September 2021

AUTHENTIC TIMESTAMP PROVIDER

GOKUL ANAND MANIMARAN
VISA

TITLE: AUTHENTIC TIMESTAMP PROVIDER

VISA

GOKUL ANAND MANIMARAN

TECHNICAL FIELD

[0001] This disclosure relates to a method and a system for providing authenticity for a document with respect to timestamp.

BACKGROUND

[0002] If we have a document X, to prove that it existed at time t, X or Hash(X) can be stored in a centralized place maintained by a trusted third party, T. But the entire control of the document lies with T. If T is compromised, X loses its authenticity.

[0003] To overcome the above-mentioned shortcomings, the present invention facilitates authenticity with respect to timestamp wherein there is no control transferred to T. Also, it makes sure that, there is a distributed and chain of trust being established.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Additional advantages and details are explained in greater detail below with reference to the exemplary embodiments that are illustrated in the accompanying schematic figures, in which:

[0005] **Figs.1a-1b** illustrates a block diagram of a system for facilitating authenticity with respect to timestamp in accordance with some embodiments of the present disclosure.

[0006] **Fig.2** illustrates an example for authenticating a document with respect to timestamp in accordance with some embodiments of the present disclosure.

[0007] **Fig.3** illustrates a process for receiving signed documents in accordance with some embodiments of the present disclosure.

[0008] **Fig.4** illustrates a process for signing the hash in accordance with some embodiments of the present disclosure.

[0009] **Fig.5** illustrates a block diagram of an exemplary computer system for implementing embodiments consistent with the present disclosure

DESCRIPTION OF THE DISCLOSURE

[0010] It is to be understood that the present disclosure may assume various alternative variations and step sequences, except where expressly specified to the contrary. It is also to be understood that the specific devices and processes illustrated in the attached drawings and described in the following specification are simply exemplary and non-limiting embodiments or aspects. Hence, specific dimensions and other physical characteristics related to the embodiments or aspects disclosed herein are not to be considered as limiting.

[0011] No aspect, component, element, structure, act, step, function, instruction, and/or the like used herein should be construed as critical or essential unless explicitly described as such. Also, as used herein, the articles “a” and “an” are intended to include one or more items and may be used interchangeably with “one or more” and “at least one.” Furthermore, as used herein, the term “set” is intended to include one or more items (e.g., related items, unrelated items, a combination of related and unrelated items, and/or the like) and may be used interchangeably with “one or more” or “at least one.” Where only one item is intended, the term “one” or similar language is used. Also, as used herein, the terms “has,” “have,” “having,” or the like are intended to be open-ended terms. Further, the phrase “based on” is intended to mean “based at least partially on” unless explicitly stated otherwise.

[0012] As used herein, the terms “communication” and “communicate” may refer to the reception, receipt, transmission, transfer, provision, and/or the like of information (e.g., data, signals, messages, instructions, commands, and/or the like). For one unit (e.g., a device, a system, a component of a device or system, combinations thereof, and/or the like) to be in communication with another unit means that the one unit is able to directly or indirectly receive information from and/or transmit information to the other unit. This may refer to a direct or indirect connection (e.g., a direct communication connection, an indirect communication connection, and/or the like) that is wired and/or wireless in nature. Additionally, two units may be in communication with each other even though the information transmitted may be modified, processed, relayed,

and/or routed between the first and second unit. In some non-limiting embodiments or aspects, a message may refer to a network packet (e.g., a data packet and/or the like) that includes data. It will be appreciated that numerous other arrangements are possible.

[0013] As used herein, the term “computing device” may refer to one or more electronic devices that are configured to directly or indirectly communicate with or over one or more networks. A computing device may be a mobile or portable computing device, a desktop computer, a server, and/or the like. Furthermore, the term “computer” may refer to any computing device that includes the necessary components to receive, process, and output data, and normally includes a display, a processor, a memory, an input device, and a network interface. A “computing system” may include one or more computing devices or computers. An “application” or “application program interface” (API) refers to computer code or other data sorted on a computer-readable medium that may be executed by a processor to facilitate the interaction between software components, such as a client-side front-end and/or server-side back-end for receiving data from the client. An “interface” refers to a generated display, such as one or more graphical user interfaces (GUIs) with which a user may interact, either directly or indirectly (e.g., through a keyboard, mouse, touchscreen, etc.). Further, multiple computers, e.g., servers, or other computerized devices, such as an autonomous vehicle including a vehicle computing system, directly or indirectly communicating in the network environment may constitute a “system” or a “computing system”.

[0014] It will be apparent that systems and/or methods, described herein, can be implemented in different forms of hardware, software, or a combination of hardware and software. The actual specialized control hardware or software code used to implement these systems and/or methods is not limiting of the implementations. Thus, the operation and behavior of the systems and/or methods are described herein without reference to specific software code, it being understood that software and hardware can be designed to implement the systems and/or methods based on the description herein.

[0015] Some non-limiting embodiments or aspects are described herein in connection with thresholds. As used herein, satisfying a threshold may refer to a value being greater than the threshold, more than the threshold, higher than the threshold, greater than or equal to the threshold, less than the threshold, fewer than the threshold, lower than the threshold, less than or equal to the threshold, equal to the threshold, etc.

[0016] **Figs.1a-1b** illustrates a block diagram of a system for facilitating authenticity with respect to timestamp in accordance with some embodiments of the present disclosure.

[0017] As shown in **Fig.1a**, the system 100 comprises a client 101, a server 103 and a poller 115. At the client side 101, the client 101 may connect with the server 103 through a browser 105 application. The server 103 comprises of a signing server 107, a key generator server 109, a database (DB) server 111 and a publishing server 113. The poller 115 may be anything from simple script to software to download the contents of the publishing server 113. As shown in **Fig.1b** at the client side 101, the hash of the document is generated and sent to the signing server 107. The key generator server 109 may generate a public-private key pair at time "t" and sign all the hashes received at time "t". In an embodiment, the key generator server may prepare a document with all the hashes at time "t". Thereafter, the publishing server 113 may be configured to publish the document and the public key to the internet. In an embodiment, the signing server 107 may send the digital signature along with timestamp to the client. The client may store the digital signature along with the timestamp.

In an embodiment, during the verification phase of the digital signature, the client 101 receives the public key from the server 103 for the time "t" and verifies the hash from the document and the signature.

As an example, as the document "A" is hashed as Hash (A) =86588509E as shown in **Fig.2**. At the server 103, the signing server 107 receives all the hashes at different time intervals such as time t=0-n. The public key is generated at the server 103 at a time t as AD81. The client 101 receives the digital signature along with time stamp as shown in **Fig.2**.

Fig.3 illustrates a process for receiving signed documents in accordance with some embodiments of the present disclosure.

As illustrated in **Fig.3**, the client 101 may have a document which may comprise confidential information. As an example, the information may be about a new planet or a new star constellation or a formula for a new drug. The client 101 calculates the hash for this document at client side 101. The hash may be calculated using existing hash generating mechanisms. The client 101 may visit the signing entity's website and may submit

the document. The aspect of server 103 signing the hash upon receiving the document is illustrated in **Fig.4**. Once the signing is performed, the client 101 may download the digital signature which is timestamped.

Fig.4 illustrates a process for signing the hash in accordance with some embodiments of the present disclosure.

As illustrated in **Fig.4,** the signing server 107 receives the hash of the document and fetches the private key from the key generating server 109. The key generating server 109 may generate a public-private key pair for the digital signature. The public key may be stored in the database table 1. In an embodiment, the database table 2 may store the hash value being signed. The publishing server 113 may fetch the data from both the database tables i.e., the data associated with public key and the hash value. Upon fetching this information, the publishing server 113 may publish the document which contains the hashes signed and the public key at time “t” which anyone can poll. The pollers may actively poll the data from the publishing server 113. As an example, the pollers may be any user or an entity.

Since communities and companies are pulling the public key data, there is no way for the third party (T) to change the private key existed at time t. Even if “T” is compromised, a new document can be included at time t, but the documents signed at t will still be valid. To make sure new documents cannot be included, the private keys will not be stored anywhere. So even if “T” is not trusted directly, if Company A is trusted and Company A trusts T, then the documents signed by T can also be trusted by pulling the public key polled by company A with respect to T, at time t.

[0018] Some of the advantages of the present disclosure are listed below.

[0019] The present disclosure achieves providing authenticity with respect to timestamp.

[0020] The present disclosure facilitates a distributed and chain of trust and hence avoids losing the authenticity of the document when a trusted entity is compromised. Privacy, since we are not storing or not even obtaining the original document

[0021] **Fig.5** illustrates a block diagram of an exemplary computer system for implementing embodiments consistent with the present disclosure.

[0022] In an embodiment, the computer system 500 may be used to implement the system. The computer system 500 may include a central processing unit (“CPU” or “processor”) 502. The processor 502 may include at least one data processor for facilitating authenticity with respect to timestamp. The processor 502 may include specialized processing units such as, integrated system (bus) controllers, memory management control units, floating point units, graphics processing units, digital signal processing units, etc.

[0023] The processor 502 may be disposed in communication with one or more input/output (I/O) devices (512 and 513) via I/O interface 501. The I/O interface 501 employ communication protocols/methods such as, without limitation, audio, analog, digital, monoaural, radio corporation of America (RCA) connector, stereo, IEEE-1394 high speed serial bus, serial bus, universal serial bus (USB), infrared, personal system/2 (PS/2) port, bayonet neill-concelman (BNC) connector, coaxial, component, composite, digital visual interface (DVI), high-definition multimedia interface (HDMI), radio frequency (RF) antennas, S-Video, video graphics array (VGA), IEEE 802.11b/g/n/x, Bluetooth, cellular e.g., code-division multiple access (CDMA), high-speed packet access (HSPA+), global system for mobile communications (GSM), long-term evolution (LTE), worldwide interoperability for microwave access (WiMax), or the like, etc.

[0024] Using the I/O interface 501, the computer system 500 may communicate with one or more I/O devices such as input devices 512 and output devices 513. For example, the input devices 512 may be an antenna, keyboard, mouse, joystick, (infrared) remote control, camera, card reader, fax machine, dongle, biometric reader, microphone, touch screen, touchpad, trackball, stylus, scanner, storage device, transceiver, video device/source, etc. The output devices 513 may be a printer, fax machine, video display (e.g., cathode ray tube (CRT), liquid crystal display (LCD), light-emitting diode (LED), plasma, plasma display panel (PDP), organic light-emitting diode display (OLED) or the like), audio speaker, etc.

[0025] In some embodiments, the processor 502 may be disposed in communication with a communication network 509 via a network interface 503. The network interface 503 may communicate with the communication network 509. The network interface 503 may employ connection

protocols including, without limitation, direct connect, ethernet (e.g., twisted pair 10/100/1000 Base T), transmission control protocol/internet protocol (TCP/IP), token ring, IEEE 802.11a/b/g/n/x, etc. The communication network 509 may include, without limitation, a direct interconnection, local area network (LAN), wide area network (WAN), wireless network (e.g., using Wireless Application Protocol), the Internet, etc. Using the network interface 503 and the communication network 509, the computer system 500 may communicate with a database 514. The network interface 503 may employ connection protocols include, but not limited to, direct connect, ethernet (e.g., twisted pair 10/100/1000 Base T), transmission control protocol/internet protocol (TCP/IP), token ring, IEEE 802.11a/b/g/n/x, etc.

[0026] The communication network 509 includes, but is not limited to, a direct interconnection, a peer to peer (P2P) network, local area network (LAN), wide area network (WAN), wireless network (e.g., using Wireless Application Protocol), the Internet, Wi-Fi and such. The communication network 509 may either be a dedicated network or a shared network, which represents an association of the different types of networks that use a variety of protocols, for example, hypertext transfer protocol (HTTP), transmission control protocol/internet protocol (TCP/IP), wireless application protocol (WAP), etc., to communicate with each other. Further, the communication network 509 may include a variety of network devices, including routers, bridges, servers, computing devices, storage devices, etc.

[0027] In some embodiments, the processor 502 may be disposed in communication with a memory 505 (e.g., RAM, ROM, etc. not shown in FIGURE 5) via a storage interface 504. The storage interface 504 may connect to memory 505 including, without limitation, memory drives, removable disc drives, etc., employing connection protocols such as, serial advanced technology attachment (SATA), integrated drive electronics (IDE), IEEE-1394, universal serial bus (USB), fiber channel, small computer systems interface (SCSI), etc. The memory drives may further include a drum, magnetic disc drive, magneto-optical drive, optical drive, redundant array of independent discs (RAID), solid-state memory devices, solid-state drives, etc.

[0028] The memory 505 may store a collection of program or database components, including, without limitation, user interface 506, an operating system 507, etc. In some embodiments, computer system 500 may store user/application data, such as, the data, variables, records, etc.,

as described in this disclosure. Such databases may be implemented as fault-tolerant, relational, scalable, secure databases such as Oracle or Sybase.

[0029] The operating system 507 may facilitate resource management and operation of the computer system 500. Examples of operating systems include, without limitation, Apple™ Macintosh ™ OS X™, UNIX™, Unix-like system distributions (e.g., Berkeley Software Distribution (BSD), FreeBSD™, Net BSD™, Open BSD™, etc.), Linux distributions (e.g., Red Hat™, Ubuntu™, K-Ubuntu™, etc.), International Business Machines (IBM™) OS/2™, Microsoft Windows™ (XP™, Vista/7/8, etc.), Apple iOS™, Google Android™, Blackberry™ operating system (OS), or the like. In some embodiments, the computer system 500 may implement web browser 508 stored program components. Web browser 508 may be a hypertext viewing application, such as Microsoft™ Internet Explorer™, Google Chrome™, Mozilla Firefox™, Apple™ Safari™, etc. Secure web browsing may be provided using secure hypertext transport protocol (HTTPS), secure sockets layer (SSL), transport layer security (TLS), etc. Web browsers 508 may utilize facilities such as AJAX, DHTML, Adobe™ Flash, Javascript, Application Programming Interfaces (APIs), etc.

[0030] The illustrated steps are set out to explain the exemplary embodiments shown, and it should be anticipated that ongoing technological development will change the manner in which particular functions are performed. These examples are presented herein for purposes of illustration, and not limitation. Further, the boundaries of the functional building blocks have been arbitrarily defined herein for the convenience of the description. Alternative boundaries can be defined so long as the specified functions and relationships thereof are appropriately performed. Alternatives (including equivalents, extensions, variations, deviations, etc., of those described herein) will be apparent to persons skilled in the relevant art(s) based on the teachings contained herein. Such alternatives fall within the scope and spirit of the disclosed embodiments. Also, the words "comprising," "having," "containing," and "including," and other similar forms are intended to be equivalent in meaning and be open ended in that an item or items following any one of these words is not meant to be an exhaustive listing of such item or items, or meant to be limited to only the listed item or items. It must also be noted that as used herein, the singular forms “a,” “an,” and “the” include plural references unless the context clearly dictates otherwise.

[0031] Furthermore, one or more computer-readable storage media may be utilized in implementing embodiments consistent with the present disclosure. A computer readable storage medium refers to any type of physical memory on which information or data readable by a processor may be stored. Thus, a computer readable storage medium may store instructions for execution by one or more processors, including instructions for causing the processor(s) to perform steps or stages consistent with the embodiments described herein. The term “computer readable medium” should be understood to include tangible items and exclude carrier waves and transient signals, i.e., are non-transitory. Examples include random access memory (RAM), read-only memory (ROM), volatile memory, nonvolatile memory, hard drives, CD ROMs, DVDs, flash drives, disks, and any other known physical storage media.

[0032] Finally, the language used in the specification has been principally selected for readability and instructional purposes, and it may not have been selected to delineate or circumscribe the inventive subject matter. Accordingly, the disclosure of the embodiments of the disclosure is intended to be illustrative, but not limiting, of the scope of the disclosure.

AUTHENTIC TIMESTAMP PROVIDER

ABSTRACT

The present disclosure relates to providing authenticity for a document with respect to timestamp. In an embodiment, if a document is created at particular time, Hash(X) can be stored in a centralized place maintained by a trusted third party. But it transfers control over to Third party, to prove the authenticity with respect to the time at which document was created, which means, if that third party is compromised, the document loses its authenticity. For every second t, a public and private key pair is generated by a third party (T). When Hash(X) is given to T at time t, it encrypts Hash(X)+t by the private key (digital signature) and sends back the response. At the same time, the public key is made available for everyone on the internet. So, to verify the authenticity of X, the digital signature provided by T, will be decrypted with the public key generated at time t. Since the public key is published to the internet and anyone can poll that data, if the document verifier don't trust T directly, but trusts some Company A that in turn trusts T, he can verify the document by getting public key polled by A from T at time t (indirectly forming a chain of trust).

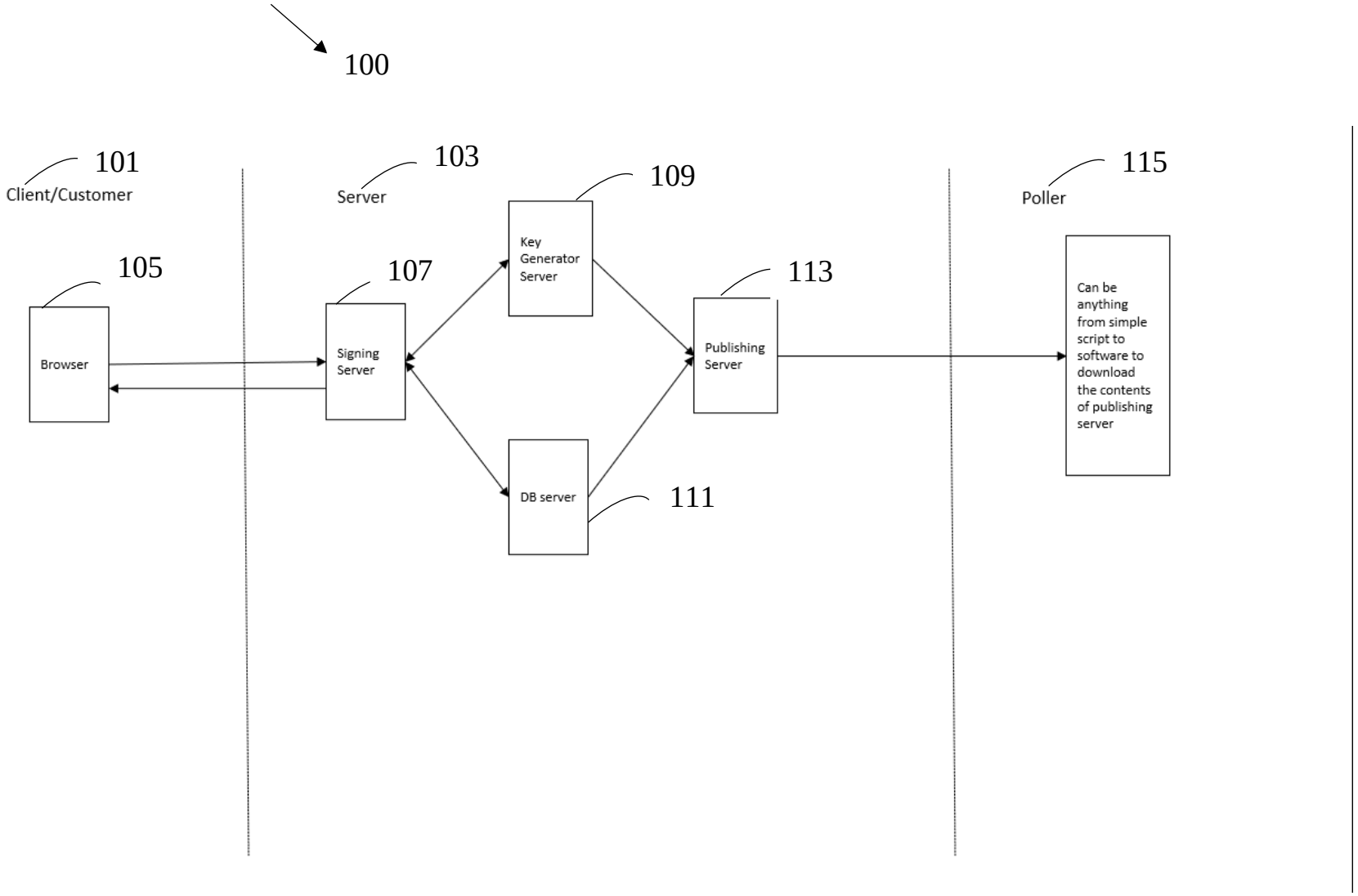
100
101
Client/Customer
103
Server
109
115
Poller
105
107
113
Key Generator Server
Browser
Signing Server
Publishing Server
Can be anything from simple script to software to download the contents of publishing server
DB server
111

Fig.1a

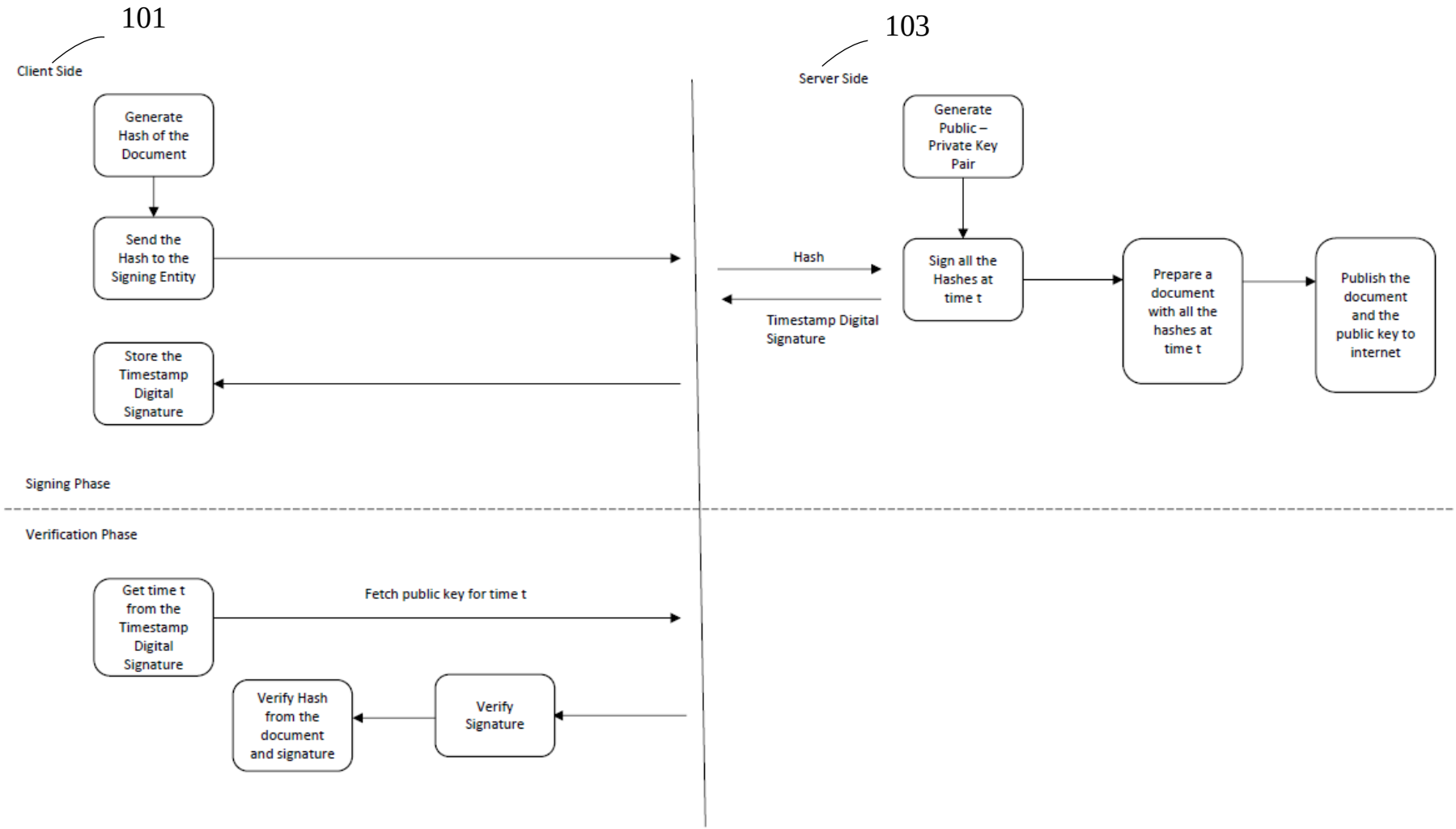
101
Client Side
103
Server Side
Generate Hash of the Document
Send the Hash to the Signing Entity
Store the Timestamp Digital Signature
Generate Public – Private Key Pair
Hash
Timestamp Digital Signature
Sign all the Hashes at time t
Prepare a document with all the hashes at time t
Publish the document and the public key to internet
Signing Phase
Verification Phase
Get time t from the Timestamp Digital Signature
Fetch public key for time t
Verify Hash from the document and signature
Verify Signature

Fig.1b

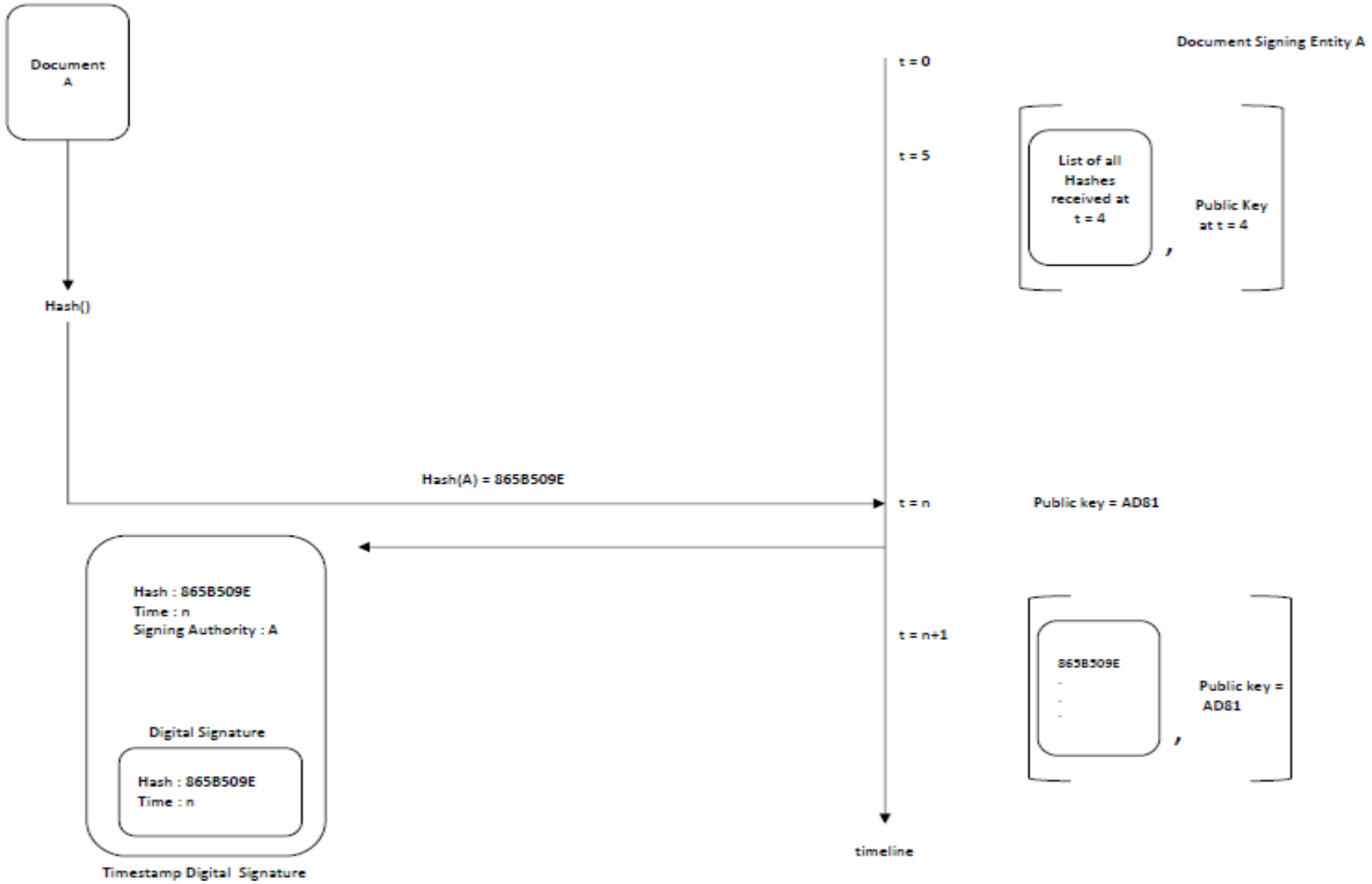
Document
A
Hash()
Hash(A) = 865B509E
Hash : 865B509E
Time : n
Signing Authority : A
Digital Signature
Hash : 865B509E
Time : n
Timestamp Digital Signature
t = 0
t = 5
t = n
t = n+1
timeline
Document Signing Entity A
List of all
Hashes
received at
t = 4
Public Key
at t = 4
Public key = AD81
865B509E
Public key =
AD81

Fig.2

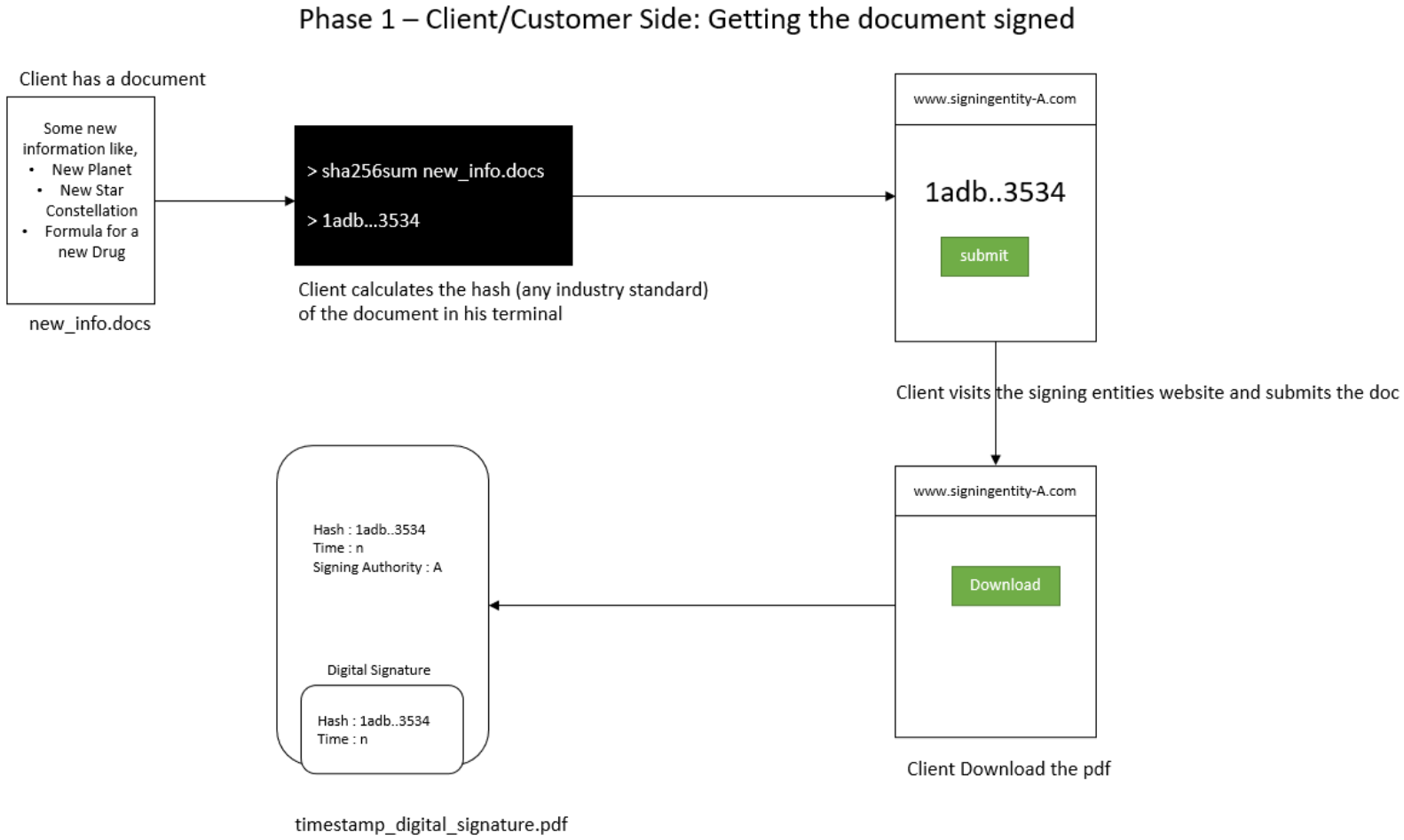
Phase 1 – Client/Customer Side: Getting the document signed
Client has a document
Some new information like,
• New Planet
• New Star Constellation
• Formula for a new Drug
new_info.docs
> sha256sum new_info.docs
> 1adb...3534
Client calculates the hash (any industry standard) of the document in his terminal
www.signingentity-A.com
1adb..3534
submit
Client visits the signing entities website and submits the doc
www.signingentity-A.com
Download
Client Download the pdf
Hash : 1adb..3534
Time : n
Signing Authority : A
Digital Signature
Hash : 1adb..3534
Time : n
timestamp_digital_signature.pdf

Fig.3

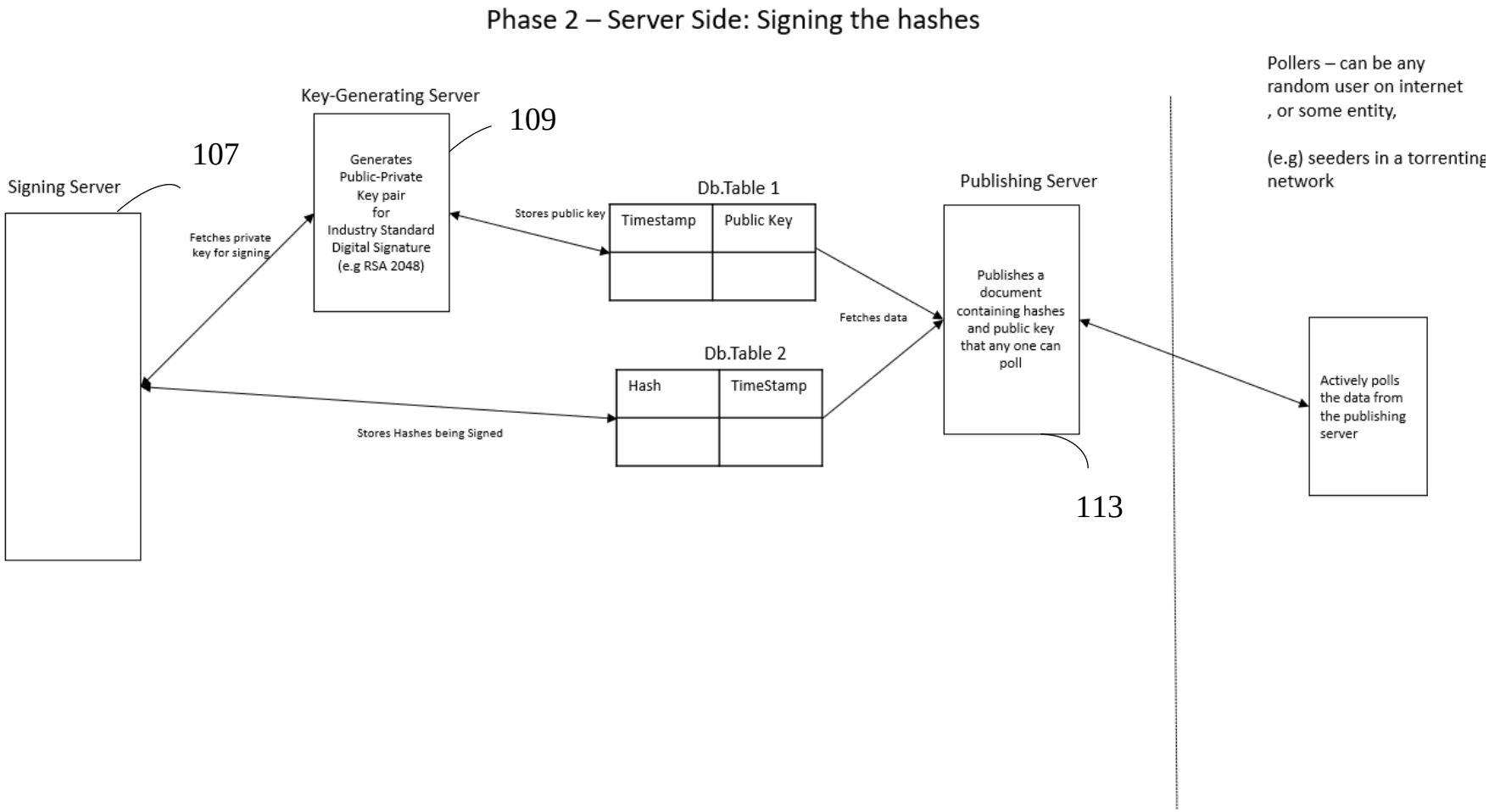
Phase 2 – Server Side: Signing the hashes
Pollers – can be any random user on internet , or some entity,
(e.g) seeders in a torrenting network
Key-Generating Server
109
107
Signing Server
Generates Public-Private Key pair for Industry Standard Digital Signature (e.g RSA 2048)
Fetches private key for signing
Stores public key
Db.Table 1
Timestamp
Public Key
Publishing Server
Publishes a document containing hashes and public key that any one can poll
Fetches data
Db.Table 2
Hash
TimeStamp
Stores Hashes being Signed
Actively polls the data from the publishing server
113

Fig.4

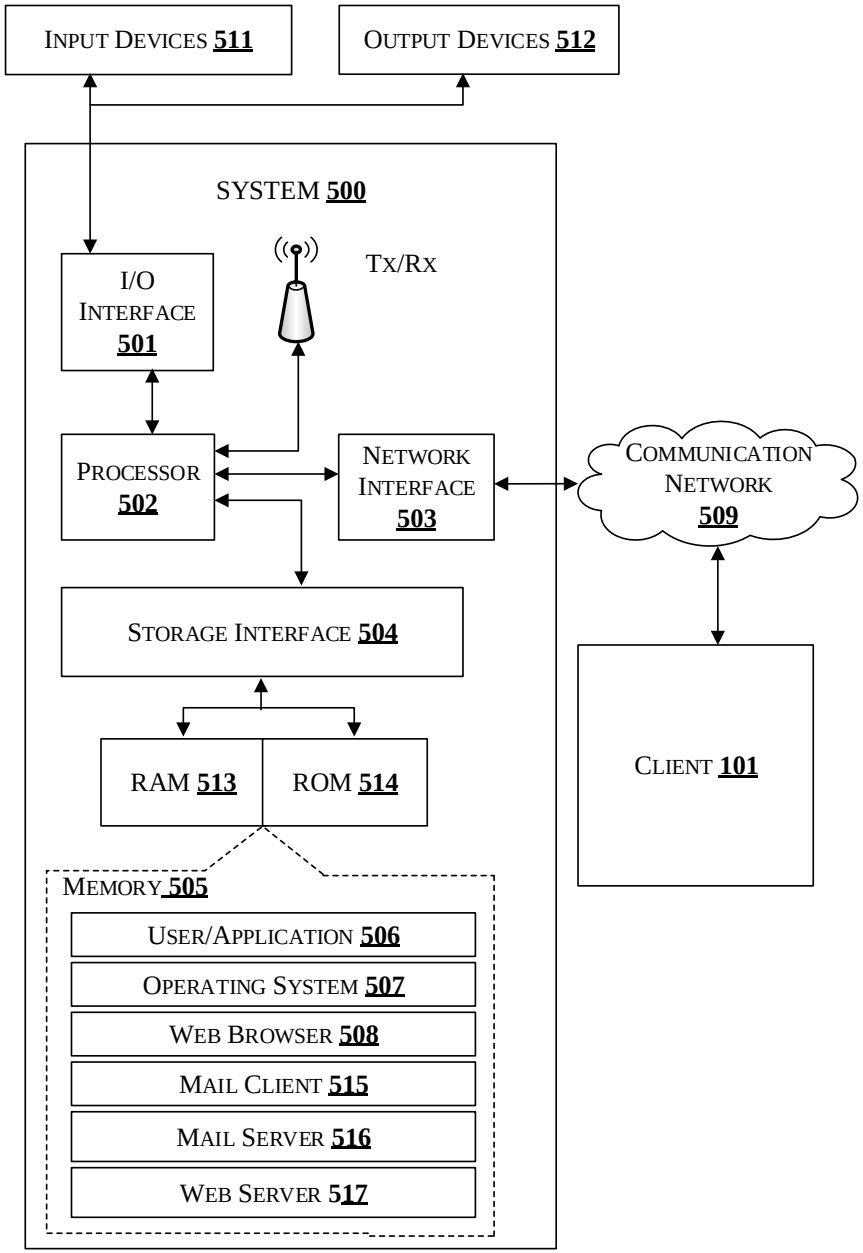
INPUT DEVICES 511
OUTPUT DEVICES 512
SYSTEM 500
I/O INTERFACE 501
TX/RX
PROCESSOR 502
NETWORK INTERFACE 503
COMMUNICATION NETWORK 509
STORAGE INTERFACE 504
RAM 513
ROM 514
CLIENT 101
MEMORY 505
USER/APPLICATION 506
OPERATING SYSTEM 507
WEB BROWSER 508
MAIL CLIENT 515
MAIL SERVER 516
WEB SERVER 517

Fig.5