Technical Disclosure Commons

Defensive Publications Series

January 19, 2016

REDUCING GRANULARITY OF BROWSER FINGERPRINTING TECHNIQUES

Mike West

Dominic Battré

Recommended Citation

West, Mike and Battré, Dominic, "REDUCING GRANULARITY OF BROWSER FINGERPRINTING TECHNIQUES", Technical Disclosure Commons, (January 19, 2016)
http://www.tdcommons.org/dpubs_series/117

REDUCING GRANULARITY OF BROWSER FINGERPRINTING TECHNIQUES

ABSTRACT

A data transformation system transforms a data representation of an image displayed by an application such as a web browser. The system receives a data representation of an image. The system fetches a stored seed value associated with a user profile. The system performs a transformation of the data representation of the image based on the seed value. The seed value updates occasionally. Transforming the image using a sporadically-changing seed value reduces canvas-based fingerprinting granularity.

PROBLEM STATEMENT

Browser fingerprinting is a technique of identifying a user by extracting user identification information when the user visits various websites through web browsers. The fingerprinting information may incorporate, for example, browser and operating system types and versions, screen resolution, hardware configurations, and plugins. "Canvas fingerprinting" is a browser fingerprinting technique that allows websites to reliably generate a unique identifier for a user that the user cannot easily reset. Canvas fingerprinting enables persistent tracking across the web. The technique uses the browser to render a complex image, capture the actual pixel values, and hash the values to create the actual fingerprint data. The technique monitors unique font rendering in a predictable fashion based on machine-specific characteristics. The operating system, video card, video card drivers, monitor size, and other factors, all join together to produce slightly different results for the same code on two different computers. Canvas fingerprinting makes use of all these factors.

There are a number of things that users can do to block canvas fingerprinting. For example, users can disable read access to the canvas elements. However, as a wide variety of sites rely on being able to read data out of `<canvas>` elements, disabling read access to canvas elements may not be an optimum solution. Therefore, there are opportunities for alternate methods of defense against browser fingerprinting techniques that identify users.

DETAILED DESCRIPTION

The systems and techniques described in this disclosure relate to a data transformation system that transforms a data representation of an image displayed by an application such as a web browser. The system can be implemented for use in an Internet, an intranet, or another client and server environment. The system can be implemented locally on a client device or implemented across a client device and server environment. The client device can be any electronic device such as a mobile device, a smartphone, a tablet computer, a handheld electronic device, a desktop computer, a laptop computer, a networked gaming console, an electronic book, etc.

Fig. 1 illustrates an example method 100 to perform a transformation of a data representation of an image. The system receives 110 a data representation of an image. The image may be a complex image, a two dimensional image, a bitmap image, a binary image, text, etc. The image may be drawn in a drawable region defined in a programming language that is used to structure and present content in a user interface of an application, for example, a web browser application. The application may be associated with the user profile stored at a server of the data transformation system.

The data representation of the image may be in the form of x and y coordinates, height and width values, RGB (Red Blue Green) individual pixel values, etc. The form may be defined in a library associated with the programming language.

The programming language uses the data representation of the image to render images on the user interface of the application. The system may receive the data representation using a function written in another programming language.

The system fetches 120 a stored seed value associated with the user profile. The system associates a unique seed value with each of the stored user profiles. The seed value may be a random value or a fixed formula-based value. The system may generate the value when the user creates the user profile with the system. Alternatively, or additionally, the system may change the value every time a user session terminates with the application or when the user attempts to erase user activity retained in memory at the application such as by clearing a browser cache.

The system performs 130 a transformation of the data representation of the image based on the random seed value. The stored seed value may store decodable or direct information about the transformations that the system may perform on the data representation of the image. The system decodes or reads the information and accordingly performs one or more transformations. The transformations may include jitter into the data representation of the image that is not perceivable by a user. The transformation may include modifying individual pixels by tiny amounts, for example, modifying 132 from “rgb(255, 255, 255)” to “rgb(255, 255, 254),” skewing 135 the image by some marginal amount which may introduce sub-pixel artifacts, and/or scaling 137 the image by some marginal amount, for example, from 0.01% to -0.01%. Although Fig.1 shows three example transformations, 132, 135, and 135, the system is capable of performing various other possible transformations. The system may perform the one or more transformations cumulatively, sequentially, or both based on the stored seed value. The sequence may be defined in the stored seed value associated with the user profile.

The subject matter described herein this disclosure can be implemented in software and/or hardware (for example, computers, circuits, or processors). The subject matter can be implemented on a single device or across multiple devices (for example, a client device and a server device). Devices implementing the subject matter can be connected through a wired and/or wireless network. Such devices can receive inputs from a user (for example, from a mouse, keyboard, or touchscreen) and produce an output to a user (for example, through a display and/or a speaker). Specific examples disclosed are provided for illustrative purposes and do not limit the scope of the disclosure.

DRAWINGS

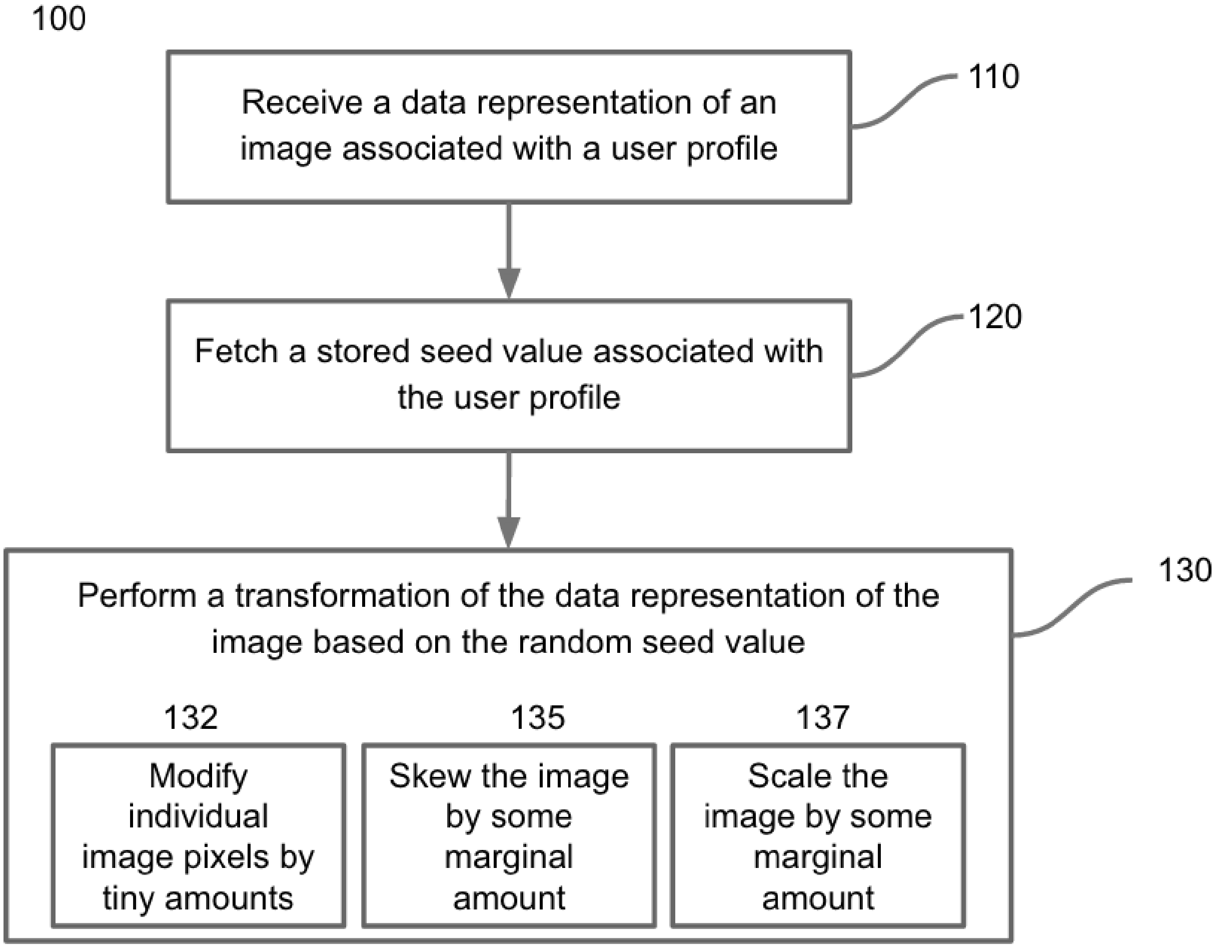
100
Receive a data representation of an image associated with a user profile
110
Fetch a stored seed value associated with the user profile
120
Perform a transformation of the data representation of the image based on the random seed value
130
132
Modify individual image pixels by tiny amounts
135
Skew the image by some marginal amount
137
Scale the image by some marginal amount

Fig. 1