November 26, 2016

Distributed Audience Lists

David Pattison

Distributed Audience Lists

An audience list is a list of users who may have performed some actions or have some common characteristics that are of interest to marketers, advertisers, or other entities. For example, an audience list can contain a set of users who may be interested in musicals because they have signed up for receiving the latest news or promotions for Broadway shows. Because of the sheer volume of information in the age of the Internet, countless audience lists each with an enormous number of users can be generated. Thus, storing large audience lists can take up significant storage resource. Furthermore, entities may maintain their own audience lists without sharing with the public and thus multiple duplicate efforts can be made and the lists are not verifiable. Thus, with users' consent, it can be beneficial to have the audience lists available to the public such that the information can be shared between different entities and can be verifiable.

This paper introduces a technique that solves the above mentioned problems by storing and retrieving audience lists in a distributed manner. According to the present technique, each user is associated with a pair of public key and private key. The user can be identified with the public key and maintains a private key that is not shared. In asymmetric cryptography, a public and private key pair comprise two uniquely related cryptographic keys. The public key is made available to the public, for example, via a publicly accessible repository or directory. The private key, on the other hand, must be kept confidential and only be known by its owner. The key pair can be generated using asymmetric encryption algorithms such as RSA (Rivest-Shamir-Adleman), Diffie-Hellman, ElGamal, Cramer-Shoup, etc. In some implementations, the keys can be generated based on a browser cookie or an Internet Protocol (IP) address associated with a user device.

4834-0162-7195.1

According to the present technique, audience lists are maintained using blockchain technology. A blockchain is a distributed database that maintains a continuously growing list of records called blocks that hold information. Each block in the blockchain can contain, among other data, a timestamp and a link to a previous block. In the present technique, users can execute client code to determine when an audience list is achieved. For example, with a user's consent, an application running on a client device of the user or a script transmitted from a server can be executed by the client device to cause the user to be added to an audience list maintained using a blockchain.

According to the present technique, a user can add his or her public key to a blockchain corresponding to the audience list by digitally signing a message with the associated private key. The message can contain the public key, the parent block identifier, and some other metadata. For example, the message can be generated by the application running on the client device of the user or the script transmitted from a server and executed by the client device. The message can be transmitted to a node in the blockchain and becomes a block in the blockchain. For example, the user can upload the message to a node in the blockchain or broadcast the message into the blockchain via a network.

After an audience list maintained using a blockchain is created, the audience list can be used to search a user. An entity (e.g., a data processing system, a content provider, a publisher) can download from a node in the network the entire blockchain corresponding to the audience list or can synchronize the audience list that has been previously downloaded with any update. For example, the blockchain can be searched sequentially for a public key associated with a user. If the public key is found in the list, it can be determined that the user is in the audience list. The

blockchain can also be searched using an index for fast lookups. The nodes that hold the blockchain can be a server associated with the audience list, a peer, or a third party.

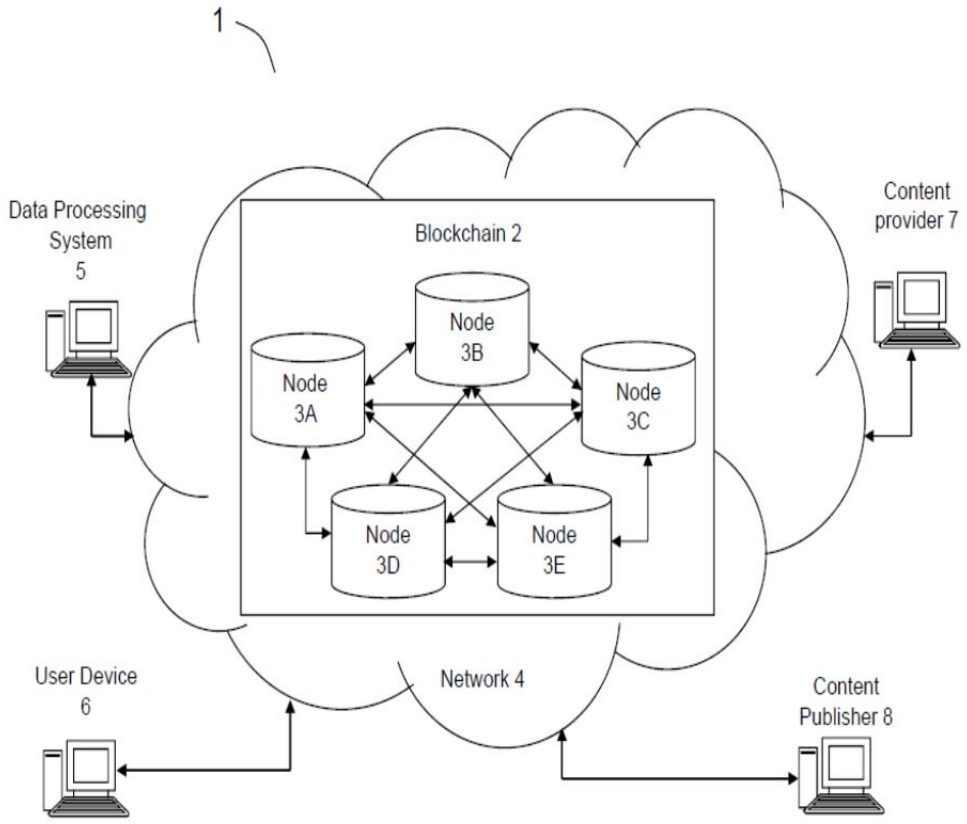

Figure 1

Figure 1 is a block diagram depicting an exemplary environment 1 for implementing the present technique. The environment 1 can include a blockchain 2 which includes a plurality of nodes 3A to 3E that can communicate with a data processing system 5, one or more user devices 6, one or more content provider devices 7, and one or more content publisher devices 8 via a network 4. The nodes 3A to 3E in the blockchain 2 can be peer nodes that connect to each other and can be located in different physical locations. Each of the nodes 3A to 3E can store a copy of a list of records or blocks, such as an audience list.

The network 4 can include one or more of any type of computer network such as the Internet, intranets, cellular network, WIFI network, WiMAX network, mesh network, Bluetooth, near field communication, satellite network, or other data network that facilitates communications between components in the environment 1. The network 4 can also include any number of computing devices (e.g., computer, servers, routers, network switches, etc.) that are configured to receive and/or transmit data within network 4. The network 4 can further include any number of hardwired and/or wireless connections. For example, the user device 6 can communicate wirelessly (e.g., via WiFi, cellular, radio, etc.) with a transceiver that is hardwired (e.g., via a fiber optic cable, a CAT5 cable, etc.) to other computing devices in network 4.

The data processing system 5 can include one or more devices, such as servers and nodes. The devices of the data processing system 5 can include one or more nodes 3A to 3E in the blockchain 2. The devices of the data processing system 5 can be located in the same physical location or in different physical locations. The devices of the data processing system 5 can include at least one processor (or a processing circuit) and a memory. The processor can include a microprocessor, application-specific integrated circuit (ASIC), field-programmable gate array (FPGA), etc., or combinations thereof. The memory can include, but is not limited to, electronic, optical, magnetic, or any other storage or transmission device capable of providing the processor with program instructions. The memory can further include a floppy disk, CD-ROM, DVD, magnetic disk, memory chip, ASIC, FPGA, read-only memory (ROM), random-access memory (RAM), electrically-erasable ROM (EEPROM), erasable-programmable ROM (EPROM), flash memory, optical media, or any other suitable memory. The memory stores machine instructions that, when executed by the processor, cause the processor to perform one or more of the operations described herein. The memory or storage devices of the one or more nodes 3A to 3E

that are included the data processing system 5 can store copies of audience lists or other lists of records.

The user device 6, the content provider device 7, and the content publisher device 8 can include desktop computers, laptop computers, tablet computers, smartphones, personal digital assistants, mobile devices, consumer computing devices, servers, clients, digital video recorders, a set-top box for a television, a video game console, or any other computing device configured to communicate via the network 4. The user device 6, the content provider device 7, and the content publisher device 8 can include a processor and a memory, i.e., a processing circuit. The memory stores machine instructions that, when executed by the processor, cause the processor to perform one or more of the operations described herein.

The user device 6 can communicate with the data processing system 5, the content provider device 7, and the content publisher device 8 via the network 4. The user device 6 can execute a software application (e.g., a web browser or a standalone application installed on the user device 6 such as a mobile application) to receive content from other devices over the network 4. The user device 6 can display data such as content provided by the content publisher device 8 (e.g., primary web page content or other information resources) and the content provided by the content provider device 7 (e.g., third party content items such as ads configured for display in a content slot of a web page). For example, the data processing system 5, upon receiving a request for third party content items from the user device 6, can transmit a third party content item provided by the content provider device 7 to the user device 6 for display at the user device 6. In some implementations, the data processing system 5 can send a script (e.g., JavaScript™) along with the content item to the user device 6. In other implementations, a script

(e.g., JavaScript™) provided by the data processing system 5 can be embedded in a web page provided by the content publisher device 8.

The script provided by the data processing system 5 can be executed by a processor of the user device 6 to add a user to an audience list. In some implementations, the script executed by the processor of the user device 6 can generate a pair of public key and private key for the user. For example, the key pair can be generated using asymmetric encryption algorithms such as RSA (Rivest-Shamir-Adleman), Diffie-Hellman, ElGamal, Cramer-Shoup, etc. In some implementations, the keys can be generated based on a browser cookie or an Internet Protocol (IP) address associated with the user device 6. The private key can be stored on the user device 6 or other places and is kept confidential. For example, only the user knows his or her private key. The script executed by the user device 6 can determine when an audience list membership is achieved. In some implementations, when the user performs certain actions, for example, completing a purchase or signing up a membership, the script executed by the user device 6 can query the user whether the user wishes to be added to a certain audience list. In some implementations, the user may wish to add himself or herself to an audience list without any query.

The script executed by the user device 6 can cause the user's public key to be added to a blockchain corresponding to an audience list. The script executed by the user device 6 can generate a message that includes the public key, the parent block identifier, and some other metadata. The parent block identifier identifies the block prior to the block associated with the user in the blockchain. The metadata can include additional fine-grained details of the user's actions associated with the audience list. For example, if the user is added to an audience list associated with an online purchase of a certain product, a shopping cart identifier associated with

the purchase can be added as metadata in the message. The user can sign the generated message with his or her private key. Once the message is signed, the script executed by the user device 6 can cause the signed message to be uploaded into the blockchain 2. It should be understood that the operations described herein above can also be performed by the user device 6 executing a standalone application (e.g., a mobile application) installed on the user device 6.

As described herein above, one or more of the nodes 3A to 3E in the blockchain 2 can be a device or server of the data processing system 5. Likewise, one or more of the nodes 3A to 3E in the blockchain 2 can also be nodes associated with the content provider device 7, the content publisher device 8, or other entities. Although Figure 1 shows five nodes in the blockchain, it should be understood that the blockchain 2 can include more, less, or different nodes from what are shown in Figure 1.

The data processing system 5, the content provider device 7, the content publisher device 8, or other entities can search an audience list to determine if a user is in the list. For example, as described herein above, the data processing system 5 can receive a request from the user device 6 for content item to present within a content slot of an information resource, such as a web page, at the user device 6. The data processing system 5 can download from a node in the network 4 the entire blockchain corresponding to the audience list or can synchronize the audience list that has been previously downloaded with any update. For example, a server in the data processing system 5 can download an audience list from one of the nodes 3A to 3E.

Continuing with the above example, in some implementations, the request received from the user device 6 by the data processing system 5 can include a public key associated with the user at the user device 6. In some implementations, the request received from the user device 6 can include an identifier, e.g., a browser cookie. The data processing system 5 can obtain the

public key associated with the user at the user device 6 based on the received identifier. The server in the data processing system 5 can search the audience list sequentially for the public key associated with a user. The audience list can also be searched using an index created by the data processing system 5 for fast lookups, for example when multiple requests are needed. If the public key is found in the list, the data processing system 5 can determine that the user is in the audience list. Once the data processing system 5 locates the block associated with the user in the blockchain, the data processing system 5 can obtain other information in the block, such as the metadata associated with the user. The obtained information can be used to select a suitable content item for display at the user device 6. For example, if the metadata in the block associated with the user indicates that the user is interested in golf, product or event information associated with golf can be displayed at the user device 6. Thus, with users' consent, by classifying users into categories using audience lists, users can be provided with more useful and needed information, and better conversions may be achieved.

For situations in which the systems discussed herein collect personal information about users, or may make use of personal information, the users may be provided with an opportunity to control whether programs or features that may collect personal information (e.g., information about a user's social network, social actions or activities, a user's preferences, or a user's current location), or to control whether or how to receive content from the content server that may be more relevant to the user. In addition, certain data may be treated in one or more ways before it is stored or used, so that certain information about the user is removed when generating parameters (e.g., demographic parameters). For example, a user's identity may be treated so that no identifying information can be determined for the user, or a user's geographic location may be generalized where location information is obtained (such as to a city, ZIP code, or state level), so

that a particular location of a user cannot be determined. Thus, the user may have control over how information is collected about the user and used by a server or node, such as the data processing system 5 or any nodes in the blockchain 2.

Accordingly, by using a distributed database such as a blockchain as described herein, as opposed to using a traditional centralized server and database, the present technique can reduce processing and storage cost for entities owning audience lists because other machines can be used to store the audience lists instead of the enteritis' fully company-owned machines. Furthermore, the present technique can provide transparency because, with users' consent, the audience lists can be available to the public and can be verifiable.

Abstract

This document describes a technique for storing and retrieving audience lists in a distributed manner by using a blockchain. Script executed by a user device can generate a pair of public and private keys for a user. The user can sign a message containing the public key and other information using the private key and upload the message to a blockchain corresponding to an audience list. A data processing system can search the blockchain for the public key and utilizing the information in the audience list maintained using the blockchain to better serve the user.