a notification should instead be sent to the sender and/or intended recipient of the message. Where the policy violation arises because of the textual content of an attachment to the message, the policy may allow the message to be sent without the relevant attachment.

Although the method described herein has been explained primarily with reference to the redaction of structured information, in the form of Payment Card Information, it is equally applicable to the redaction of other structured information, such as Personal Identifiable Information (PII), for example Social Security numbers, Passport numbers, financial data or the like, or to Computer Infrastructure Information such as IP addresses, or to any other structured information.

In addition, although the method described herein has been explained primarily with reference to the redaction of structured information, it can also be used for the redaction of unstructured information, such as names, addresses, or the like.

There is thus disclosed a method of policy enforcement that allows for improved results, allowing redaction to take place while maintaining the intelligibility of the document.

CLAIMS

1. A method of automatic redaction of a file comprising:

defining at least one type of potentially sensitive information which may exist in said file;

defining a respective template corresponding to each type of sensitive information for generating replacement tokens, wherein each template has:

a predefined structure,

a starting iteration value, and

a mechanism for modifying the starting iteration value of the template to produce subsequent iteration values;

identifying a first token within the file as being an example of one type of potentially sensitive information;

generating a replacement token by using a first iteration value of the corresponding template; and

replacing said first token with said replacement token; and,

if a subsequent token is identified as another example of the same type of potentially sensitive information;

generating a second replacement token by using a subsequent iteration value of the corresponding template, and

replacing said subsequent token with said second replacement token; wherein

if a token is identified multiple times within said file, the token is replaced with the same replacement token on each occasion that it is identified.

2. A method as claimed in claim 1, comprising using the iteration values as replacement tokens.

3. A method a claimed in claim 1, comprising using the iteration values to retrieve respective stored replacement tokens.

4. A method as claimed in one of claims 1 to 3, wherein the mechanism for modifying the starting iteration value of the template to produce subsequent iteration values comprises incrementing a numerical value.

5. A method as claimed in one of claims 1 to 3, wherein the mechanism for modifying the starting iteration value of the template to produce subsequent iteration values comprises using a pseudorandom method.

6. A method as claimed in one of claims 1 to 5, wherein, for a type of potentially sensitive information comprising structured information, the predefined structure of the respective template matches the structure of the structured information.

7. A method as claimed in one of claims 1 to 6, wherein the type of potentially sensitive information comprises system information.

8. A method as claimed in one of claims 1 to 7, wherein the type of potentially sensitive information comprises personal information.

9. A computer program product, comprising instructions for causing a programmed device to operate in accordance with the method of any of claims 1 to 8.

10. A computer program product as claimed in claim 9, for use in association with a network policy server.

11. A computer program product as claimed in claim 9, for use in association with an endpoint protection device.

Intellectual Property Office

Application No:	GB1418961.7	**Examiner:**	Mr Jim Calvert
Claims searched:	1-11	**Date of search:**	31 March 2015

Patents Act 1977: Search Report under Section 17

Documents considered to be relevant:

Category	Relevant to claims	Identity of document and passage or figure of particular relevance
X,E	1-11	GB2518433 A (CLEARSWIFT) See whole document
A	1-11	WO2011/059510 A1 (THOMSON REUTERS) See e.g. paras 0030-34
A	1-11	US2009/222527 A1 (ARCONATI ET AL) See e.g. para 0190

Categories:

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.

Field of Search:

Search of GB, EP, WO & US patent documents classified in the following areas of the UKC^{X} :

Worldwide search of patent documents classified in the following areas of the IPC

G06F; G06K

The following online and other databases have been used in the preparation of this search report

WPI, EPODOC, TXTE, Internet

International Classification:

Subclass	Subgroup	Valid From
G06F	0021/62	01/01/2013