

METHOD AND APPARATUS FOR ASSIGNING DEVICE FINGERPRINTS TO INTERNET DEVICES

[0001] This application is based upon and claims priority to Chinese Patent Application No. 201510072290.5, filed February 11, 2015, the entire contents of which are incorporated herein by reference.

TECHNICAL FIELD

[0002] The present disclosure generally relates to the field of computing devices, and particularly to a method and apparatus for assigning device fingerprints to internet devices.

BACKGROUND

[0003] With the use of more and more internet applications, security becomes more and more important because financial losses caused by hackers and frauds have been significant. By associating the device fingerprints with user behaviors, it may be more efficient to crack fraudulent activities over the internet to protect security. For example, after a mobile phone is assigned a device fingerprint, the mobile phone may be associated with corresponding account information, such that when the account information is used to log in on a device having different device fingerprints, the security of the account may be protected by employing various security measures.

SUMMARY

[0004] Consistent with embodiments of the present disclosure, there is provided a

method for assigning a device fingerprint to a target device. The method includes acquiring first attributes of the target device, the first attributes including at least one browser plug-in or one font; calculating a correlation value with respect the first attributes of the target device and second attributes of at least one second device having a device fingerprint stored in a device library. The first attributes correspond to the second attributes. The method further includes: if the correlation value is greater than or equal to a pre-determined threshold value, assigning the device fingerprint of the at least one second device to the target device; and if the correlation value is smaller than the pre-determined threshold value, storing the first attributes of the target device to the device library and assigning a new device fingerprint to the target device.

[0005] Consistent with embodiments of the present disclosure, there is provided an apparatus for assigning a device fingerprint to a target device. The apparatus includes one of more processors, and a memory device configured to store instructions executable by the one or more processors. The one or more processors are configured to perform: acquiring first attributes of the target device, the first attributes including at least one browser plug-in or one font; calculating a correlation value with respect the first attributes of the target device and second attributes of at least one second device having a device fingerprint stored in a device library. The first attributes correspond to the second attributes. The one or more processors are further configured to perform: if the correlation value is greater than or equal to a pre-determined threshold value, assigning the device fingerprint of the at least one second device to the target device; and if the correlation value is smaller than the pre-determined threshold value, storing the first attributes of the target device to the device library and

assigning a new device fingerprint to the target device.

[0006] Consistent with embodiments of the present disclosure, there is provided a non-transitory computer-readable storage medium storing instructions, which, when executed by a computer, causes the computer to perform: acquiring first attributes of the target device, the first attributes including at least one browser plug-in or one font; calculating a correlation value with respect the first attributes of the target device and second attributes of at least one second device having a device fingerprint stored in a device library. The first attributes correspond to the second attributes. The computer is further configured to perform: if the correlation value is greater than or equal to a pre-determined threshold value, assigning the device fingerprint of the at least one second device to the target device; and if the correlation value is smaller than the pre-determined threshold value, storing the first attributes of the target device to the device library and assigning a new device fingerprint to the target device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The accompanying drawings, which are incorporated in and constitute a part of this specification, illustrate embodiments consistent with the disclosure and, together with the description, serve to explain the principles of the disclosure.

[0008] Fig. 1A is a flow chart of an exemplary method for assigning a device fingerprint to an internet device, consistent with the embodiments of this disclosure;

[0009] Fig. 1B show a flow char of calculating correlation values consistent with the embodiments of this disclosure;

[0010] Fig. 2 is a flow chart of an exemplary method for assigning a device

fingerprint to an internet device, consistent with the embodiments of this disclosure; and

[0011] Fig. 3 shows an exemplary apparatus 300 for assigning device fingerprints to target devices, consistent with embodiments of the present disclosure.

DETAILED DESCRIPTION

[0012] Reference will now be made in detail to exemplary embodiments, examples of which are illustrated in the accompanying drawings. The following description refers to the accompanying drawings in which the same numbers in different drawings represent the same or similar elements unless otherwise represented. The implementations set forth in the following description of exemplary embodiments do not represent all implementations consistent with the disclosure. Instead, they are merely examples of apparatuses and methods consistent with aspects related to the disclosure as recited in the appended claims.

[0013] Generally, a device fingerprint is a device value determined by the following steps: (1) obtaining attribute values corresponding to a plurality of attributes of the device, for example, an operating system version, a number of CPU cores, and a primary operation frequency, (2) matching the attribute values with attribute values in a device library storing sets of attribute values, and (3) determining a device identification corresponding to the device therein.

[0014] Fig. 1A shows a flow chart of a method 100 for assigning a device fingerprint to an internet device, consistent with the embodiments of this disclosure. An internet device may be a mobile phone, a smart phone, a tablet, a laptop computer, a desktop computer, or any other electronic devices that can connect to the internet. Method 100 may be performed

by a server device connected to the internet. As shown in Fig. 1A, method 100 may include the following steps.

[0015] In step 102, attributes of a target internet device to be assigned a device fingerprint are acquired. The attributes of the target device include at least browser plug-ins and installed on the target device. In some other embodiments, the attributes include font types.

[0016] In step 104, correlation values with respect to the attributes of the target device and attributes of one or more second devices stored in a device library are calculated. In some embodiments, the device library stores at least one set of attributes of a second device that has been assigned a device fingerprint. Correlation values are calculated between the attributes of the target device and attributes of each of second devices stored in the device library. For example, correlation values of the browser plug-ins and/or font types of the target device and browser plug-ins and/or font types of each of the second devices are calculated. One correlation value can be obtained between attributes of the target device and attributes of a second device. When there are multiple second devices in the device library, a number of correlation values corresponding to the number of the second devices can be calculated and obtained. In some embodiments, sequence and category of the attributes of the target device correspond to the sequence and category of the attributes of the second device in order to calculate the correlation values. For example, if the sequence and category of the attributes of the target device are: [plug-in I], [plug-in II], and [font I], then a correlation value may be calculated if the sequence and category of the attributes of the second device are: [plug-in I'], [plug-in II'], and [font I'] as well.

[0017] In step 106, the largest correlation value obtained in step 104 is selected and compared with a pre-determined threshold value. In some embodiments, each of the correlation values obtained in step 104 may be compared with the pre-determined threshold value, without having to select the largest correlation value.

[0018] If the largest correlation value is equal to or greater than the pre-determined threshold value, in step 108, the device fingerprint of the second device corresponding to the largest correlation value is assigned to the target device. If the largest correlation value is smaller than the pre-determined threshold value, in step 110, the attributes of the target device including its browser plug-ins and font types are saved in the device library and a new device fingerprint is assigned to the target device. The new device fingerprint may include, for example, a device identification of the target device.

[0019] For example, the target device may be a personal computer (PC). The PC may have several browsers installed therein. For example, Google Chrome may be installed in the PC. At least one plug-in is installed on the browser, and the browser also has at least one font type. The plug-in and/or font on the browser may be obtained by a server device to generate a list of browser plug-ins and/or font types of the PC.

[0020] In some embodiments, a device library may be stored in a server device so that device fingerprints can be assigned to target devices by method 100 as described above. The device library stores at least one browser plug-in and/or font type of a second device. The browser plug-ins and/or font types of the target device can be compared with each of the browser plug-ins and the font types of each of second devices in the device library to calculate at least one correlation value. The details of calculating the correlation values will

be described below.

[0021] Fig. 1B shows a flow char of calculating correlation values in step 104. In step 104-1, each of the attributes of the target device is compared with each of the attributes of the second device stored in the device library to determine the correspondence therebetween. In step 104-2, the correlation values are calculated based on the comparison. After acquiring browser plug-ins and/or font types of the target device, they are compared with each of the browser plug-ins and the font types of each of second devices in the device library. The correlation values may be obtained from the comparison results.

[0022] For example, the browser plug-ins and/or font types of the target device is expressed as a list $I = [I_1, I_2, \dots, I_M]$, where I_i is an item of the attributes, e.g., a plug-in or a font type, and M is an integer equal to or greater than one. A list of browser plug-ins and/or font types of each of the second devices stored in the device library can be expressed as $I' = [I'_1, I'_2, \dots, I'_M]$. In some embodiments, the browser plug-in and/or font list I of the target device have the same sequence and category as those of items list I' of a second device stored in the device library, but the attribute values may be different. For example, although a browser plug-in and/or font type of the target device, e.g., I_2 , is not installed in the target device, I_2 is still kept in list I and assigned a value to show it is not installed so that list I can include an item corresponding to item I'_2 of list I' to facilitate the comparison.

[0023] Moreover, I_1 and the I'_1 correspond to each other as plug-ins or font types. That is, their attributes are the same but their values may be different. I'_1 is compared with I_1 to determine whether they have the same attribute, for example, whether they are both of a plug-in or a font. Depending on the comparison result, a comparison value is given to

represent the degree of their correlation. The comparison process will continue until all of the items in the lists are compared. Each comparison of items will be given a comparison value. In some embodiments, a correlation value may be obtained by adding all of the comparison values.

[0024] The present disclosure also contemplates that the attribute values of browser plug-ins and font types may be changed. In some embodiments, the calculation of correlation values may need to consider a change probability of those attribute values. Fig. 2 shows a flow chart of a method 200 for assigning a device fingerprint to an internet device that takes the changes of the attribute values into consideration when calculating correlation values, consistent with embodiments of the present disclosure. An internet device may be a mobile phone, a smart phone, a tablet, a laptop computer, a desktop computer, or any other electronic devices that can connect to the internet. Method 200 may be performed by a server device connected to the internet.

[0025] Referring to Fig. 2, method 200 includes the following steps. In step 202, a list of attributes of a target device to be assigned a device fingerprint is compiled. The attributes of the target device may include at least browser plug-ins and/or font types of the target device. For example, a server device may compile a list **I** of attributes including browser plug-ins and/or font types of the target device, where $I = [I_1, I_2, \dots, I_M]$, M being an integer equal to or greater than one. In some embodiments, list **I** may include M items of plug-ins and/or font types, some of which are commonly used in a browser.

[0026] In Step 204, changes of each of the items (attributes) in the list are determined based on a plurality of state transition vectors over a period of time. For example, when the

items in the list are browser plug-ins and/or fonts, the state transition vectors for these items may be installed, uninstalled, never installed, no change after installation, upgrade, or degrade. In some embodiments, the changes in state transition vectors of the items in the list are determined by observing the changes of a pre-determined number, e.g., one thousand, of sample devices, over a pre-determined period of time, e.g., one day.

[0027] In step 206, a change probability of the state transition vectors for each of the items in the list is determined. For example, in the illustrated embodiment above, one thousand sample devices are monitored to determine the change of each item $I_1, I_2, \dots, I_M$ of list I. For example, the state of each of the items $I_1, I_2, \dots, I_M$ changed to “installed” within a pre-determined time period, e.g., one day, is monitored. Similarly, the state of each of the items $I_1, I_2, \dots, I_M$ changed to “uninstalled,” “never installed,” “no change after installation,” “upgrade,” or “degrade” within a pre-determined time period may be monitored. The change probability is determined by the monitored data. For example, with respect to item I_1 , which may be a plug-in or font, if the plug-in or font was uninstalled in nine hundred sample devices out of one thousand over a period of one day, at the end of the period the change probability for the state transition vector “uninstalled” may be determined to be 90%.

[0028] In step 208, each item, i.e., an attribute, of the list I of the target device is compared with each item of a list of a second device, which is stored in a device library. For example, the device library stores at least one set of attributes of a second device that has been assigned a device fingerprint. The attributes, $I_1', I_2', \dots, I_M'$, of the second device may be compiled as a list $I' = [I_1', I_2', \dots, I_M']$. Each of items $I_1, I_2, \dots, I_M$ of list I is compared with a corresponding item $I_1', I_2', \dots, I_M'$ of list I' .

[0029] In step 210, based on the comparison of lists in step 208, state transition vectors and corresponding change probabilities are determined. In the illustrated embodiment above, item I_1 is compared with corresponding $I_{1'}$. If the state of I_1 indicates that a plug-in or font is installed in the browser of the target device and the state of $I_{1'}$ indicates that the plug-in or font is not installed in the browser of the second device, the state transition vector from I_1 to $I_{1'}$ may be determined to be “uninstall.” The change probability 90% of “uninstalled” as illustrated in step 206 may then be utilized. The comparison of $I_1, I_2, \dots, I_M$ with $I_{1'}, I_{2'}, \dots, I_{M'}$ results in M numbers of change probabilities for state transition vectors.

[0030] In step 212, a correlation value of list I of the target device and list I' of the second device is calculated based on state transition vectors and their corresponding change probability. For example, in the above illustrated embodiment, comparing $I_1, I_2, \dots, I_M$ with $I_{1'}, I_{2'}, \dots, I_{M'}$ results in M numbers of state transition vectors and their corresponding change probabilities. In some embodiments, a correlation value for list I and list I' may be obtained by adding each of the change probabilities. If P_{match} represents a correlation value, it can be obtained through the following function:

$$P_{match} = \sum_{i=1}^M \log(P_{i,j}); i \in [1, M], j \in [1, K],$$

wherein M represents the number of the attributes, e.g., browser plug-ins and/or font lists, and K represents the number of the state transition vectors.

[0031] When there are n numbers of second devices stored in the device library, the comparison between the target device and each of the second devices may results in n numbers of correlation values.

[0032] In step 214, the largest correlation value obtained in step 212 is selected and

compared with a pre-determined threshold value. In some embodiments, each of the correlation values obtained in step 212 may be compared with the pre-determined threshold value, without having to select the largest correlation value.

[0033] If the largest correlation value is equal to or greater than the pre-determined threshold value, in step 216, the device fingerprint of the second device corresponding to the largest correlation value is assigned to the target device. If the largest correlation value is smaller than the pre-determined threshold value, in step 218, the attributes of the target device including its browser plug-ins and font types are saved in the device library and a new device fingerprint is assigned to the target device. The new device fingerprint may include, for example, a device identification of the target device.

[0034] Fig. 3 shows an apparatus 300 for assigning device fingerprints to target devices, consistent with embodiments of the present disclosure. Apparatus 300 may be a server device that includes one or more processors and a memory device storing instructions executable by the one or more processors. Referring to Fig. 3, apparatus 300 includes an acquiring module 302, a matching module 304, and a processing module 306.

[0035] Acquiring module 302 is configured to acquire attributes including a browser plug-in and/or font list of a target device to be assigned a device fingerprint. Matching module 304 is configured to calculate a correlation value of the attributes of the target device with respect to attributes of at least one second device having a device fingerprint stored in a device library. In some embodiments, to calculate the correlation value, each of the attributes of the target device has a corresponding attribute of the at least one second device.

[0036] Processing module 306 is configured to assign a device fingerprint to the

target device based on the calculated correlation values. In some embodiments, processing module 306 compares the correlation value with a pre-determined threshold value. If the correlation value is greater than or equal to a pre-determined threshold value, processing module assigns the device fingerprint of the second device corresponding to the correlation value to the target device. If the correlation value is smaller than the pre-determined threshold value, processing module saves the attributes of the target device to the device library and assigns a new device fingerprint to the target device.

[0037] In some embodiments, apparatus 300 may further include a sampling module 308. Sample module 308 is configured to acquire a list of attributes including browser plug-ins and/or font types of a target device to be assigned a device fingerprint; determine the changes of each item in the lists of a plurality of sample devices over a pre-determined period of time based on a plurality of state transition vectors; and calculate a change probability of each item in the list corresponding to each state transition vector.

[0038] Persons skilled in the art can understand that, for convenience and brevity of the description, the specific processes performed by apparatus 300 and its modules 302, 304, 306, 308 have been described in methods 100 and 200 shown in Figs. 1A and 2 and related descriptions, and are not described again here. For example, referring to Figs. 1A, 2, and 3, acquiring module 302 may be configured to perform steps 102 and 202; matching module 304 may be configured to perform steps 104 and 212; processing module 306 may be configured to perform steps 106, 108, 110, 214, 216, and 218; and sampling module 308 may be configured to perform steps 204, 206, 208, and 210.

[0039] In the embodiments provided in the present disclosure, it shall be understood

that the disclosed apparatus and methods may be implemented in some other ways. For example, the apparatus embodiments described above are only schematic. The above modules are described by their functional divisions, while in practical implementation, a plurality of modules or components can be combined or integrated into another apparatus, or some features can be omitted or are not implemented. Further, each of the modules may be directly or indirectly coupled to each other through certain interfaces. The coupling or communication of apparatus or modules may be electrical, mechanical or in other forms.

[0040] The illustrated modules as separated component may or may not be physically separated. A module may be located at a place or it can also be distributed as multiple network modules. All or part of the modules therein can be selected according to the actual needs to realize the illustrated embodiment.

[0041] In some embodiments, various modules consistent with the embodiments of the present disclosure can be integrated into one processing module, or they can also be modules independently and physically existing, or they can be integrated into a module using two or more sub-modules. The above-mentioned integrated module can be achieved by means of hardware, or a combination of hardware and software.

[0042] The illustrated modules may be achieved by means of software function module stored in a computer readable storage medium, executable by one or more physical processors. The software function module may be stored in a storage medium, including instructions causing a computer device (such as a personal computer, a server, or a network device) or a processor to perform some of the steps of the methods described in various embodiments of the present disclosure. Moreover, the storage medium may include various

non-transitory media that can store program codes, such as a USB disk, a mobile hard drive, a Read-Only Memory (ROM), a Random Access Memory (RAM), a magnetic disc or optical disc.

[0043] Other embodiments of the disclosure will be apparent to those skilled in the art from consideration of the specification and practice of the disclosure disclosed here. This application is intended to cover any variations, uses, or adaptations of the disclosure following the general principles thereof and including such departures from the present disclosure as come within known or customary practice in the art. It is intended that the specification and examples be considered as exemplary only, with a true scope and spirit of the disclosure being indicated by the following claims.

[0044] It will be appreciated that the present disclosure is not limited to the exact construction that has been described above and illustrated in the accompanying drawings, and that various modifications and changes can be made without departing from the scope thereof. It is intended that the scope of the disclosure only be limited by the appended claims.

1. A computer-implemented method for assigning a device fingerprint to a target device, comprising:
 - acquiring (102; 202) first attributes of the target device, the first attributes including at least one browser plug-in;
 - determining (204) changes of the first attributes based on a plurality of state transition vectors over a period of time;
 - determining (206) change probabilities for the state transition vectors;
 - comparing (208) each of the first attributes with each of second attributes of at least one second device having a device fingerprint stored in a device library, the first attributes corresponding to the second attributes;
 - determining (210) a first state transition vector from the state transition vectors for each of the first attributes based on the comparison;
 - giving a first change probability to each of the first attributes based on the determined first state transition vector;
 - calculating (104; 212) a correlation value with respect to the first attributes of the target device and the second attributes of at least one second device wherein the correlation value is calculated based on the first change probability;
 - if the correlation value is greater than or equal to a pre-determined threshold value, assigning (108; 216) the device fingerprint of the at least one second device to the target device; and
 - if the correlation value is smaller than the pre-determined threshold value, storing (110; 218) the first attributes of the target device to the device library and assigning a new device fingerprint to the target device.

2. The method according to claim 1, wherein the state transition vectors include one or more of installation, uninstallation, never uninstalled, no change after installation, upgrade, or downgrade.
3. The method according to claim 1 or claim 2, wherein the new device fingerprint includes a device identification of the target device.
4. An apparatus for assigning a device fingerprint to a target device, comprising:
 - one or more processors; and
 - a memory device configured to store instructions executable by the one or more processors,wherein the one or more processors are configured to perform the method of any of the preceding claims.
5. A computer-readable storage medium storing instructions, which, when executed by a computer, cause the computer to perform the method of any of claims 1 to 3.