

12. (canceled)

13. The method of claim 1, wherein the first and second URLs correspond to webpages, the method further comprising, in response to the receiving of the first request, sending first webpage data requested in the first request, including the second URL, to the user device, wherein the second request is received in response to operation of the user device to select the second URL.

14. The method of claim 1, further comprising, in response to the receiving of the second request, sending second webpage data requested in the second request, including the third URL, to the user device, wherein the third request is received in response to operation of the user device to select the third URL.

15. The method of claim 1, wherein the first request is sent from the user device in response to the user operating the user device to cause a webpage request for a webpage corresponding to the first URL to be sent to a further server means, wherein the first request is a first tracking request; wherein the second request is sent from the user device in response to a user operating the user device to cause a webpage request for a webpage corresponding to the second URL to be sent to the further server means, wherein the second request is a second tracking request.

16. The method of claim 15, further comprising: after receiving the first tracking request and storing the first URL, and before receiving the second tracking request, receiving a further tracking request deriving from a predetermined event or interaction relating to a web browser running on the user device, wherein the further tracking request is not sent in response to the user operating the user device to cause a webpage request to be sent to the further server means, wherein the further tracking request includes the first URL; determining, by matching the first URL received in the further tracking request and the stored first URL, that the first tracking request and the further tracking request were sent from the user device.

17. The method of claim 15, wherein the third request is sent from the user device in response to a user operating the user device to cause a webpage request for a webpage corresponding to the third URL to be sent to the further server means, wherein the third request is a third tracking request.

18. The method of claim 15, wherein the third request is a yet further tracking request, the yet further tracking request deriving from a predetermined event or interaction relating to a web browser running on the user device, wherein the yet further tracking request is not sent in response to the user operating the user device to cause a webpage request to be sent to the further server means, wherein the yet further tracking request includes the second URL; determining, by matching the second URL received in the yet further tracking request and the stored second URL, that the second tracking request and the yet further tracking request were sent from the user device.

19. The method of claim 16, further comprising: in response to receiving the further tracking request, determining that a session identifier cannot be located in the further tracking request and then performing the determining, by matching the first URL received in the second request and the stored first URL, that the first request and the second request were sent from the user device.

20. The method of claim 15, wherein the further tracking request includes a set of data points, further comprising: further to the receiving of the further tracking request, comparing the data points of the further tracking request against the stored sets of data points deriving from a plurality of requests received at the server means from a plurality of user devices, wherein the received requests include the first tracking request, determining that the data points of the further tracking request match to the stored data points of at least two of the plurality of received tracking requests, the at least two received tracking requests including the first tracking request, wherein the determining, by matching the first URL received in the first tracking request and the stored first URL, includes scanning stored URLs associated with the at least two requests to identify the first URL.

21. (canceled)

22. A computer program product comprising computer program code stored on a computer readable storage medium, which, when executed by a processing means, causes the following steps to be performed:

receiving at a server means a first request from a user device, the request including a first uniform resource locator (URL), and storing the first URL; receiving a second request from the user device, including a second URL and the first URL; determining, by matching the first URL received in the second request and the stored first URL, that the first request and the second request were sent from the user device; storing the second URL, so that a third request including the second URL can be matched using the stored second URL.

23. (canceled)

24. Apparatus comprising: a processing means; and a memory means having a computer program stored thereon, wherein the computer program comprises computer program code, wherein the processing means, together with the memory means and the computer program code, are configured to cause the following method to be performed: receiving at a server means a first request from a user device, the request including a first uniform resource locator (URL), and storing the first URL; receiving a second request from the user device, including a second URL and the first URL; determining, by matching the first information and the stored first URL, that the first request and the second request were sent from the user device; storing the second URL, so that a third request including the second URL can be matched the stored second URL.

25. (canceled)

Description

FIELD OF THE INVENTION

[0001] The invention relates to a method of determining that requests received at a server means are sent from a same device, where the server means receives requests from multiple devices. The invention also relates to related apparatus and a computer program product.

BACKGROUND

[0002] In an environment in which a server receives requests from multiple client applications using a stateless protocol, it is often wanted to configure the server to determine which requests are received from each client application. Thus, when the server receives a first request from a particular client application, the server initiates a session, assigns a unique session identifier to the session, and sends the session identifier to the client application. Then, each time that the client application sends a request to the server, the client application includes the session identifier so that the server can associate the requests from that client application. The server may store session data in association with the session identifier. Particular session data is thus associated with a particular client application.

[0003] Session identifiers are commonly used by web servers. When a client application in the form of a web browser receives a session identifier, the web browser stores the session identifier as part of a cookie. The cookie also includes the domain of the web server. When the web browser next sends an HTTP request to the same domain, the web browser identifies the stored cookie using the domain and includes the session identifier with the HTTP request.

[0004] A major problem with the above described approach is that some web browsers are prevented from storing cookies and thus cannot store session identifiers. Accordingly, session identifiers cannot be sent with requests by such web browsers to the server and so later requests cannot be reconciled with earlier ones. It is an object of the present invention to address this problem.

SUMMARY OF THE INVENTION

[0005] In accordance with a first aspect of the present invention, there is provided a method comprising: receiving at a server means a first request from a user device, the request including a first uniform resource locator (URL), and storing the first URL; receiving a second request from the user device, including a second URL and first information indicative of the first URL; determining, by matching the first information and the stored first URL, that the first request and the second request were sent from the user device; storing the

fingerprinting process has been performed, the daisy chain process is then run. If the daisy chain process fails to match a received request to a previously established session, a new session is initiated and a session identifier is generated and stored in the session data store.

[0026] Use of these three processes in combination results in a high probability that, if a session has been initiated, the corresponding session identifier will be identified for any received request. Nevertheless, the daisy chain process may be carried out independently of either the cookie based process and the fingerprinting process. Alternatively, the daisy chain process may be carried out following one of the cookie based process or the fingerprinting process, where the other is not used for session reconciliation.

[0027] The daisy chain process is described first in isolation with reference to FIG. 3 when used to reconcile received requests. At step 300, the server 100 receives a first request from the user device 102, the first request including a destination URL. The request does not derive from selecting a link on a webpage of the website, but from selection of a link on another web site or entering of an address of the webpage. In response, the server 100 generates a session identifier and stores the session identifier in the session data store at step 302. The server 100 also sends the requested webpage to the user device 102 at step 304. The server 100 also stores the destination URL in association with the session identifier.

[0028] Next, the user selects a link for another webpage on the received webpage, thereby causing the user device 102 to send a second request to the server 100, the second request also including a destination URL of the other webpage and the referrer URL, which is the destination URL in the first request. The server 100 receives the second request at step 306.

[0029] In response to the second request, the server 100 determines whether the second request is from a user device for which a session has already been established. To do this, the server 100 scans at step 308 candidate stored destination URLs associated with current session identifiers to determine whether any destination URL is the same as the referrer URL in the second request. The server 100 is configured so that the candidate stored destination URLs that are scanned are only the destination URLs received in the most recently received request relating to a session.

[0030] If at step 308 the server 100 identifies a matching destination URL, then at step 310, the server 100 determines that the second request relates to the session identified by the session identifier with which the matching destination URL is associated. The server 100 may then use session data stored for the session in response to the second request and/or store further session data in response to the second request, as indicated at step 312.

[0031] The server 100 also stores the destination URL of the second request in association with the session identifier, and updates so that the destination URL in the second request is stored as the most recently received destination URL relating to the session. In response to the second request, the server 100 sends a second webpage to the user device, the webpage including an URL of a third webpage. If the user operates the web browser to send a third request with the URL of the third webpage as the destination URL and the URL of the second webpage as the referrer URL, the daisy chain process can be used to match the third request to the session in the same way that the second request was matched to the session, that is, by matching the referrer URL to a destination URL of the second request stored at the server 100. Each further request from the user device 102 for webpages of the website that includes a destination URL and a referrer URL, the referrer URL also identifying a page of the same website, may be matched to the session identifier using the same process, that is, by matching the referrer URL with the destination URL of the previous request relating to the session.

[0032] As mentioned above, the daisy chain process may be used alone and would work well in reconciling received requests where the number of received requests is low and/or where the session has a short expiry time. However, it is envisaged that the process is used in combination with the other processes for determining that received requests have originated from the same user device 102. Accordingly, in a preferred embodiment, first the cookie based process is attempted in which the session identifier is used to enable the server 100 to match received requests to a particular session.

[0033] Referring to FIG. 4, first and second steps are the same as steps 300 and 302. At step 404, the server 100 sends the requested first webpage to the user device 102, like at step 304, together with a cookie

two sets of data points corresponding to first requests from at least two user devices 102, in embodiments the server 100 is configured to perform the daisy chain process to determine which, if any, of those first requests come from the same user device as the second request.

[0055] In an example scenarios in which embodiments of the invention may be implemented. in response to a first request sent to the server 100, the server sends to the user device 102 a webpage including an online form in response to a first request. The form has a form identifier with it. The form identifier is stored as session data. The user then enters data into the online form and selects an URL to send the entered data to the server 100 with a second request. The server 100 then uses the above described processes to associate the second request with the session identifier, and then stores the entered data as session data in association with the session identifier and the form identifier. Forms may be on multiple pages and the data entered by the users is associating using the session identifier using the above described processes.

[0056] In another example, a user may send a first request to the server 100, which in this case hosts a shopping website, and receive a first webpage. The user may then interact with the webpage, involving sending a second request to the server 100 indicating that a particular item is to be added to an online shopping cart. Information indicative of this is then added to the session data. The user may then selects a URL such that a request is sent for a webpage enabling payment. It is essential that the request for this webpage is associated with the previous requests, otherwise, as far as the user is concerned, the contents of the online cart will be lost.

[0057] In another embodiment, a data collection server 101 is also connected to the communications network 104. The data collection server 101 is indicated using dashed lines in FIG. 1 as it is not part of the embodiment as described above in relation to FIG. 1. In this embodiment, the server 100 still sends webpages to any of the user devices 102 on receipt of requests from the user devices 102, and may operate as described above using any one or more of the cookie based process, the fingerprinting process and the daisy chain process. The server 100 may alternatively match requests received from user devices 102 using other methods or not match received requests. However, in this embodiment, each webpage includes a portion of tracking code that is executed by the web browser each time a predetermined interaction or event occurs on the webpages, or alternatively that executes to pull some code to the web browser, that pulled code being executed each time a predetermined interaction or event occurs. The tracking code is typically JavaScript and the web browser includes a JavaScript engine for executing the JavaScript. The result is that a tracking request is sent to the data collection server 101 each time the predetermined interaction or event occurs. The predetermined interaction may be interaction by a cursor with an HTML element.

[0058] It is known for the JavaScript to include a cookie. Conventionally, the cookie is sent with tracking requests sent to the data collection server 101. If the network, computer or web browser does not allow this, the same problem of matching tracking requests occurs at the data collection server 101 as may occur at the web server 100, as described above.

[0059] A method of reconciling received requests at the data collection server 101 will now be described with reference to FIG. 6. A request for a webpage, including a destination URL of the requested webpage, is sent from the user device 102 and is received at the server 100, as at step 300. In response, the server 100 sends the requested webpage to the user device 102.

[0060] The JavaScript executes to cause data collection to begin. The JavaScript causes the web browser to send a first tracking request to the data collection server 101, which is received at step 600. The first tracking request includes the URL of the received webpage.

[0061] The server 100 may have sent a cookie with the webpage to the user device 102 and, if such a cookie was sent, the user device 102 may have received it and the web browser stored it. In this case, the JavaScript reads the cookie and sends a copy of the cookie to the tracking server 101 with the first tracking request. Provided future tracking requests sent from the user device 102 to the data collection server 101 include a copy of the cookie, the cookie-based process can be used to reconcile the received tracking requests. Where the cookie-based process cannot be used or fails, the fingerprinting process can be used. The daisy chaining process may be used where either these processes fail or are not used, as described above, although the daisy chain process may also be used independently.

[0062] At step 602, in response to receiving the first tracking request, the data collection server 101 determines whether the first tracking request matches to any session. This may be done using any of the processes described above to try to match the first tracking request to a session. Since this is a first tracking request, a session has not been established, so the data collection server 101 will not be able to identify a session. The data collection server 101 thus initiates a session, generates a session identifier for the session, and stores the URL of the webpage in association with the session identifier. Other data may be stored in association with the session identifier.

[0063] When the user operates the user device 102 to interact with the webpage, the JavaScript causes sending of a second tracking request including information relating to the interaction to the data collection server 101, together with the URL of the webpage. The interaction may relate, for example, to any of a number of interactions for which data may be collected.

[0064] The second tracking request includes a copy of the URL of the current webpage. Where the daisy chain process is used, at step 604, the data collection server 101 receives the second tracking request and scans the URLs stored in association with current session identifiers against the URL received in the second tracking request. If one such URL matches the URL in the tracking request, the data collection server 101, at step 606, determines the session identifier and thus the session data for the session identifier, which is then updated to include the information relating to the interaction.

[0065] The user may also operate the user device 102 to select a link on the webpage, which causes a request including a second destination URL to be sent to the server 100. In addition to this second request being sent to the server 100, selecting of the link causes the JavaScript to cause a third tracking request to be sent to the data collection server 101. The third tracking request includes the referrer URL, that is, the URL of the webpage on which the link was selected, and the destination URL, that is, the URL of the new webpage. At step 608, the data collection server 101 receives the third tracking request and then scans potentially relevant URLs stored with current sessions to try to match the referrer URL with one of the stored URLs. If one such URL matches the referrer URL, the data collection server 101, at step 610, determines the session identifier and thus the session data for the session identifier associated with that referrer URL. The potentially relevant URLs are those where is then updated to include the new destination URL and optionally other data.

[0066] The second tracking request is an example of a tracking request that includes the current URL of the webpage. The daisy chain process works by matching the current URL with a stored URL that is associated with a session identifier. Many such second tracking requests may be sent from the user device 102 to the data collection server 101 and matched in this way. The second tracking request may have fields corresponding to a referrer URL and a destination URL, in which case both may be populated with the URL of the current webpage. Such a second tracking request is sent in response to a predetermined interaction other than selection of link to request a new webpage. As well as being sent in response to predetermined user interactions, such request may be sent in response to other predetermined events, as is well known to persons skilled in the art.

[0067] The third tracking request is an example of a tracking request typically including fields corresponding to a referrer URL and a destination URL, where the fields are populated with different URLs. This occurs when the user has selected a link resulting in a new webpage being requested. In this case, the daisy chain process works by matching the old URL (in the referrer URL) with a stored URL that is associated with session identifiers. Many such third tracking requests may be sent from the user device 102 to the data collection server 101 and the old URL matched to the URL of the previous webpage, with the URL of the new webpage being stored.

[0068] The daisy chain process works particularly well where the cookie based process and/or the fingerprinting is attempted beforehand. Where the cookie based process is attempted, the scanning of the URLs in the daisy chain process may include scanning to establish where a cookie has previously been associated with a particular session identifier. If it has, the destination URL associated with that session identifier need not be scanned.

[0069] Similarly, where the fingerprinting process has previously been successfully used at reconciling requests from a user device, the URL associated with the session identifier corresponding to that user device need not be scanned in the daisy chain process. In this way, when the daisy chain process is used, a large

