(19) DANMARK

(10) **DK/EP 3047370 T3**

(12) Oversættelse af europæisk patentskrift

Patent- og Varemærkestyrelsen

(51) Int.Cl.: *G 06 F 21/51 (2013.01)* *G 06 F 21/57 (2013.01)* *G 06 F 21/62 (2013.01)*
G 06 Q 10/06 (2012.01) *G 06 Q 30/02 (2012.01)* *H 04 L 29/06 (2006.01)*

(45) Oversættelsen bekendtgjort den: **2020-02-03**

(80) Dato for Den Europæiske Patentmyndigheds bekendtgørelse om meddelelse af patentet: **2019-10-30**

(86) Europæisk ansøgning nr.: **14845878.9**

(86) Europæisk indleveringsdag: **2014-09-12**

(87) Den europæiske ansøgnings publiceringsdag: **2016-07-27**

(86) International ansøgning nr.: **US2014055494**

(87) Internationalt publikationsnr.: **WO2015041955**

(30) Prioritet: **2013-09-19 US 201361879909 P** **2014-09-05 US 201414478714**

(84) Designerede stater: **AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR**

(73) Patenthaver: **LiveRamp, Inc., 225 Bush Street, 17th Floor, San Francisco, CA 94104, USA**

(72) Opfinder: **PALAN, Vivek, 155 Jackson Street 1901, San Francisco, CA 94111, USA**
OWEN, Paul, 385 NW 17th Street, Bend, OR 97701, USA
LEDO, Frank, 3255 California Street, Berkeley, CA 94703, USA
JOLITZ, Ben, 22570 Citation Drive, Los Gatos, CA 95033, USA

(74) Fuldmægtig i Danmark: **Budde Schou A/S, Hausergade 3, 1128 København K, Danmark**

(54) Benævnelse: **Fremgangsmåde og system til udledning af risikoen for datalækage fra tredieparts tags**

(56) Fremdragne publikationer:
US-A1- 2010 186 088
US-A1- 2010 281 536
US-A1- 2011 289 582
US-A1- 2013 212 638

DESCRIPTION

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. provisional patent application number 61/879,909, filed on September 19, 2013, and entitled "Method and System for Inferring Risk of Data Leakage from Third-Party Tags."

STATEMENT REGARDING FEDERALLY SPONSORED RESEARCH OR DEVELOPMENT

[0002] Not applicable.

BACKGROUND OF THE INVENTION

[0003] In recent years, website advertisers and publishers have begun adding tracking code from third-party websites in order to better understand who their users are, and further to monetize these insights into their customers. Monetization can take numerous forms, including the placement of display advertisements on a content publisher's own website, or by targeted advertising using data aggregated from a number of websites. This data is normally exchanged and sold by means of the real-time or offline synchronization of anonymized cookie information. The types of code used for this purpose may include web "pixels," that is, objects embedded in a web page, and JavaScript tags. All such code may generally be referred to as third-party tags.

[0004] As the digital advertising industry has grown, so have the number of third-party tags placed on any given site. As a representative example, the television network website cbs.com has at a certain point had at least eleven third-party tags firing on the homepage during a web browser's visit, with many more on other pages within the overall site. Since a website may partner with or terminate its relationships with one or more of these partners over time, the large number of third-party tags from different companies means that keeping each tag up to date has become difficult.

[0005] Because of the widespread use of third-party tags, another issue that has arisen is the protection of a website's user data from unauthorized use or malicious activity, which is commonly known as data "leakage." Even when the online content publisher has a relationship with a third-party tracking company, the publisher may have restrictions on how its user's data can be utilized. Third-party tag requests can be forwarded from third parties to their secondary partners and so on, without a website owner's knowledge or permission, or even the knowledge that such forwarding is taking place. Malicious activity using third-party tags may include extraction of data from a web page; maliciously modifying content of a web page;

performing an activity that is intended to harm the end user or the website owner; the presence of a Trojan horse that is automatically and surreptitiously downloaded by visiting the webpage; and the hiding of website re-direct activities in order to avoid the user realizing that the web browser has been redirected to an unsafe or potentially unsafe site.

[0006] Consumer web browsers have become extremely complex in order to process and display modern web pages, which may be composed of many independent pieces belong to a number of different parties and hosted at different servers with different IP addresses. Any web browser must be able to handle a request for a single web document and subsequent requests for external resources required by that document. Some resources are programs, such as those written in the webpage scripting language JavaScript, which will run within the context of the web browser and terminate only when the user closes the browser or navigates to a different document. Given this complexity and the resulting difficulty of understanding what occurs as a user browses the web, most users are technically unable to investigate this issue, and simply expect that first party and third party players behave appropriately, with potentially damaging results when that expectation is violated.

[0007] Many consumers and businesses wish to take advantage of increasingly accurate metrics, analytics and targeted advertising. The data collected to enable such offerings has in turn become extremely valuable. Due to a lack of industry oversight and comprehensive safeguards, unscrupulous companies may use third-party tags to forward information about a consumer visiting a website without obtaining permission from the client website (such as a content provider) with which they are partnered. Furthermore, malicious entities may want to masquerade as legitimate entities in hopes of harvesting sensitive information about a consumer. Third-party tags may facilitate unsafe or unexpected behavior, which may include the loading of off-site scripts, collection of sensitive data, participation in denial-of-service (DoS) attacks, or making contact with questionable domains for undisclosed tracking of the consumer's online behavior.

[0008] Due to the nature of JavaScript in particular, legitimate and malicious third-party tags are difficult to analyze piecemeal, forcing users and the publishers who provide content to these users to choose between security and convenience. To make matters worse, some third-party JavaScript tags will load additional third-party JavaScript tags in a chain of scripted calls, reducing the practicality of human directed analysis significantly. It may be seen that it would be highly desirable to both advertisers and content publishers to have a clear understanding of what actions these third-party tags, such as JavaScript tags and pixel tags, are taking on their websites, in order that they may take appropriate action in order to protect the privacy of their users. Consumers likewise would find this understanding highly desirable in order to safeguard their privacy and to allow them to navigate various web pages without fear that their information is being misused or misdirected. US2010186088 discloses a method and system for automated identification of phishing, phony and malicious web sites. US 2010281536 discloses a method and system for determining a phish probability scoring model.

[0009] References mentioned in this background section are not admitted to be prior art with

respect to the present invention.

BRIEF SUMMARY OF THE INVENTION

[0010] The present invention is an apparatus and method to determine if data leakage is occurring through a particular web domain. In various embodiments, the invention mimics common web browsers and visits client websites (such as content publishers with third-party advertisements) in a manner that appears to the website software to be similar to a typical user. Unlike the standard web browser, however, the software records requests for information and programmatic operation in order to construct a threat score. The system
presents the appearance of a consumer using a web browser in order to monitor and observe activity taking place with respect to JavaScript tags, pixel tags, and the like. The system creates a taxonomy of first- and third-party requests in order to construct a hierarchical model that ascribes provenance to each resource. The results of the analysis are, in certain embodiments, made available as a cumulative threat score. The threat scores are calculated based upon various attributes that are assigned to each third-party tag identified.

[0011] The various embodiments of the invention may be seen to provide transparency, such that the website owner is aware of the likelihood that a partner is dealing with other companies in a suspect fashion. It facilitates the privacy compliance for a website owner, such that the website owner can block or terminate partners that do not comply with its own privacy standards or applicable laws or regulations. It also allows for better monetization of the audience for a content publisher by avoiding the user being exposed to random advertisements, such that the user will be more likely to respond to relevant advertisements on the website itself, thereby increasing conversion rates. The website visitor will also be more likely to trust the website with his or her own data. The website owner can thus be informed of the actions of third parties interacting with its website; evaluate the relative safety or threat presented by third-party tags on its website; unmask previously hidden third parties that may be operating without the knowledge of the client; and determine performance impacts to its website caused by the actions of third-party tags.

[0012] These and other features, objects and advantages of the present invention will become better understood from a consideration of the following detailed description of certain embodiments and appended claims in conjunction with the drawings as described following:

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0013]

Fig. 1 is a diagram showing a networked system according to certain embodiments of the present invention.

Fig. 2 is a flow chart depicting a process for identifying data leakage according to certain embodiments of the present invention.

Fig. 3 is a diagram showing a computing device for implementing certain embodiments of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

[0014] Before the present invention is described in further detail, it should be understood that the invention is not limited to the particular embodiments described, and that the terms used in describing the particular embodiments are for the purpose of describing those particular embodiments only, and are not intended to be limiting, since the scope of the present invention will be limited only by the claims.

[0015] In various embodiments, the invention is a method and system for inferring whether data leakage is occurring on a website with third-party tags, including but not limited to pixels and JavaScript tags. It is important to note that the mere presence of a third-party tag, and forwarding (or "piggybacking") that results when the code calls another server, is not itself an indication of data leakage. There may be legitimate reasons to forward HTTP requests to multiple partners. The invention analyzes a webpage in a manner that appears to the website and its partners as if the webpage is interacting with an ordinary consumer's web browser, but the analysis is used to determine a data leakage threat score, which is then used to determine whether leakage is likely to be occurring at the website.

[0016] Referring now to Fig. 1, a system of interconnected computing devices is shown as may, for example, be various types of devices connected to each other across the Internet. Although each is generally connected across the Internet, the actual communications taking place between individual components is shown by arrows in Fig. 1 for clarity. Marketing services provider (MSP) server 10 hosts the software that, together with computer hardware, results in specially programmed computer hardware that implements the method and system described herein. Specifically, MSP server 10 includes routines 11 to mimic a standard computer browser in order to analyze websites for data leakage, as will be described following. MSP server 10 communicates with a content publisher server 12, which hosts the website of interest. This website may be, for example, a news, entertainment, or other information site, a retail site, or any of many other sites that would be of interest to various types of consumers or individuals. This website is designed to be accessed through consumer computer 14, which runs web browser 16 so that a consumer may interact with the webpage hosted at publisher server 12. Consumer computer 14 may be a desktop computer, a laptop computer, a smartphone, a tablet, or any other electronic device that allows for a connection to a website or other web content as hosted by publisher server 12. Third-party tags on the website hosted at publisher server 12 redirect to resource servers 18 and 20. Since third-party tags may further

redirect to subsequent third-party tags, in this particular example resource server 20 redirects to resource servers 22 and 24, which are thus two steps removed from the consumer accessing publisher server 12 and the website hosted at publisher server 12 through consumer computer 14. This configuration of Fig. 1 is only one example of many possible variations, with any number of publisher servers, consumer computers, and resource servers possible in various embodiments and configurations, with any number of levels of resource servers as material may be redirected from one to the other.

[0017] Referring now to Fig. 2, the process for determining a threat score-and thus identifying the likelihood of data leakage-may be described using the system of Fig. 1. At step 30, the software executing at MSP server 10 accesses the website at publisher server 12. The software then analyzes whether there are third-party tags on this website at step 32. The software is designed to mimic a standard web browser. If there are no such tags, then the website may be determined to be no threat at step 48, this result is returned by the system, and processing ends.

[0018] If there are third-party tags, then at step 36 it is determined whether any of the domain names to which the HTTP request is being forwarded (piggybacked) by the third-party tags are known data leakers. A list of known data leaker domain names is maintained in known leaker table 34, which may be part of or in communication with a data storage device in communication with MSP server 10. If any of these domain names are found in leaker table 34, then the website is determined to be a threat at step 46, this result is returned by the system, and processing ends.

[0019] If at step 36 it is determined that none of the domain names to which the third-party tags are forwarded are in the list of known leakers in leaker table 34, then processing moves to step 40 to calculate individual threat scores for each of the redirected domain names. Each such domain name is measured against certain attributes that correlate to data leakage sources. For each attribute checked, if such attribute is found for the domain name then a threat score is assigned based on a value assigned to that attribute in attribute table 38, which is a part of or in communication with MSP server 12. For example, one attribute that may be checked for is whether the third-party tag forwards to a server with a registration date that is less than thirty days old. Short registration periods could be evidence of a service that has been forced to keep changing hosting services because of fraudulent or disreputable activity. Another example is in indication that third-party tag opens a connection to another site that performs data "scraping" to capture material that is intended to be entered at the original site, such as personal information of a consumer. A third example is a call to open a "pop-up" window to redirect to a different domain.

[0020] Each attribute is associated with a particular threat score. Multiple attributes may apply to a single third-party tag. Each third-party tag is analyzed in the manner described herein to determine which attributes match from attribute table 38, and the threat score for each third-party tag is determined as the sum of each of the attribute threat scores for that domain. As shown in Fig. 1, there may be multiple levels of third-party tags from different resource servers,

and each tag at each level is analyzed in this manner to calculate a threat score for that attribute. If a third-party tag leads to another third-party tag with a higher threat score, then this higher threat score is assigned to each third-party tag up the chain.

[0021] In a particular example of multiple levels of third-party tags being analyzed at step 40, consider a website that has a base third-party tag 1 that directs to a forwarded tag 2 and forwarded tag 3, forwarded tag 2 redirects to a forwarded tag 4, and forwarded tag 4 redirects to a forwarded tag 5. Further consider that there are attributes A, B, C, D, and E that are associated with the threat scores 2, 3, 4, 5, and 7, respectively, higher numbers indicating a more significant threat to privacy or of malicious activity. The following threat scores can then be calculated based on the attributes of each third-party tag as follows, with the threat score in parentheses representing a score inherited from a third-party tag further down in the chain that has a higher threat score:

Tag	Attributes	Threat Score
1	A	2(15)
2	B, D, E	15
3	B, D	8
4	B, D	8(9)
5	A, B, C	9

Because the attributes and associated threat scores are maintained in attribute table 38, the threat assigned to each attribute may be easily updated as the inferred threat associated with each of the attributes is better understood through experience using the various embodiments of the invention.

[0022] Once the individual threat scores are calculated at step 40, then processing moves to step 42 to calculate a cumulative threat score associated with the website at publisher server 12. This process may be as simple as summing each of the individual threat scores for each third-party tag analyzed, or assigning the highest threat score from among the third-party tag individual threat scores. At decision step 44, it is determined whether the cumulative threat score exceeds a certain threshold. For example, this threshold may be set as the top 10% of the websites that have been analyzed using the system executing at MSP server 10. If the threshold is met, then a threat is identified at step 46, this result is returned, and processing ends. If the threshold is not met, then the domain is identified as not being a threat at step 48, this result is returned, and processing ends.

[0023] In certain embodiments, the marketing services provider may operate multiple MSP servers 10, each such server being located in a geographically remote location from the other servers. This may be advantageous since the behavior of certain websites may change, and the resulting analysis and threat score calculation thus may change, depending upon the geographic location from which the website is accessed. By utilizing multiple MSP servers 10 in geographically remote locations, the marketing services provider can better analyze the data leakage risk associated with a website as particular to different locations from which a consumer might be accessing the website.

[0024] Each of the threat scores in attribute table 38 may be dynamically modified in various embodiments, beginning with a training set of known data leakers and innocuous third-party tags. Some attributes are likely to be common for both innocuous third-party tags and the data leaker tags, such as, for example, setting a cookie on consumer computer 14 through browser 16. Other attributes, however, will be more likely to only be characteristic of data leakers. One manner in which the attribute scores may be determined based on this approach is through a Bayesian inference, which is a statistical technique that is well known in the art. An alternative approach is to consider loose conditions for threats and rebalance the magnitude of the threat scores through an averaged centroid based off of the frequency and aggregate scores relative to each threat. Using a centroid model, one may calculate the ideal median threat percentage based upon the distribution of the threats in all sites, define it as occurring half the time with a set threat score and then scale the scores of other threats around it. One may in this approach define what the highest score should be by doubling the default score for something that occurs half of the time. To de-emphasize more common appearing threats, one may create an inverse relationship between the percent of time a threat is seen against the ideal percentage of the most common threat. The new score ("newscore") of a threat is then defined by:
newscore = (1 – percent / hp) * highest_score_allowed,
where:
hp = 2 * (max(score_percents_seen) + min(score_percents_seen) +

median_percent) / 3
If the ideal midpoint is defined as the median_percent on a number line from zero to one after being normalized between the lowest and highest percentage seen, then one can multiply it by two to get what the ideal highest percentage of a threat should be if the median were the middle of the distribution.

[0025] The preferred embodiment of the invention is implemented as a number of computing devices 500 as illustrated in Fig. 3, each of which is programmed by means of instructions to result in a special-purpose computing device to perform the various functionality described herein. It may be, for example, the manner in which MSP server 10, publisher server 12, and various resource servers 18, 20, 22, and 24 are implemented as shown in Fig. 1. Computing device 500 may be physically implemented in a number of different forms. For example, it may be implemented as a standard computer server as shown in Fig. 3; as a group of servers; as a personal computer; or as a laptop computer.

[0026] Computing device 500 includes microprocessor 502, memory 504, an input/output device or devices such as display 506, and storage device 508, such as a solid-state drive or magnetic hard drive. Each of these components is interconnected using various buses or networks, and several of the components may be mounted on a common PC board or in other manners as appropriate.

[0027] Microprocessor 502 may execute instructions within computing device 500, including instructions stored in memory 504. Microprocessor 502 may be implemented as a single microprocessor or multiple microprocessors, which may be either serial or parallel computing

microprocessors.

[0028] Memory 504 stores information within computing device 500. The memory 504 may be implemented as one or more of a computer-readable medium or media, a volatile memory unit or units such as flash memory or RAM, or a nonvolatile memory unit or units such as ROM. Memory 504 may be partially or wholly integrated within microprocessor 502, or may be an entirely stand-alone device in communication with microprocessor 502 along a bus, or may be a combination such as on-board cache memory in conjunction with separate RAM memory. Memory 504 may include multiple levels with different levels of memory 504 operating at different read/write speeds, including multiple-level caches as are known in the art.

[0029] Display 506 provide for interaction with a user, and may be implemented, for example, as an LCD (light emitting diode) or LCD (liquid crystal display) monitor for displaying information to the user, in addition to a keyboard and a pointing device, for example, a mouse, by which the user may provide input to the computer. Other kinds of devices may be used to provide for interaction with a user as well.

[0030] Various implementations of the systems and methods described herein may be realized in computer hardware, firmware, software, and/or combinations thereof. These various implementations may include implementation in one or more computer programs that are executable and/or interpretable on a programmable system including at least one programmable microprocessor 502, which may be special or general purpose, coupled to receive data and instructions from, and to transmit data and instructions to, a storage system, one or more input device, and one or more output device.

[0031] The computing system can include clients and servers. In this case, client device 512 runs a web browser 514 in order to access the Internet 510, which allows interconnection with computing device 500. A client and server are generally remote from each other and typically interact through a communication network. For example, publisher server 12 of Fig. 1 may be structured as a server in this configuration and consumer computer 14 as a client device. The relationship of client and server arises by virtue of computer programs running on the respective computers and having a client-server relationship to each other.

[0032] Unless otherwise stated, all technical and scientific terms used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this invention belongs. Although any methods and materials similar or equivalent to those described herein can also be used in the practice or testing of the present invention, a limited number of the exemplary methods and materials are described herein. It will be apparent to those skilled in the art that many more modifications are possible without departing from the inventive concepts herein.

[0033] All terms used herein should be interpreted in the broadest possible manner consistent with the context. In particular, the terms "comprises" and "comprising" should be interpreted as referring to elements, components, or steps in a non-exclusive manner, indicating that the

referenced elements, components, or steps may be present, or utilized, or combined with other elements, components, or steps that are not expressly referenced. When a grouping is used herein, all individual members of the group and all combinations and subcombinations possible of the group are intended to be individually included.

[0034] The present invention has been described with reference to certain preferred and alternative embodiments that are intended to be exemplary only and not limiting to the full scope of the present invention, as set forth in the appended claims.

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US61879909 **[0001]**
- US2010186088A **[0008]**
- US2010281536A **[0008]**

PATENTKRAV

1. Computer-implementeret fremgangsmåde til identifikation af datalækage, omfattende følgende trin:

a. tilgåelse (30), via software hosted ved i det mindste en markedsførings-serviceudbyder (marketing services provider), MSP, -server (10), af en website, som er hosted ved en publisher server, hvor MSP-serveren (10) er konfigureret til at efterligne en standard web browser, således at det fra websiten ser ud som om, at MSP-serveren (10) anvender en standard web browser;

b. søgning (32) på websiten for at identificere tredieparts tags eller pixler på websiten;

c. hvis tredieparts tags eller pixels findes, dynamisk beregning af en individuel trussel-score (40) knyttet til hver tredieparts tag eller pixel fundet på websiten, baseret på anvendelse af hver tredieparts tag eller pixel af enhver personlig information knyttet til et individ, som tilgår websiten;

d. hvis tredieparts tags eller pixels findes, bestemmelse af om nogle af tredieparts tags eller pixels er forbundet med nogle fremsendte tredieparts tags eller pixels;

e. hvis fremsendte tredieparts tags eller pixels findes, baseret på bestemmelse af, om nogle af tredieparts tags eller pixels er forbundet med nogle fremsendte tredieparts tags eller pixels, dynamisk beregning af en individuel trussel-score knyttet til enhver fremsendt tredieparts tag eller pixel fundet på websiten, baseret på anvendelse af hver fremsendt tredieparts tag eller pixel af enhver personlig information knyttet til individet, som tilgår websiten;

f. hvis fremsendte tredieparts tags eller pixels findes og individuelle trussel-score knyttet til hver fremsendt tredieparts tag eller pixel beregnes, tilknytning af den individuelle trussel-score knyttet til hver fremsendt tredieparts tag eller pixel til enhver tredieparts tag eller pixel forbundet med den fremsendte tredieparts tag eller pixel, hvis den individuelle trussel-score knyttet til den fremsendte tredieparts tag eller pixel er højere end den individuelle trussel-score knyttet til den tredieparts tag eller pixel;

g. beregning (42) af en kumulativ trussel-score knyttet til websiten, baseret på de individuelle trussel-score for tredieparts tags eller pixels fundet og fremsendte tredieparts tags eller pixels fundet; og

h. hvis en kumulativ trussel-score er beregnet, så bestemmes (44) om den kumulative trussel-score er over en trussel-scoretærskel, og hvis dette er tilfældet, identificering (46) af websiten som en datalækagetrussel.

2. Computer-implementeret fremgangsmåde til identificering af datalækage ifølge krav 1, hvor websiten er knyttet til et domæne, og yderligere omfattende følgende trin:

a. søgning (36) efter domænet i en tabel over kendte datalækagedomæner (34);

b. identificering af websiten som en datalækagetrussel, hvis domænet knyttet til websiten er fundet i tabellen over kendte datalækagedomæner; og

c. opdatering af tabellen over kendte datalækager, hvis domænet er delt med andre websites.

3. Computer-implementeret fremgangsmåde til identificering af datalækage ifølge krav 2, hvor trinnet med beregning (40) af individuelle trussel-scorer omfatter trinnet med bestemmelse af, om en ressource knyttet til hver tredieparts tag eller pixel udviser et af adskillige attributter (38), hvor hvert attribut er knyttet til en datalækagetrussel.

4. Computer-implementeret fremgangsmåde til identificering af datalækage ifølge krav 3, hvor hvert attribut er knyttet til en genbalancerbar numerisk trussel-score.

5. Computer-implementeret fremgangsmåde til identificering af datalækage ifølge krav 4, hvor den individuelle trussel-score for hver tredieparts tag eller pixel beregnes som summen af trussel-scorerne knyttet til hvert attribut knyttet til den respektive tredieparts tag eller pixel.

6. Computer-implementeret fremgangsmåde til identificering af datalækage ifølge krav 5, hvor den kumulative trussel-score beregnes som summen af de individuelle trussel-scorer.

7. Computer-implementeret fremgangsmåde ifølge krav 1, hvor tærsklen er en numerisk score, som er i det mindste så høj som en identificeret procentdel af adskillige reference-kumulative trussel-scorer.

8. System til bestemmelse af en trussel for datalækage for en website, omfattende:

en markedsføringstjenesteudbyder (marketing service provider), MSP, -server (10), som hoster software, som tilvejebringer adgang til en website hosted ved en publisher server, hvor MSP-serveren er konfigureret til at efterligne en standard web browser, således at det fra websiten ser ud som om, at MSP-serveren (10) anvender en standard web browser;

softwaren er konfigureret til at søge (32) på websiten for at identificere tredieparts tags eller pixels på websiten;

hvis tredieparts tags eller pixels findes, er softwaren konfigureret til dynamisk at beregne en individuel trussel-score (40) knyttet til hver tredieparts tag eller pixel fundet på websiten, baseret på anvendelse af hver tredieparts tag eller pixel af enhver personlig information knyttet til et individ, som tilgår websiten;

hvis tredieparts tags eller pixels findes, er softwaren konfigureret til at bestemme, om nogle af tredieparts tags eller pixels er knyttet til nogle fremsendte tredieparts tags eller pixels;

hvis fremsendte tredieparts tags eller pixels findes, baseret på bestemmelse af, om nogle af de tredieparts tags eller pixels er knyttet til nogle fremsendte tredieparts tags eller pixels, er softwaren konfigureret til dynamisk at beregne en individuel trussel-score knyttet til hver af de fremsendte tredieparts tags eller pixels fundet på websiten, baseret på anvendelse af hver fremsendt tredieparts tag eller pixel af nogen personlig information knyttet til individet, som tilgår websiten;

hvis fremsendte tredieparts tags eller pixels findes, og individuel trussel-score knyttet til hver fremsendt tredieparts tag eller pixel beregnes, er softwaren konfigureret til at tilknytte den individuelle trussel-score knyttet til hver fremsendt tredieparts tag eller pixel til vilkårlige tredieparts tag eller pixel knyttet til den fremsendte tredieparts tag eller pixel, hvis den individuelle trussel-score knyttet til den fremsendte tredieparts tag eller pixel er højere end den individuelle trussel-score knyttet til den tredieparts tag eller pixel;

softwaren er konfigureret til at beregne (42) en kumulativ trussel-score knyttet til websiten baseret på de individuelle trussel-score for tredieparts tags eller pixels fundet og fremsendte tredieparts tags eller pixels fundet; og

hvis en kumulativ trussel-score beregnes, er softwaren konfigureret til at bestemme (44), om den kumulative trussel-score er over en trussel-scoretærskel og hvis dette er tilfældet, at identificere (46) websiten som en datalækagetrussel.

9. System til bestemmelse af en trussel for datalækage for en website ifølge krav 8, som yderligere omfatter i det mindste en yderligere MSP-server placeret i en geografisk fjern position fra MSP-serveren (10), og konfigureret til at tilgå en website på en måde som en standard web-browser, men konfigureret til at analysere websiten for at identificere og undersøge eventuelle tredieparts piggybacking tags eller pixels fundet på websiten.

10. Fremgangsmåde ifølge krav 1, hvor trinnet (40) med dynamisk beregning af de individuelle trussel-scorer knyttet til hver tredieparts piggybacking tag eller pixel omfatter

en bayesisk inferens-analyse af attributter knyttet til adskillige domæner, således at domæner med modsvarende attributter som kendte ikke datalækage domæner modtager lavere trussel-scorer.

11. Fremgangsmåde ifølge krav 1, hvor trin (40) med dynamisk beregning af de individuelle trussel-scorer knyttet til hver tredieparts tag eller pixel omfatter trinnet med hensyntagen til løse betingelser for et sæt af trusler og rebalancering af en størrelse af trussel-scorerne via en midlet centroid baseret på frekvens, og aggregatet scorer relativt til hver trussel.

12. Fremgangsmåde ifølge krav 11, hvor trinnet (40) med dynamisk beregning af de individuelle trussel-scorer omfatter trinnet med beregning af en ideel middel trusselprocentdel baseret på en fordeling af trusler i alle analyserede websites.

13. Fremgangsmåde ifølge krav 12, hvor den ideelle middel trusselprocentdel beregnes baseret på fordelingen af trusler i alle analyserede websites, og er defineret som optrædende halvdelen af tiden med et sæt trussel-scorer og derefter skalering af trussel-scorerne for de øvrige trusler omkring en ideel middel trusselprocentdel.

14. Fremgangsmåde ifølge krav 13, som yderligere omfatter beregningen af den højeste score ved fordobling af en default scoring for en trussel, som optræder halvdelen af tiden.

15. Fremgangsmåde ifølge krav 12, hvor mere almindeligt forekommende trusler undertrykkes i trinnet med dynamisk beregning af individuelle trussel-scorer ved tilvejebringelse af en omvendt relation imellem en procent af tid, hvor en trussel ses imod en ideel procentdel af den hyppigst forekommende trussel.

DRAWINGS

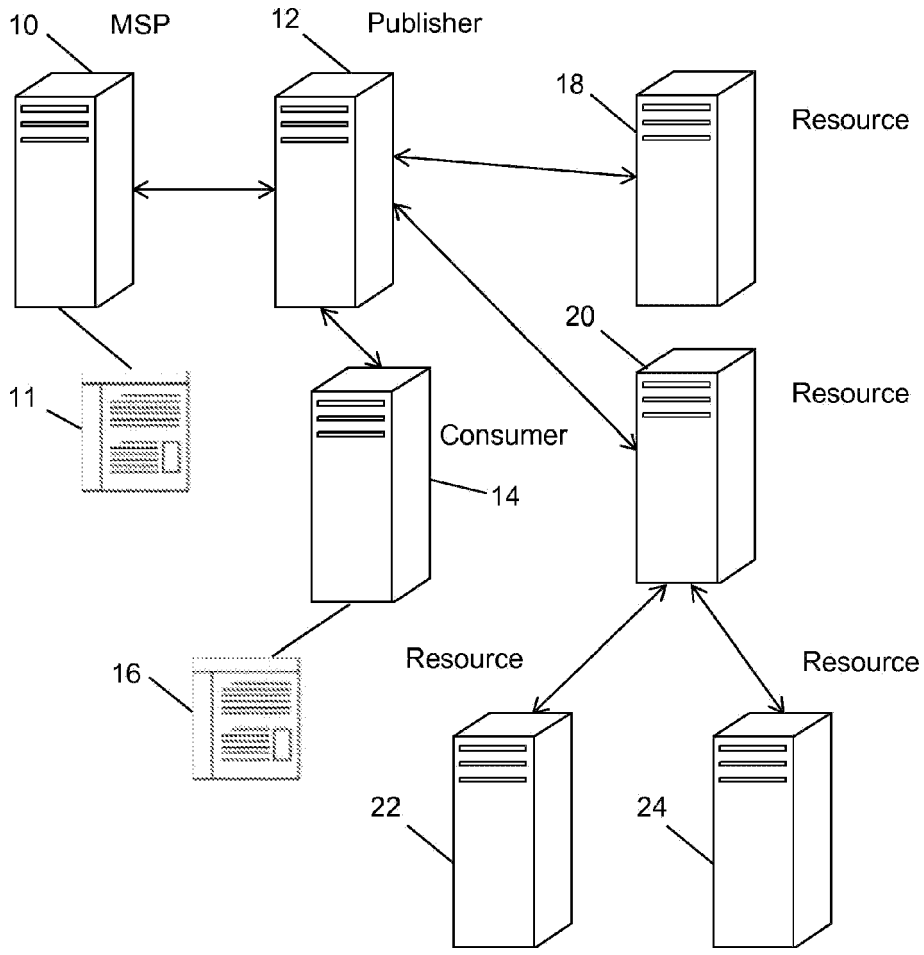

Fig. 1

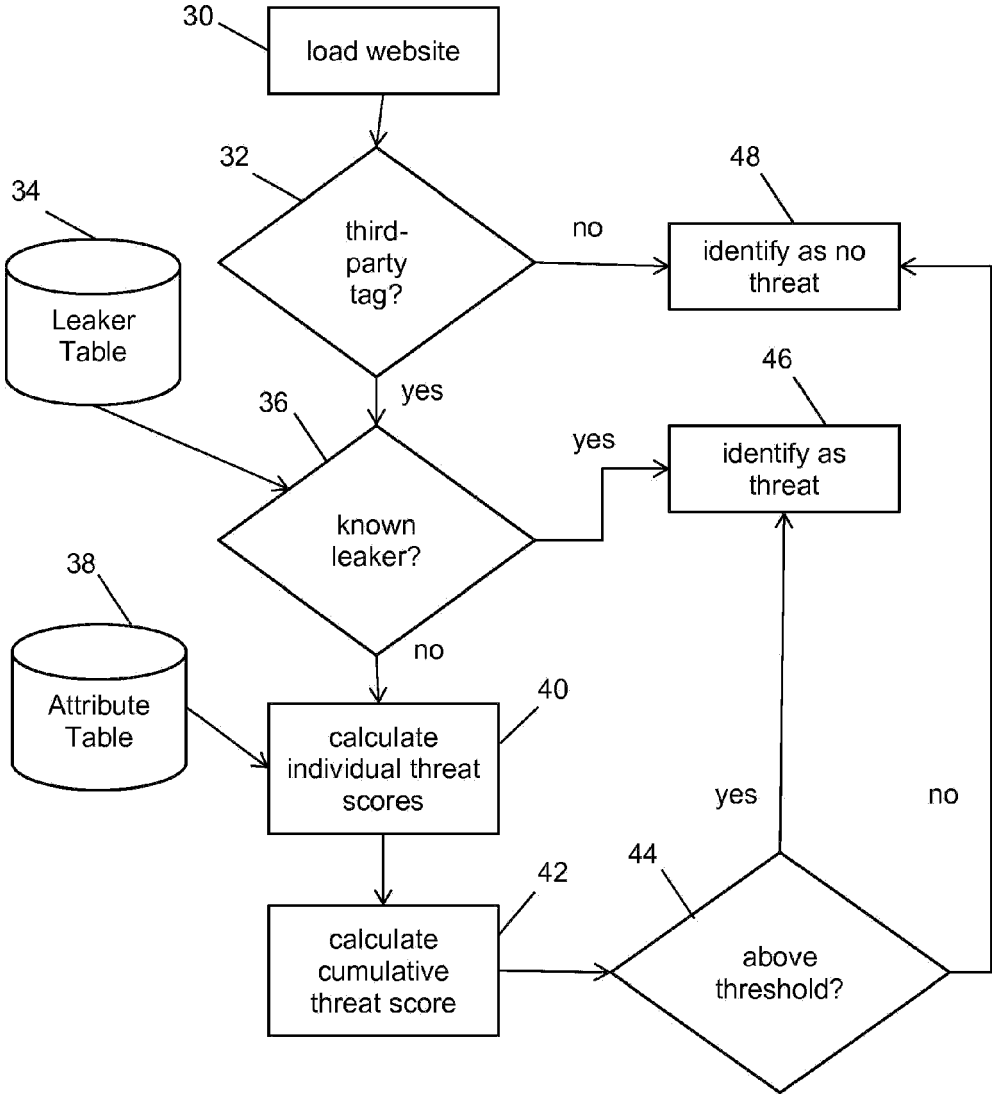
30
load website
32
third-party tag?
no
48
identify as no threat
34
Leaker Table
yes
46
36
yes
identify as threat
known leaker?
38
no
Attribute Table
40
calculate individual threat scores
yes
no
44
42
calculate cumulative threat score
above threshold?

Fig. 2

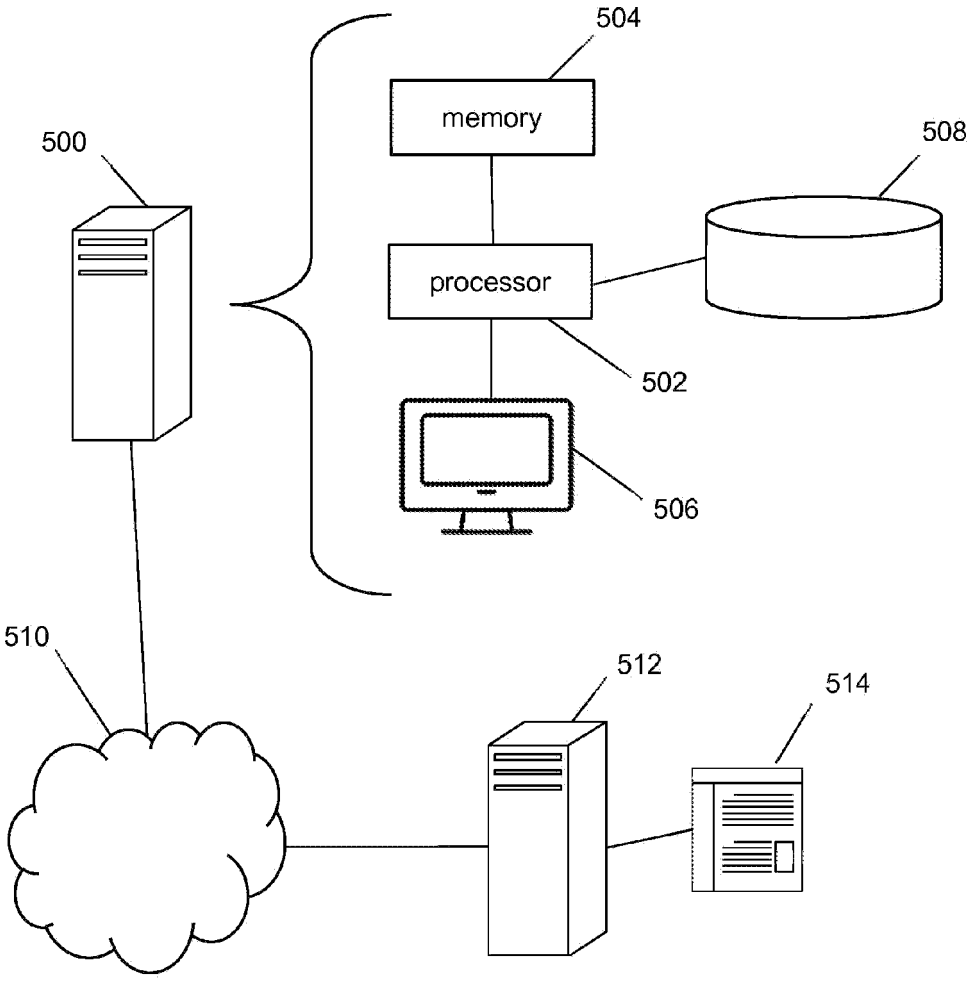

Fig. 3