(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 April 2018 (12.04.2018)

WIPO | PCT

(10) International Publication Number
WO 2018/067710 A1

(51) International Patent Classification:
H04L 29/12 (2006.01)

(21) International Application Number:
PCT/US2017/055156

(22) International Filing Date:
04 October 2017 (04.10.2017)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

62/404,723	05 October 2016 (05.10.2016)	US
15/389,276	22 December 2016 (22.12.2016)	US
15/389,302	22 December 2016 (22.12.2016)	US
15/389,314	22 December 2016 (22.12.2016)	US

(63) Related by continuation (CON) or continuation-in-part (CIP) to earlier application:
US 15/389,276 (CON)
Filed on 22 December 2016 (22.12.2016)

(71) Applicant: **AMAZON TECHNOLOGIES, INC.** [US/US]; P.O. Box 81226, Seattle, WA 98108-1226 (US).

(72) Inventors: **UPPAL, Hardeep, Singh**; 410 Terry Avenue North, Seattle, WA 98109-5210 (US). **VASQUEZ, Jorge**; 410 Terry Avenue North, Seattle, WA 98109-5210 (US). **HOWARD, Craig, Wesley**; 410 Terry Avenue North, Seattle, WA 98109-5210 (US). **RADLEIN, Anton, Stephen**; 410 Terry Avenue North, Seattle, WA 98109-5210 (US).

(74) Agent: **ANDERSON, Maria, Culic**; Knobbe, Martens, Olson & Bear, LLP, 2040 Main Street, 14th Floor, Irvine, CA 92614 (US).

(54) Title: NETWORK ADDRESSES WITH ENCODED DNS-LEVEL INFORMATION

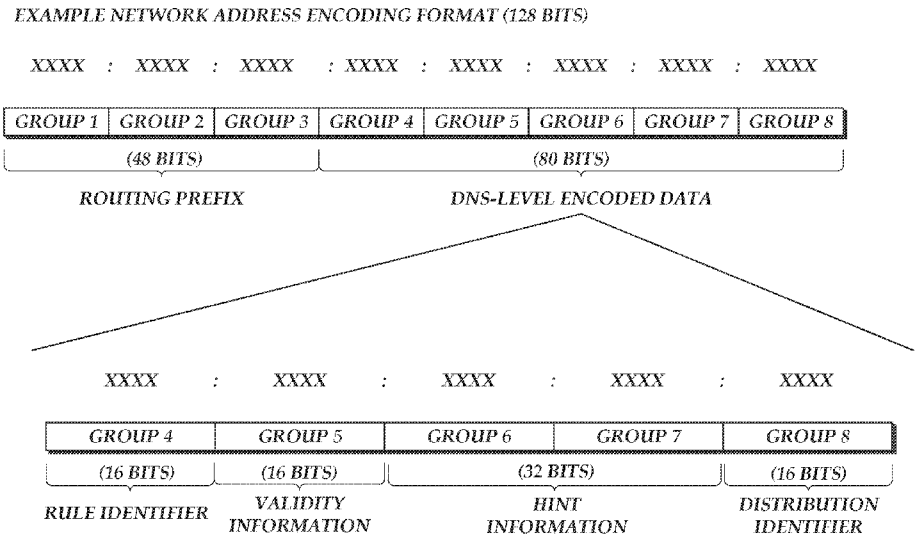

Fig. 4

(57) Abstract: Systems and methods are described to enable a DNS service to encode information into a network address to be advertised by the DNS service. Information encoded by a DNS service may include, for example, an identifier of a content set to which the network address corresponds (e.g., a domain name) and validity information, such as a digital signature, that verifies the validity of the network address. On receiving a request to communicate with the network address, a destination device associated with the network address may decode the encoded information within the network address to assist in processing the request. In some instances, the encoded information may be used to identify malicious network transmissions, such as transmissions forming part of a network attack, potentially without reliance on other data, such as separate mappings or contents of the data transmission.

WO 2018/067710 A1

[Continued on next page]

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

NETWORK ADDRESSES WITH ENCODED DNS-LEVEL INFORMATION

INCORPORATION BY REFERENCE TO ANY PRIORITY APPLICATIONS

[0001] Any and all applications for which a foreign or domestic priority claim is identified in the Application Data Sheet as filed with the present application are hereby incorporated by reference under 37 CFR 1.57.

BACKGROUND

[0002] Generally described, computing devices utilize a communication network, or a series of communication networks, to exchange data. Companies and organizations operate computer networks that interconnect a number of computing devices to support operations or provide services to third parties. The computing systems can be located in a single geographic location or located in multiple, distinct geographic locations (e.g., interconnected via private or public communication networks). Specifically, data centers or data processing centers, herein generally referred to as "data centers," may include a number of interconnected computing systems to provide computing resources to users of the data center. The data centers may be private data centers operated on behalf of an organization or public data centers operated on behalf, or for the benefit of, the general public.

[0003] Service providers or content creators (such as businesses, artists, media distribution services, etc.) can employ a series of interconnected data centers to deliver content (such as web sites, web content, or other digital data) to users or clients. These interconnected data centers are sometimes referred to as "content delivery networks" (CDNs) or content delivery systems. Existing routing and addressing technologies can enable multiple data centers associated with a content delivery system to provide similar or identical content to client computing devices. In some instances, each data center providing a set of content may be referred to as a point-of-presence ("POP"). A content delivery system can maintain POPs over a wide area (or worldwide) to enable the system to efficiently service requests from clients in a variety of locations.

[0004] To provide access to content, server computing devices are generally associated with network addresses, enabling requests for communication with the server computing devices to be routed across a network. These network addresses are generally formed

by a series of bits formatted according to a defined protocol, such as the Internet Protocol (IP) version four (IPv4) or version 6 (IPv6). Because network addresses may be difficult for humans to recognize and remember, CDNs can utilize resolution systems that function to resolve human-readable identifiers into corresponding network addresses. One example of such a resolution system is the Domain Name System (DNS), which functions to resolve domain names into corresponding network addresses.

[0005] Generally, DNS services and CDNs operate independently and according to differing protocols. For example, DNS requests from a client computing device are generally routed via the DNS protocol through multiple DNS components to a DNS service, where a network address corresponding to a domain name is returned via the DNS protocol. When a network address is obtained, the client computing device may then communicate with a server computing device associated with the network address (e.g., via the hypertext transport protocol, or "HTTP"). The respective processes of DNS resolution and communication with a network address are therefore generally logically distinct, such that a server computing device obtaining a request for content at a network address is unaware of a prior DNS resolution (if any) that facilitated the request.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. 1 is a block diagram depicting an illustrative logical network including multiple accessing computing devices 102 and content providers 104, as well as a content delivery system 110 including a DNS service and multiple POPs;

[0007] FIG. 2 is a block diagram depicting an illustrative configuration of one embodiment of a server that may implement the DNS service of FIG. 1;

[0008] FIG. 3A is a block diagram depicting interactions between an accessing computing device and the DNS service of FIG. 1 to resolve a distribution identifier into a network address including encoded DNS-level information;

[0009] FIG. 3B is a block diagram depicting interactions between an accessing computing device and a POP of FIG. 1 to provide content corresponding to a distribution identifier via a network address including encoded DNS-level information; and

[0010] FIG. 4 is an illustrative visualization of a format for encoding DNS-level information into a network address.

DETAILED DESCRIPTION

[0011] Generally described, aspects of the present disclosure relate to the utilization of specifically formatted network addresses in order to facilitate the retrieval and transmission of content on a content delivery system. More particularly, aspects of the present disclosure enable a domain name system (DNS) service to generate and use network addresses that include encoded information available to the DNS service at least at a time that a client computing device requests to resolve an identifier, such as a domain name, into a corresponding network address. Such information may be generally referred to herein as "DNS-level information" or "DNS-level data," and may include, for example, a domain name that a client computing device requested to resolve, security information (such as transport security layer ["TLS"] or secure sockets layer ["SSL"] certificates associated with a domain name), validity information (such as time-to-live, or "TTL," information), or other information available at a DNS service. In particular, a DNS service may encode all or a portion of such DNS-level information within a network address advertised by the DNS service (e.g., included within DNS records provided by the DNS service), and thus make such DNS-level information available to routers or other computing devices that receive requests to communicate with the network address. In this manner, these routers or other computing devices may gain access to the DNS-level information, which would otherwise not be communicated to the routers or other computing devices in a traditional request addressed to a network address. Use of such information may enable more efficient operation of such routers or other computing devices. For example, a router or computing device may directly obtain a domain name to which a request relates from a network address of the request, without relying on external information or mappings. As another example, a router or computing device may determine whether a request to communicate with a network address should be considered valid, based on validity information encoded into the network address. Should the request be invalid (e.g., due to an expired TTL), the request can be handled as erroneous or potentially malicious, thus enabling the router or computing device to determine validity as a function of a network address, potentially without referencing external information regarding the request. Thus, aspects of the present disclosure significantly increase the amount of information available to a computing device or router based on a network address,

and enable such routers or computing devices to more efficiently handle network communications by utilizing this newly-available information.

[0012] Aspects of the present disclosure may be utilized, for example, by a content delivery system hosting content of a multitude of entities. Such a content delivery system can function to provide client computing devices with access to a variety of different sets of content hosted by the content delivery system on behalf of third-party content providers or an operator of the content distribution service itself. Such a set of content can be generally referred to as a "distribution," and may correspond to a specific web site (e.g., as associated with a specific domain name) or other network-accessible service. Content delivery systems generally make distributions available at one or more network addresses (e.g., internet protocol or "IP" addresses), to which a computing device—either legitimate or malicious—may transmit a request for information. Traditionally, such network addresses are very limited in the amount of information they convey, because the network addresses are generally formatted to identify one or more destination computing devices on a network, and in some instances to assist in routing requests to those destination computing devices. However, due to the limited number of possible network addresses under some formats, such as IPv4, it was common to utilize the same network address to provide multiple sets of content (e.g., web sites). Thus, the same IPv4 address may identify an endpoint for many different distributions, which may or may not be associated with one another. Such scenarios presented numerous difficulties. For example, in a scenario where a specific web site was under a denial of service ("DOS") attack, data packets used to implement the attack could be addressed to a network address associated with many different web sites, making it difficult or impossible to determine the specific web site under attack based solely on the network address to which malicious data packets were addressed. Prior patent filings, such as U.S. Patent Application No. 14/815,863, entitled "IDENTIFYING TARGETS OF NETWORK ATTACKS," and filed 07/31/2015, the entirety of which is hereby incorporated by reference, sought to address such shortcomings by, for example, providing web sites with unique or substantially unique collections of network addresses.

[0013] The present application provides an additional or alternative solution to the above-noted problems, among others. Specifically, embodiments of the present disclosure can utilize one or more encoding rules to encode DNS-level data, such as a domain name to which a DNS request is directed, into a network address. Thus, a router or other computing device

receiving a data transmissions addressed to the network address may decode the network address itself in order to obtain the encoded DNS-level data. Illustratively, encoding of DNS-level data into a network address can enable routers or other computing devices to determine, from a network address to which a data packet is addressed, what domain name was included within a DNS request that preceded generation of the data packet. Advantageously, DNS-level data may therefore be determined subsequent to DNS resolution, without relying on independent or additional communication with a DNS service, and without relying on the content of data packets themselves (which, particularly in the case of network attacks, may be malformed). For example, a computing device may determine that malicious data packets transmitted to a given network address are the result of a DNS resolution of a particular domain name, and thus may further determine that the distribution associated with the domain name is the target of a network attack. The computing device may then take mitigating action to reduce or eliminate negative effects of the network attack.

[0014] Because the present disclosure enables encoding of DNS-level information into network addresses, embodiments of the present disclosure may be particularly suited for implementation with respect to network address formats having a large address space. One example of such a network address format is the IPv6 format, which utilizes 128 bit network addresses. The large number of potential network addresses within the IPv6 format (particularly in comparison to the IPv4 format) can enable DNS-level information to be encoded without negatively impacting the traditional identification and routing functions of network addresses.

[0015] While examples are provided above enabling domain names to be encoded into network addresses, other DNS-level information may additionally or alternatively be encoded into a network address under the present disclosure. For example, a DNS service may encode within a network address "hint information," that enables a router or other computing device (e.g., a server within a POP) to more efficiently service requests to communicate with the network address. Illustratively, the hint information may include an identifier of a security certificate (e.g., and SSL or TLS certificate) associated with a domain name to which the request is directed. Accordingly, when a computing device receives a request to communicate with a network address including encoded hint information, the computing device may decode the hint information, determine the security certificate identified by the hint information, and establish a secure communication channel with a client computing device. In some instances, encoding of

hint information within a network address may reduce or eliminate the need for a router or other computing device (e.g., of a CDN) to maintain a mapping of network addresses to associated security certificates, or to inspect data packets for information identifying a relevant security certificate (e.g., a "host header" within an HTTP packet). Because decoding of hint information within a network address may require less processing power or memory usage than would packet inspection or using lookups to external mappings, use of hint information can reduce the computing resources used by a router or other computing device.

[0016] Additional examples of hint information may include a network address or other information of a device from which a DNS resolution request was received (e.g., an accessing computing device, a DNS resolver used by that accessing computing device, etc.). In some instances, components of a CDN may use such information to determine how to handle subsequent requests to access content, stemming from an initial DNS resolution request. For example, where request to access content results in a "cache miss" at a POP of a CDN (e.g., where an accessing computing device requests content not stored at the POP), a POP can be configured to retrieve the requested information from another server (sometimes referred to as an "origin server"). Where the content is available at multiple origin servers, the POP may in some instances use hint information to select an appropriate origin server. For example, where a content request is addressed to a network address with hint information identifying a particular geographic or network location (e.g., as the region from which a corresponding DNS request originated), the POP may select an origin server nearby to that geographic or network location from which to obtain the relevant content.

[0017] In some embodiments, DNS-level information may further include validity information encoded into a network address by a DNS service. When such validity information is encoded into a network address, and a client computing device attempts to communicate with a destination computing device (e.g., a server within a POP) associated with the network address, that destination computing device may verify the validity information prior to initiating communication with the client computing device. In this way, a DNS service and a destination computing device may work cooperatively to ensure that all client computing devices accessing the destination computing device are legitimate users of the DNS service. Such cooperation may assist, for example, in mitigating network attacks. For example, some types of DoS attack function by instructing a first device to resolve a domain name of an attack target into an IP

address using a DNS service, and then instructing a large number of additional devices to transmit malicious data packets to the IP address. These additional devices do not interact directly with the DNS service, but instead repeat IP address information maintained by the first device. Thus, embedding validity information within an IP address that is tied specifically to the first device may enable an attack target (or other computing device) to distinguish between the first device and the additional device, substantially reducing the volume of attack traffic.

[0018] Further, network attacks are in some instances "non-re-resolving," such that resolution of a domain name of an attack target into a network address occurs only once (or very infrequently). Thus, changes to DNS records for an attack target, such as changing the network address of the attack target, can function to redirect legitimate traffic to a new network address while malicious traffic continues to flow to the defunct network address. To reduce the effect of such malicious traffic, some systems may modify configurations of routers or other computing devices to halt processing of traffic directed to a defunct network address. Such halting is sometimes referred to as "blackholing" the defunct network address, and generally involves discarding or "dropping" packets addressed to the defunct network address, either at a receiving computing device or at an intermediary network device. Traditional creation of a blackhole network address generally involves propagation of new routing information (e.g., routing tables, such as a forwarding information base or "FIB") across a network, a process that introduces additional traffic into a network and therefore can utilize significant computing resources. By inclusion of validity information into a network address, blackholing can be automated without requiring distribution of new routing information across a network. Rather, each device on a network may be configured such that any packet with invalid validity information is automatically dropped by a router or other computing device. In some instances, the protocol by which validity information is generated or verified may be modified over time, such as on an hourly basis. Thus, a network attacks that continue after modification of validity information, without re-resolving a domain name into a corresponding valid network address, would result in data packets being transmitted to an invalid network address, and such data packets could be automatically dropped within a network. Thus, the inclusion of validity information within a network address can enable computing devices to readily and efficiently distinguish legitimate from illegitimate traffic.

[0019] While illustrative examples of DNS-level information are provided above, a network address encoded according to embodiments of the present disclosure may include additional or alternative DNS-level data. For example, a DNS service may in some instances encode a unique "request identifier" into each network address provided in response to a DNS request, in order to uniquely identify content requests that stem from the DNS request. Thereafter, data can be collected from both the DNS service and corresponding POPs of a CDN, in order to determine correlations between DNS requests and subsequent content requests. For example, the CDN may determine that a given DNS request resulted in *n* content requests to the network address referenced in the DNS request. Thus, the examples of DNS-level data provided herein are illustrative in nature.

[0020] To enable encoding and decoding of information within a network address, a DNS service and a destination computing device (e.g., a device associated with a network address identified within a DNS record, which maps a domain name or other identifier to a network address) can each maintain one or more rules indicating how information is encoded within a network address. The rules may specify, for example, which bits of a network address represent different types of DNS-level information, and how that information is represented. Illustratively, a rule may specify that the last 16 bits of a network address represent an identifier of a distribution associated with the network address (e.g., a domain name or other unique identifier of the distribution). A rule may further specify that the penultimate 16 bits of a network address represent validity information for the network address, hint information, etc. For each type of information encoded within a network address, the rule may specify how that information is encoded. For example, a domain name may be encoded by processing the domain name according to a hash algorithm, to result in a hash value of the corresponding number of bits. Thereafter, the hash value may be included in a network address as an encoded domain name. In some instances, information may be encoded according to a cryptographic hash algorithm, such as an algorithm of the Secure Hash Algorithm family (e.g., SHA-2 or SHA-3) or an algorithm of the Message-Digest Algorithm family (e.g., MD6). For example, validity information may be generated by passing some set of inputs (e.g., values of the remaining fields encoded within a network address, a current time, an identifier of a client computing device, etc.) through a cryptographic hash to produce a digest, and then encrypting the digest with a private key to produce a digital signature that can be included in the validity information. The process

of utilizing cryptographic hashes and encoding via private keys to result in digital signatures is sometimes referred to as "public key cryptography" or "asymmetric cryptography." This process is known in the art and therefore will not be described in detail herein. However, with application to the embodiments of the present disclosure, the use of public key cryptography to generate digital signatures for inclusion in a network address can provide substantial certainty that any valid network address (e.g., with a verifiable digital signature) was obtained by interaction with a DNS service that had access to a corresponding private key.

[0021] In some instances, all or a portion of a network address may be encrypted, to avoid conveyance or modification of information to unauthorized parties. For example, DNS-level data (e.g., including a digital signature value) may be encrypted according to public key encryption. Illustratively, a DNS service may utilize a public key associated with a content server to encrypt DNS-level data of a network address (or any given portion of a network address), and the content server, on receiving data addressed to the network address, may utilize a corresponding private key to decrypt the portion and utilize the DNS-level data (or other data represented by the portion) in provide the requested information. As a further illustration, a router of the CDN may utilize a corresponding private key to decrypt a network address to which data is addressed, and utilize a decrypted value to route the data on the CDN. Encryption of DNS-level data (or other values represented in network addresses) may be beneficial in preventing authorized entities from obtaining knowledge of actual network addresses utilized within the CDN (e.g., where the decrypted network address represents a network address of a content server), or obtaining knowledge of DNS-level data encoded within a network address. In some embodiments, digital signatures and encryption of network address information may be used in conjunction to provide confidentiality, integrity, and authenticity of the network address. To ensure routability of data addressed to an encrypted network addresses, a portion of such network addresses (e.g., a routing prefix) may be left encrypted.

[0022] In some instances, network addresses may be formatted to enable an alteration of the rules for encoding information within those network addresses. For example, one or more bits of a network address may be assigned to represent a version identifier, indicative of a format of the network address. On receiving a request from a client computing device to communicate with a network address, a destination device may inspect the version identifier to determine corresponding rules, indicative of how DNS-level information is encoded within the network

address. Thus, a DNS service and corresponding CDN may alter encoding rules for network addresses periodically, or in some instances utilize multiple different encoding formats simultaneously. In one embodiment, the DNS service and CDN may identify "high risk" distributions, associated with a higher likelihood of being targeted in a network attack, and provide a distinct set of encoding rules for network address of these distributions. These rules, for example, may include stronger validity information, or may by altered more rapidly than network addresses of other, non-high risk distributions. In some instances, the rules may specify a similar or identical encoding format of information, but may vary other parameters used in such encoding. For example, the rules may specify a particular public/private key pair to utilize in encoding network addresses. Version information encoded within a network address can enable the DNS service and CDN to determine the rules under which a given network address has been encoded.

[0023] While the above description refers generally to a DNS service and corresponding destination device (e.g., as part of a CDN) that each maintain knowledge of rules for encoding network addresses, some embodiments of the present disclosure may utilize a distinct service to maintain such rules. Illustratively, both a DNS service and destination devices of a CDN may be provided with access to an application programming interface (API) of a network address encoding service, such that the DNS service and destination devices may transmit requests to the network address encoding service to generate network addresses for a given distribution or to decode DNS-level information from a network address. Use of such a network address encoding service may, for example, simplify requirements to synchronize rules between a DNS service and destination devices.

[0024] While examples are provided herein with respect to content distribution systems, embodiments of the present application may be implemented with respect to any network of computing devices that operates to serve discrete sets of content to client computing devices. Moreover, while some examples are provided with respect to a content distribution network as a whole, embodiments of the present application may also be implemented in whole or in part by discrete portions of the content delivery system. Thus, the examples provided herein are intended to be illustrative, and not exhaustive, in nature.

[0025] As will be appreciated by one of skill in the art in light of the description above, the embodiments disclosed herein substantially increase the ability of computing systems,

such as content delivery systems, to handle network communications. Specifically, embodiments disclosed herein enable more efficient routing or communication handling by providing hint information directly within a network address, reducing or eliminating the need for routing or destination devices to maintain such information separately. Moreover, embodiments disclosed herein enable identification of a specific set of content (a distribution) to which transmissions are directed, based on the network address of the transmission and regardless of the contents of the transmission. Still further, embodiments disclosed herein enable inclusion of validity information, such as a digital signature, within a network address, to assist in distinguishing legitimate communications from invalid and potentially malicious communications. Thus, the presently disclosed embodiments represent an improvement in the functioning of such computing systems, by enabling content delivery systems or other networked devices to more efficiently route and handle communications, to more efficiently identify and discard malicious communications, and to continue to service legitimate client requests even while receiving large numbers of illegitimate requests. Moreover, the presently disclosed embodiments address technical problems inherent within computing systems; specifically, the limited ability of computing systems to process network-based requests, the reliance of traditional network routing technologies on external information when routing requests to communicate with a network address, and the disassociation between a DNS service and a destination network device caused by commonly used network communication protocols. These technical problems are addressed by the various technical solutions described herein, including the encoding at a DNS service of DNS-level data within a network address, the decoding of such information at a destination device, and the use of such information to handle requests to communicate with the network address. Thus, the present application represents a substantial improvement on existing network systems and computing systems in general.

[0026] The foregoing aspects and many of the attendant advantages of the present disclosure will become more readily appreciated as the same become better understood by reference to the following, when taken in conjunction with the accompanying drawings.

[0027] FIG. 1 is a block diagram depicting an illustrative logical network 100 including multiple accessing computing devices 102 and multiple content providers 104 in communication with a content delivery system 110 via a network 106. As shown in FIG. 1, the content delivery system 110 includes a DNS service 112 enabling accessing computing

devices 102 to resolve identifiers of distributions on the content delivery system 110 (e.g., domain names) into network addresses. The content delivery system 110 further includes one or more points-of-presence (POPs) providing access to content of the distributions on the content delivery system 110. While the accessing computing devices 102 and the content providers 104 are shown as a group within FIG. 1, the accessing computing devices 102 and content providers 104 may be geographically distant, and independently owned or operated. For example, the accessing computing devices 102 could represent a multitude of users in various global, continental, or regional locations accessing the content delivery system 110. Further, the content providers 104 could represent a multitude of related or distinct parties that have associated with the content delivery system 110 to provide content, such as web sites, multimedia, or other digital, network-deliverable content to the accessing computing devices 102. Accordingly, the groupings of accessing computing devices 102 and content providers 104 within FIG. 1 is intended to represent a logical, rather than physical, grouping. Similarly, each of the components of the content delivery system 110 may be located within geographically diverse areas. For example, the DNS service 112 and POPs 124 within the content delivery system 110 may be globally, continentally, or regionally disparate, in order to provide a wide geographical presence for the content delivery system 110.

[0028] Network 106 may be any wired network, wireless network, or combination thereof. In addition, the network 106 may be a personal area network, local area network, wide area network, cable network, satellite network, cellular telephone network, or combination thereof. In the example environment of FIG. 1, network 106 is a global area network (GAN), such as the Internet. Protocols and components for communicating via the other aforementioned types of communication networks are well known to those skilled in the art of computer communications and thus, need not be described in more detail herein. While each of the accessing computing devices 102, content providers 104, and content delivery system 110 is depicted as having a single connection to the network 106, individual components of the accessing computing devices 102, content providers 104, and content delivery system 110 may be connected to the network 106 at disparate points. Accordingly, communication times and capabilities may vary between the components of FIG. 1.

[0029] Accessing computing devices 102 may include any number of different computing devices capable of communicating with the content delivery system 110. For

example, individual accessing computing devices may correspond to a laptop or tablet computer, personal computer, wearable computer, server, personal digital assistant (PDA), hybrid PDA/mobile phone, mobile phone, electronic book reader, set-top box, camera, digital media player, and the like. Further, accessing computing devices 102 may include devices utilized by both legitimate clients of the content delivery system 110 and devices utilized by malicious parties to undertake network-based attacks, such as DoS attacks, on the content delivery system 110.

[0030] Content providers 104 may include any computing device owned or operated by an entity that has provided content to the content delivery system 110 for subsequent transmission to client computing devices (which may include one or more accessing computing devices 102). For example, content providers 104 may include servers hosting web sites, streaming audio, video, or multimedia services, data analytics services, or other network-accessible services. While illustratively shown in FIG. 1 as a network-attached computing device, content providers 104 may additionally or alternatively provide content to the content delivery system 110 via non-networked communication channels (e.g., via physical delivery of data).

[0031] The content provided to the content delivery system 110 by the content providers 104 may be wholly or partially hosted in POPs 120. Each POP 120 may include a variety of computing devices configured to serve content to accessing computing devices 102. Specifically, each POP 120 may include one or more content servers 122 associated with network addresses to which accessing computing devices 102 may address requests to obtain content from the content delivery system 110. Each POP 120 may further include a data cache 126 on which such content can be stored. Because the amount of storage space of each data cache 126 may be limited, content servers 122 of the POPs 112 can be configured to detect "cache misses"—instances in which data requested by an accessing computing device 102 is not available in the data cache 126—and to retrieve such data from another device (e.g., the content providers 104 or other data stores not shown in FIG. 1). Thus, the content servers 112 may be logically viewed as "edge" devices, providing a first-level cache of content on the content delivery system 110. The general operation of content servers 112 within a POP 120 to obtain and provide content is known within the art, and therefore will not be described in detail herein. In accordance with embodiments of the present disclosure, each POP 120 may further include a

decoding rules data store 124 configured to store one or more rules indicating how DNS-level data is encoded into network addresses. As will be described below, such rules may indicate the specific bits of a network address that represent different types of DNS-level information, and the algorithms or encoding by which such information is encoded.

[0032] Each POP 120 may be associated with a number of network addresses via which accessing computing devices 102 may address the POP 120 via the network 106. In one embodiment, each network address is an internet protocol version 6 (IPv6) address. For example, each POP 114 may be associated with one or more specific "blocks" of network addresses, such as the "2001:db8::/48" block of IPv6 addresses (represented in classless inter-domain routing or "CIDR" notation, as is common in the art). Further, each POP 120 may be configured to provide multiple sets of content, each of which may be associated with a specific content provider 104. Generally, these discrete sets of content may be referred to herein as a "distribution." Each distribution may represent a specific network-accessible service, such as a web site, available from the POP 114 or the content delivery system 110 generally. As described below, network addresses encoded with DNS-level information may include a first section, sometimes referred to as a routing prefix, corresponding to the block of network addresses associated with a POP 120. Thus, communications transmitted to these network addresses via the network 106 would be expected to arrive at the POP 120 associated with the routing prefix. One or more remaining bits of such network addresses (e.g., other than the routing prefix) may be used to designate a distribution of the content delivery system 110 from which content is requested, as well as other information utilizable by the content servers 112 to handle requests to communicate with the network addresses.

[0033] The illustrative content delivery system 110 of FIG. 1 further includes a DNS service 112 configured to enable accessing computing devices 102 to resolve human-readable network identifiers of distributions (such as domain names or URLs) into network addresses (such as IP addresses) encoded with DNS-level information, to which requests to access content of a distribution may be transmitted. Specifically, the DNS service 112 includes one or more DNS servers 114, each including one or more processors, memories, and data storage devices collectively configured to receive requests from accessing computing devices 102 for a specific domain name associated with a distribution. The DNS servers 112 may further be configured, in response, to provide one or more network addresses, associated with one or more computing

devices within a POP 120, at which content of that distribution may be obtained. In accordance with embodiments of the present disclosure, such network addresses may include encoded DNS-level information (e.g., information available at the DNS service 112 at least at the time of the request). Thereafter, the accessing computing device 114 may communicate with the POP 120, via the network addresses, to access the distribution. In one embodiment, network addresses corresponding to an identifier of content on the content distribution system 110 (a "distribution identifier," which may correspond, for example, to a domain name or uniform resource identifier ["URI"]) may be included within DNS records pre-generated by the DNS service 112, and stored within a DNS record data store 119. Illustratively, one or more DNS servers 114 may generate DNS records based on one or more rules included within the encoding rules data store 116, which rules specify the format of network addresses encoded with DNS-level information, as well as information regarding a distribution to which the network address should correspond. Such information may be stored, for example, within the distribution data store 118, and may comprise prefix information for one or more POPs 120 at which the distribution can be accessed, hint information for the distribution (e.g., an identifier of a security certificate applicable to the distribution, an identifier of a resolver from which a corresponding DNS request was received or a location of that resolver, etc.), identifiers of rules applicable to the distribution, etc. For example, the DNS service 112 may identify a prefix associated with a POP 120 from which the distribution can be accessed in order to determine the first portion of a network address for the distribution. The DNS service 112 may further utilize the one or more rules to determine a set of bits corresponding to an identifier of the distribution, a set of bits corresponding to hint information for the distribution, a set of bits corresponding to validity information for the network address, and a set of bits corresponding to a version identifier for the one or more rules. The DNS service 112 may then concatenate the prefix, set of bits for the identifier, set of bits for the hint information, set of bits for the validity information, and the set of bits for the version identifier to result in a network address in an applicable format specified under the one or more rules (e.g., IPv6). The DNS server 112 may then store a DNS record associating the identifier of the distribution with the generated network address with the DNS record data store 119. When a request is obtained from an accessing computing device 102 to resolve the identifier of the distribution, the DNS server 112 may retrieve a DNS record corresponding to the identifier (e.g., identifying one or more network addresses mapped to the identifier), and return the DNS record

to the accessing computing device 102. In some instances, a DNS server 112 may generate DNS records for a distribution periodically (e.g., every hour). In other instances, a DNS server 112 may generate DNS records for a distribution "on-the-fly," in response to requests from accessing computing devices 102 to resolve a distribution identifier. Such on-the-fly record generation may be beneficial, for example, in order to allow for use of request-specific information during generation of a network address. For example, in some instances, a DNS server 114 may use a region identifier associated with a resolution request (e.g., an identifier of a geographical or logical region from which the resolution request stemmed) as an input to an algorithm for generating validity information for a network address, such that POPs 120 respond to requests to communicate with the network address only if such requests stem from the same region.

[0034] The encoding rules data store 116, distribution data store 118, and DNS record data store 119 of the DNS service 112, as well as the decoding rules data store 124 and data cache 126 of the POPs 120, may correspond to any persistent or substantially persistent data storage, such as a hard drive (HDD), a solid state drive (SDD), network attached storage (NAS), a tape drive, or any combination thereof.

[0035] As noted above, while the encoding and decoding of DNS-level data into or from network addresses is generally described with reference to the DNS server 112 and POPs 120, either or both the DNS server 112 and POPs 120 may in some embodiments utilize an external address encoding service (not shown in FIG. 1) to encode or decode DNS-level data into or from network addresses. For example, the address encoding may include one or more servers configured to provide an API or other interface through which requests may be received to encode DNS-level data into a network address of a distribution, or to decode DNS-level data from a network address. Such servers may be in communication with a rules data store, and utilize rules from such a data store to encode or decode network addresses, in accordance with embodiments of the present disclosure.

[0036] It will be appreciated by those skilled in the art that the content delivery system 110 may have fewer or greater components than are illustrated in FIG. 1. In addition, the content delivery system 110 could include various web services and/or peer-to-peer network configurations. Thus, the depiction of the content delivery system 110 in FIG. 1 should be taken as illustrative. For example, in some embodiments, components of the content delivery system 110, such as the DNS server 112, may be executed by one more virtual machines

implemented in a hosted computing environment. A hosted computing environment may include one or more rapidly provisioned and released computing resources, which computing resources may include computing, networking and/or storage devices. A hosted computing environment may also be referred to as a cloud computing environment.

[0037] Any one or more of the DNS servers 112, encoding rules data store 116, DNS record data store 119, distribution data store 118, content servers 122, decoding rules data store 124, and data cache 126 may be embodied in a plurality of components, each executing an instance of the respective DNS servers 112, encoding rules data store 116, DNS record data store 119, distribution data store 118, content servers 122, decoding rules data store 124, and data cache 126. A server or other computing component implementing any one of DNS servers 112, encoding rules data store 116, DNS record data store 119, distribution data store 118, content servers 122, decoding rules data store 124, and data cache 126 may include a network interface, memory, processing unit, and computer readable medium drive, all of which may communicate which each other may way of a communication bus. The network interface may provide connectivity over the network 106 and/or other networks or computer systems. The processing unit may communicate to and from memory containing program instructions that the processing unit executes in order to operate the respective DNS servers 112, encoding rules data store 116, DNS record data store 119, distribution data store 118, content servers 122, decoding rules data store 124, and data cache 126. The memory may generally include random access memory (RAM), read only memory (ROM), other persistent and auxiliary memory, and/or any non-transitory computer-readable media.

[0038] FIG. 2 depicts one embodiment of an architecture of a server 200 that may implement a DNS server 114 of the DNS service 112 or other components described herein. The general architecture of server 200 depicted in FIG. 2 includes an arrangement of computer hardware and software components that may be used to implement aspects of the present disclosure. As illustrated, the server 200 includes a processing unit 204, a network interface 206, a computer readable medium drive 207, an input/output device interface 220, a display 222, and an input device 224, all of which may communicate with one another by way of a communication bus. The network interface 206 may provide connectivity to one or more networks or computing systems, such as the network 106 of FIG. 1. The processing unit 204 may thus receive information and instructions from other computing systems or services via a

network. The processing unit 204 may also communicate to and from memory 210 and further provide output information for an optional display 222 via the input/output device interface 220. The input/output device interface 220 may also accept input from the optional input device 224, such as a keyboard, mouse, digital pen, etc. In some embodiments, the server 200 may include more (or fewer) components than those shown in FIG. 2. For example, some embodiments of the server 200 may omit the display 222 and input device 224, while providing input/output capabilities through one or more alternative communication channel (e.g., via the network interface 206).

[0039] The memory 210 may include computer program instructions that the processing unit 204 executes in order to implement one or more embodiments. The memory 210 generally includes random access memory (RAM), read only memory (ROM) and/or other persistent or non-transitory memory. The memory 210 may store an operating system 214 that provides computer program instructions for use by the processing unit 204 in the general administration and operation of the server 200. The memory 210 may further include computer program instructions and other information for implementing aspects of the present disclosure. For example, in one embodiment, the memory 210 includes user interface software 212 that generates user interfaces (and/or instructions therefor) for display upon a computing device, e.g., via a navigation interface such as a web browser installed on the computing device In addition, memory 210 may include or communicate with one or more auxiliary data stores, such as data store 226, which may correspond to any persistent or substantially persistent data storage, such as a hard drive (HDD), a solid state drive (SDD), network attached storage (NAS), a tape drive, or any combination thereof. Illustratively, the data store 226 may correspond to the encoding rules data store 116, the DNS record data store 119, the distribution data store 118, or any combination thereof.

[0040] In addition to the user interface module 212, the memory 210 may include address encoding software 216 that may be executed by the processing unit 204. In one embodiment, the address encoding software 216 implements various aspects of the present disclosure, e.g., the generation DNS records including network address encoded with DNS-level information.

[0041] With reference to FIGS. 3A and 3B, a set of illustrative interactions will be described for the generation and use of network addresses encoded with DNS-level information

on a content delivery system 110. Specifically, FIG. 3A depicts illustrative interactions for the generation of DNS records mapping a distribution identifier to a network address encoded with DNS-level information, and for transmission of such a DNS record to an accessing computing device 102A. FIG. 3B depicts illustrative interactions for the handling of a communication request from the accessing computing device 102A to the network address encoded with DNS-level information. For the ease of illustration, the interactions of FIGS. 3A and 3B will be described with reference to an individual accessing computing device 102A requesting to access an individual distribution on the content delivery system 110, and with reference to an individual network address of that distribution. However, similar interactions may occur in some embodiments with respect to multiple accessing computing devices 102A (accessing one or more distributions), multiple distributions, multiple network addresses for a given distribution, or any combination thereof.

[0042] The interactions of FIG. 3A begin at (1), where the DNS service 112 loads encoding rules for encoding DNS-level information of a distribution into a network address. Illustratively, such rules may be loaded from the encoding rules data store 116. In some instances, the loaded rules may be generally static or constant across time or distributions. In other instances, the loaded rules may differ based on the specific distribution (e.g., as a high risk or non-high risk distribution), based on the time of day (e.g., to vary network address formats to prevent knowledge by unauthorized devices), or based on other criteria. Thereafter, at (2), the DNS service 112 encodes DNS-level data into a network address according to the obtained rules. Illustratively, the DNS service 112 may determine one or more fields for DNS-level information specified by the rules, generate data corresponding to each field according to an algorithm specified in the rule, and concatenate that data according to an ordering specified within the rule to result in a network address. Thereafter, at (3), the DNS service 112 can generate a DNS record mapping an identifier of the distribution, such as a domain name, to the network address encoded with DNS-level data, and store the DNS record within the DNS record data store 119.

[0043] Further detail regarding the generation of network addresses with encoded DNS-level data will be described with reference to FIG. 4, which is an illustrative visualization or graphical depiction of a potential format for network addresses with encoded DNS-level data. Specifically, FIG. 4 depicts an illustrative format for a 128 bit network address, which may conform, for example, to the IPv6 format. For ease of depiction, the 128 bit network address of

FIG. 4 is divided into 8 logical bit groups, each corresponding to 16 of the 128 bits of the network address. Each field of DNS-level data within an encoded network address may be represented by the values of bits within a given bit group. For example, as shown in FIG. 4, the first 48 bits of the network address may be used to represent a routing prefix. Each routing prefix may be associated, for example, with one or more POPs 120, such that network packets addressed to any network address with a given prefix will be routed via the network 106 to the corresponding POP 120 to which the prefix is assigned. Illustratively, the length and content of the routing prefix may be collectively agreed on by participants in the network 106, and thus the DNS service 112 may be configured to maintain such prefixes without encoding DNS-level data into the prefixes. As shown in FIG. 4, all or a portion of the remaining bits of a network address (e.g., other than the routing prefix) may be used to represent encoded DNS-level information. For example, bit group 8 (the final 16 bits of the network address) may be used to uniquely identify a distribution associated with the network address. Illustratively, where each distribution on the content delivery system 110 is associated with a unique identifier representable in 16 bits or less, a network address encoded according to the format of FIG. 4 may directly represent that unique identifier within bit group 8. Where each distribution on the content delivery system 110 is associated with a unique identifier that may or may not be representable in 16 bits or less (e.g., a domain name of arbitrary length), a network address encoded according to the format of FIG. 4 may represent that unique identifier as a hash value, generated by passing the unique identifier through a hashing function. The specific number of bits utilized to represent a distribution identifier may vary, for example, on the number of different distribution identifiers supported under a given format, and the probability of collisions between different identifiers under a given encoding scheme. On receiving a request to communicate with the network address, a receiving device (e.g., a content server 122) may identify the distribution to which the request is directed based on the final bit group of the network address. Additional DNS-level information that may be included within a network address according to the illustrative format of FIG. 4 includes a rule identifier represented by bit group 4, validity information represented by bit group 5, and hint information represented by bit group 6. Illustratively, the bits of group 4, representing a rule identifier, may identify a rule that describes the network address encoding format shown in FIG. 4 (e.g., identifier "0001"), and may be used to identify network addresses encoded according to the network address encoding

format shown in FIG. 4 from other potential formats. Thus, on receiving a request to communicate with the network address, a receiving device (e.g., a content server 122) may identify the format of the network address according to the bits of group 4 (e.g., enabling the receiving device to obtain rules specifying how other DNS-level data is encoded into the network address). The bits of group 5, representing validity information, may represent a digital signature generated based on a private key of the DNS service 112 along with a set of inputs, which may include, for example, the values of other bit groups of the network address. Thus, on receiving a request to communicate with the network address, a receiving device (e.g., a content server 122) may verify the validity of the network address by confirming the digital signature (e.g., decrypting the digital signature using a corresponding cryptographic public key and comparing a resulting value to an independent hash of the same set of inputs). The bits of groups 6 and 7, representing hint information, may represent information informing a receiving device of how to handle requests to communicate with the network address, such as an identifier of a security certificate to utilize in creating a secure communication channel with an accessing computing device 102. Thus, on receiving a request to communicate with the network address, a receiving device (e.g., a content server 122) may retrieve a security certificate corresponding to the hint information and use such a certificate to establish secure communications with the accessing computing device 102. Additional examples of hint information may include, for example, an identifier or network address of a resolver from which a DNS request corresponding to the encoded network address was received, a location of that resolver, etc.

[0044] While one potential format of a network address with encoded DNS-level information is shown in FIG. 4, other formats are possible and contemplated under the present disclosure. For example, more than 48 bits may be included within a routing prefix, or an additional bit group (e.g., bit group 4) may be used as a subnet prefix, in accordance with IPv6 protocols. Moreover, the specific bits used to identify each type of DNS-level data may vary according to different potential formats. While these bits are shown in FIG. 4 as 16-bit groups for ease of description, any number of bits may be used to identify each type of DNS-level data. For example, less than 16 or 32 bits may be sufficient to store a rule identifier or hint information, respectively (e.g., in the instance where less 65,536 total rule identifiers or 2^{32} hint information values are intended to be supported). More than 16 bits may be used to store validity information (e.g., according to a strength of security desired) or distribution identifiers

(e.g., according to the number of identifiers supported and a potential for collisions should a hash function be utilized). Any number of algorithms, including a variety of known hash algorithms, may be utilized to determine the bit values representative of given DNS-level data. Thus, the size, content, and arrangement of DNS-level data as encoded into a network address may vary.

[0045] As noted above, all or a portion of a network address may be encrypted. For example, the values of any one or more of bit groups 4 through 8 as shown in FIG. 4 may be encrypted according to public key encryption (using a public key) to represent an encrypted bit group, and a network address may be returned including at least one encrypted bit group. Thereafter, the values of the encrypted bit groups may be decrypted utilizing a corresponding private key, to determine the unencrypted values of the bit groups. Illustratively, a DNS server 112 may be configured to provide network addresses including at least one encrypted bit group within a DNS response, and one or more routers of the content delivery system 110 may be configured to decrypt the bit groups on receiving a data packet addressed to a network address including an encrypted bit group. In one embodiment, the router may validate or verify integrity of the network address (e.g., using validity information encoded into the network address, such as a digital signature), before forwarding the data packet within the content delivery system 110 (e.g., utilizing the unencrypted network address). In this manner, the network addresses utilized within the content delivery system 110 may be kept confidential to devices outside of the content delivery system 110 (or without a private key utilized by the content delivery system 110). While DNS-level information may be encrypted as described above, other data conveyed by a network address may additionally or alternatively be encrypted. For example, where a network assigns network addresses sequentially, randomly, or via another distribution mechanism, all or a portion of such network addresses may be encrypted before being provided in a DNS response, and a receiving device (e.g., a content server 112, a router, etc.) may be configured to decrypt the network address on receiving a data packet addressed to the encrypted network address

[0046] Returning to the interactions of FIG. 3A, once a DNS record for a distribution has been created, the DNS service 112 may function to obtain requests to resolve an identifier of the distribution from an accessing computing device 102A, as shown at (4). While this request is shown in FIG. 3A as beginning at the accessing computing device 102A and terminating at the DNS service 112, the request may be transmitted according to well-known DNS transmission protocols, and thus may pass through any number of intermediary devices (e.g., DNS resolvers)

prior to arriving at the DNS service 112. Thus, the DNS service 112 may not be directly aware of the identity of the accessing computing device 102A.

[0047] At (5), in response to the request, the DNS service 112 may return the DNS record referencing the network address encoded with DNS-level data. Similarly to interaction (4), while this response is shown as beginning at the DNS service 112 and terminating at the accessing computing device 102A, the response may be transmitted according to well-known DNS transmission protocols, and thus may pass through any number of intermediary devices (e.g., DNS resolvers) prior to arriving at the accessing computing device 102A. On receiving the DNS record, the accessing computing device 102A may then transmit network data to the network address referenced in the DNS record, as will be described with reference to FIG. 3B.

[0048] While a simplified set of interactions are described with reference to FIG. 3A, additional or alternative interactions may occur under embodiments of the present disclosure. For example, the DNS service 112 may in some instances maintain multiple DNS records for a distribution (e.g., each including a network address associated with a different POP 120), and distribute the DNS records according to a load balancing or traffic shaping algorithm (e.g., round robin). As a further example, the DNS service 112 may in some instances operate to generate or modify DNS records on-the-fly, in response to requests to resolve a distribution identifier. For example, the DNS service 112 may generate validity data for a network address based at least in part on attributes of the request, such as a geographic or logical region from which the request was received. As another example, the DNS service 112 may in some instances modify a TTL value for a DNS record to corresponding to a length of time until the DNS service 112 is configured to generate new validity data for network addresses of a distribution associated with the DNS record. In this way, where accessing computing device 102A comply with the TTL value of the DNS record, these devices can be expected to consider a DNS record expired at or before the content delivery system 110 would consider the network address in the record as invalid due to a change in validity information. Thus, different accessing computing devices 102 may obtain different DNS records from the DNS service 112, even when requesting to resolve the same distribution identifier, and the ordering and content of the interactions of FIG. 3A should be viewed as illustrative in nature.

[0049] With reference to FIGS. 3B, illustrative interactions will be described for the handling of a communication request from the accessing computing device 102A to the network address encoded with DNS-level information. For ease of reference, the interactions of FIG. 3B are numbered in sequence with those of FIG. 3A; however, the interactions of FIG. 3A and 3B may in some instances occur separately (e.g., in the case of a network attack, where multiple accessing computing devices 102A may transmit data to a network address obtained by DNS request of a different accessing computing device 102A).

[0050] The interactions of FIG. 1 begin at (6), where the accessing computing device 102A transmits a request to access content of a distribution to a network address including encoded DNS-level data (e.g., as referenced within a DNS record obtained from the DNS service 112). The request may be routed via the network to a POP 120 associated with a routing prefix within the network address, where the request may be processed according to a content server 122 (e.g., as selected according to load balancing of the POP 120). Specifically, at (7), the POP 120 may load one or more decoding rules for the network address from the decoding rules data store 124. In some instances, the loaded rules may be generally static, and the POP 120 may load a current set of decoding rules. In other instances, the POP 120 may load a set of rules corresponding to a version identifier represented by bits of the encoded network address (e.g., where the bits of the encoded network address representing the version identifier are identifiable without the rules themselves, such as by locating the bits in a common position across a set of potential rules).

[0051] At (8), the POP 120 can decode DNS-level data from the network address, according to the decoding rules. For example, the POP 120 may utilize the decoding rules to determine a distribution identifier, hint information, and validity information associated with the request. Illustratively, each type of DNS-level data may be obtained my inspecting a set of bits corresponding to the DNS-level data, as identified in the decoding rules. In some instances, a set of bits may be further processed according to algorithms of the decoding rules to transform the bits into a format useable by the POP 120 (e.g., by reversing an encoding of a particular type of DNS-level data).

[0052] At (9), the POP 120 can validate the request to access content at least partly based on validity information included within the network address to which the request is transmitted. For example, validity information may include both a time of generation of a

network address and a TTL value for the network address (or for a DNS record including the network address). Accordingly, a receiving device may decode the time of generation value and TTL value, and utilize these values to determine whether the network address should be considered valid (e.g., wherein if the time of generation value plus the TTL value does not meet or exceed a current time, the network address may be considered invalid. As another example, where the validity information is a digital signature generated based on a cryptographic hash of remaining bits of the network address, the POP 120 may generate a corresponding hash using those remaining bits of the network address and a decrypted value representing a decryption of the digital signature using a public key of the DNS service 112, and verify that the decrypted value and generated hash match. In some embodiments, the public/private key pair utilized to generate a digital signature may be specified within the network address (e.g., by a version identifier identifying a public/private key pair from a pre-established list). In other instances, the public/private key pair utilized to generate a digital signature may be separately agreed upon by the DNS service 112 and the POP 120. For example, the DNS server 112, the POP 120, or a third component (not shown in FIGS. 3A-3B) may periodically generate or select a new public/private key pair for use by the DNS service 112 and the POP 120, and the POP 120 may, after a threshold period of time subsequent to generation of a new public/private key pair (e.g., matching a TTL value of DNS records provided by the DNS service 112), consider network addresses with digital signatures generated under prior public/private key pairs to be invalid. This may enable the POP 120 to independently enforce TTL values of DNS records using digital signature verification.

[0053] In an instance where the decoding rules specify other inputs for validity information, such as a region from which the request originates, the POP 120 may verify the digital signature using such inputs. In the instance that the network address is considered valid, the POP 120 may continue to process the request. Otherwise, the POP 120 may consider the request invalid, and decline to further process the request (e.g., by "blackholing" the request). While validation of a request is described with reference to the POP 120 generally, such validation may occur at a variety of locations within the POP 120, such as at a content server 122 or at routers of the POP 120 (which routers are not shown in FIG. 3B) acting as intermediaries between a content server 112 and the accessing computing device 102A. In some instances, the public key corresponding to a digital signature may be made publically available by the content

delivery system 110. Thus, components external to the content delivery system 110 may be enabled to verify digital signatures included within network addresses. For example, an accessing computing device 102 may include software configured to validate digital signatures in network addresses returned as part of a response to a DNS request, which may prevent or deter falsification or modification of DNS responses (e.g., via a man-in-the-middle attack). As another example, routers or communication devices on the network 106 may be configured to verify digitally signed network addresses before forwarding such addresses, to prevent malicious traffic on the network 106.

[0054] At (10), in the instance that the network address is considered valid, the POP 120 initiates communication with the accessing computing device 102A and obtains content responsive to the request, based at least partly on the DNS-level information. Illustratively, the POP 120 may initiate secure communications with the accessing computing device 102A based on a security certificate identified by the hint information of the network address. The POP 120 may further access a requested portion of content corresponding to the distribution within the data cache 126 based on a distribution identifier of the network address. Thereafter, the POP 120 can return the requested content to the accessing computing device 102, at (11). Thus, the accessing computing device 102A can utilize a network address encoded with DNS-level information to access content on the content delivery system 110.

[0055] While the interactions of FIG. 3B generally describe interactions that may occur with respect to a legitimate accessing computing device 102A, POPs 120 may additionally be configured to process and handle malicious communications from illegitimate accessing computing devices 102. Illustratively, such malicious communications may include invalid validity information (e.g., by virtue of the fact that the DNS service 112 has altered the validity information since a targeted network address was obtained by the illegitimate accessing computing devices 102), and thus may be ignored by the POP 120. Furthermore, because network addresses may include a distribution identifier as encoded DNS-level data, the POPs 120 may be enabled to identify, from the network address (and potentially without any additional information), a distribution targeted by a network attack. The POPs 120 may then undertake actions to mitigate the network attack, such as requesting that the DNS service 112 modify DNS records for the targeted distribution to include new validity information. Further illustrations of actions that may be used to mitigate network attacks are described in more detail

within U.S. Patent Application No. 14/864,683, entitled "MITIGATING NETWORK ATTACKS," and filed 09/24/2015, the entirety of which is hereby incorporated by reference.

[0056] All of the methods and processes described above may be embodied in, and fully automated via, software code modules executed by one or more general purpose computers or processors. The code modules may be stored in any type of non-transitory computer-readable medium or other computer storage device. Some or all of the methods may alternatively be embodied in specialized computer hardware.

[0057] Conditional language such as, among others, "can," "could," "might" or "may," unless specifically stated otherwise, are otherwise understood within the context as used in general to present that certain embodiments include, while other embodiments do not include, certain features, elements and/or steps. Thus, such conditional language is not generally intended to imply that features, elements and/or steps are in any way required for one or more embodiments or that one or more embodiments necessarily include logic for deciding, with or without user input or prompting, whether these features, elements and/or steps are included or are to be performed in any particular embodiment.

[0058] Disjunctive language such as the phrase "at least one of X, Y or Z," unless specifically stated otherwise, is otherwise understood with the context as used in general to present that an item, term, etc., may be either X, Y or Z, or any combination thereof (e.g., X, Y and/or Z). Thus, such disjunctive language is not generally intended to, and should not, imply that certain embodiments require at least one of X, at least one of Y or at least one of Z to each be present.

[0059] Unless otherwise explicitly stated, articles such as 'a' or 'an' should generally be interpreted to include one or more described items. Accordingly, phrases such as "a device configured to" are intended to include one or more recited devices. Such one or more recited devices can also be collectively configured to carry out the stated recitations. For example, "a processor configured to carry out recitations A, B and C" can include a first processor configured to carry out recitation A working in conjunction with a second processor configured to carry out recitations B and C.

[0060] Any routine descriptions, elements or blocks in the flow diagrams described herein and/or depicted in the attached figures should be understood as potentially representing modules, segments, or portions of code which include one or more executable instructions for

implementing specific logical functions or elements in the routine. Alternate implementations are included within the scope of the embodiments described herein in which elements or functions may be deleted, or executed out of order from that shown or discussed, including substantially synchronously or in reverse order, depending on the functionality involved as would be understood by those skilled in the art.

[0061] The foregoing may be better understood in view of the following sets of clauses:

Clause 1. A system to provide digitally signed network addresses, the system comprising:

a domain name system (DNS) computing device configured with computer-executable instructions to:

obtain a request to resolve a domain name into a network address;

determine a routing prefix of the network address, wherein the routing prefix is associated with a network including a content server associated with the domain name;

determine an additional portion of the network address associated with the content server;

encrypt the additional portion of the network address with a cryptographic public key to result in an encrypted portion;

generate the network address, wherein the network address includes at least the routing prefix and the encrypted portion; and

return the network address in response to the request; and

a router computing device associated with the network and configured with computer-executable instructions to:

obtain a data packet addressed to the network address;

decrypt the encrypted portion of the network address using a cryptographic private key, corresponding to the cryptographic public key, to result in decrypted information representing the additional portion of the network address; and

route the data packet based at least in part on the decrypted information.

Clause 2. The system of Clause 1, wherein the router computing device is configured to route the data packet based at least in part on the decrypted information by routing the data packet to the content server.

Clause 3. The system of Clause 1, wherein the network address is formatted as an Internet Protocol version 6 (IPv6) address.

Clause 4. The system of Clause 1, wherein the network address is formatted to include the routing prefix as a first set of bits within the network address and the encrypted portion as a second set of bits within the network address.

Clause 5. The system of Clause 1, wherein the routing prefix represents routing information to route a data packet to the network including the content server from a publically addressable network.

Clause 6. A computer-implemented method comprising:

obtaining a DNS request to resolve a domain name into a network address;

determining a routing prefix of the network address, wherein the routing prefix is associated with a network including a computing device associated with the domain name;

determining an additional portion of the network address associated with the content server;

encrypting the additional portion of the network address with a cryptographic public key to result in an encrypted portion;

generating the network address, wherein the network address includes at least the routing prefix and the encrypted portion; and

returning the network address in response to the request.

Clause 7. The computer-implemented method of Clause 6, wherein the additional portion includes at least one of a value assigned to the content server, DNS-level information associated with the DNS request, or a digital signature associated with a DNS server.

Clause 8. The computer-implemented method of Clause 6, wherein the DNS-level information includes at least one of the domain name, security information associated with the domain name, timing information of the DNS request, or information specifying a source of the DNS request.

Clause 9. The computer-implemented method of Clause 8 further comprising generating the DNS-level information by encoding at least one of the domain name according to one or more encoding rules.

Clause 10. The computer-implemented method of Clause 6 further comprising:

obtaining a data packet addressed to the network address;

decrypting the encrypted portion of the network address using a cryptographic private key, corresponding to the cryptographic public key, to result in decrypted information representing the additional portion of the network address; and

routing the data packet based at least in part on the decrypted information.

Clause 11. The computer-implemented method of Clause 10, wherein routing the data packet based at least in part on the decrypted information comprises:

identifying within the decrypted information a time of generation of the network address and a time-to-live (TTL) value for the network address; and

routing the data packet based at least in part on whether a current time exceeds a combination of the time of generation and the TTL value.

Clause 12. The computer-implemented method of Clause 10, wherein routing the data packet based at least in part on the decrypted information comprises:

identifying a domain name encoded into the decrypted information according to one or more encoding rules; and

routing the data packet to the content server associated with the domain name.

Clause 13. The computer-implemented method of Clause 10, wherein routing the data packet based at least in part on the decrypted information comprises:

identifying a digital signature included within the decrypted information;

checking a validity of the digital signature utilizing a second cryptographic public key; and

routing the data packet based at least in part on the validity of the encrypted value by at least one of routing the data packet to a content server when

the encrypted value is valid or discarding the data packet when the encrypted value is invalid.

Clause 14. Non-transitory computer-readable media comprising computer-executable instructions that, when executed, cause a computing system to:

obtain information corresponding to a DNS request to resolve a domain name into a network address;

determine a routing prefix of the network address, wherein the routing prefix is associated with a network including a content server associated with the domain name;

determine an additional portion of the network address associated with the content server;

encrypt the additional portion of the network address with a cryptographic public key to result in an encrypted portion;

generate the network address, wherein the network address includes at least the routing prefix and the encrypted portion; and

return the network address in response to the request.

Clause 15. The non-transitory computer-readable media of Clause 14, wherein the computer-executable instructions further cause the computing system to:

hash the additional portion of the network address according to a cryptographic hash function to result in a hash value; and

encrypt the hash value according to a cryptographic private key of a DNS server to result in a digital signature;

wherein the network address further includes the digital signature.

Clause 16. The non-transitory computer-readable media of Clause 14 further comprising second computer executable instructions that, when executed by a router computing device, cause the router computing device to:

obtain a data packet addressed to the network address;

decrypt the encrypted portion of the network address using a cryptographic private key, corresponding to the cryptographic public key, to result in decrypted information representing the additional portion of the network address; and

route the data packet based at least in part on the decrypted information.

Clause 17. The computer-implemented method of Clause 16, wherein the second computer executable instructions cause the router computing device to route the data packet based at least in part on the decrypted information by at least:

identifying a digital signature included within the decrypted information;

checking a validity of the digital signature utilizing a second cryptographic public key; and

routing the data packet based at least in part on the validity of the encrypted value by at least one of routing the data packet to a content server when the encrypted value is valid or discarding the data packet when the encrypted value is invalid.

Clause 18. The non-transitory computer-readable media of Clause 14, wherein the computer-executable instructions cause the computing system to determine the additional portion of the network address by referencing information mapping the content server to the additional portion.

Clause 19. The non-transitory computer-readable media of Clause 14, wherein the computer-executable instructions cause the computing system to determine the additional portion of the network address by encoding DNS-level information associated with the DNS request according to one or more encoding rules.

Clause 20. The non-transitory computer-readable media of Clause 14, wherein the network address further comprises a version identifier indicating at least one of the cryptographic public key or a cryptographic private key corresponding to the cryptographic public key.

Clause 21. A content delivery system configured to mitigate network attacks on a domain name, wherein the domain name is associated with content provided by the content delivery system, the content delivery system comprising:

a domain name system (DNS) computing device comprising a processor configured with computer-executable instructions to:

obtain one or more encoding rules for encoding DNS-level information into network addresses provided by the DNS computing device;

obtain a request to resolve a domain name into a network address;

encode, according to one or more encoding rules, the domain name into the network address; and

return the network address in response to the request;

a computing device comprising a processor configured with computer-executable instructions to:

obtain one or more decoding rules for decoding DNS-level information encoded into network addresses by the DNS computing device;

detect a malicious data packet addressed to the network address, wherein the malicious data packet forms at least a part of a network attack on the content delivery system;

decode, according to the one or more decoding rules, the domain name from the network address to which the malicious data packet is addressed; and

identify the domain name decoded from the network address to which the malicious data packet is addressed as a target of the network attack.

Clause 22. The content delivery system of Clause 21, wherein the network address is formatted as an Internet Protocol version 6 (IPv6) address.

Clause 23. The content delivery system of Clause 21, wherein the one or more encoding rules specify information to be included within individual bits of the network addresses.

Clause 24. The content delivery system of Clause 21, wherein the network address comprises a first portion corresponding to routing information on a publically addressable network, and a second portion including encoded DNS-level information.

Clause 25. The content delivery system of Clause 21, wherein the computing device is at least one of a content server computing device or a networking routing computing device.

Clause 26. A computer-implemented method for providing DNS-level information within encoded network addresses, the computer-implemented method comprising:

obtaining one or more rules for encoding DNS-level information into the encoded network addresses and decoding DNS-level information from the encoded network addresses, wherein the rules specify a format of the DNS-level information when encoded in the encoded network addresses and individual bits of the encoded network addresses to utilize in representing the DNS-level information;

receiving a DNS request to resolve a domain name into a network address;

using the one or more rules to encode DNS-level information associated with the DNS request into the network address;

returning the network address in response to the DNS request;

receiving a network packet addressed to the network address;

using the one or more rules to decode the DNS-level information from the network address; and

routing the data packet based at least in part on the DNS-level information decoded from the network address.

Clause 27. The computer-implemented method of Clause 26, wherein the DNS-level information includes at least one of the domain name, security information associated with the domain name, timing information of the DNS request, information specifying a source of the DNS request, or validity information indicating a validity of the network address generated based at least in part on the DNS request.

Clause 28. The computer-implemented method of Clause 26, wherein the DNS-level information includes the domain name, the computer-implemented method further comprising:

identifying the network packet as malicious and forming at least part of a network attack; and

identifying the domain name decoded from the network address to which the malicious network packet is addressed as a target of the network attack.

Clause 29. The computer-implemented method of Clause 26, wherein the DNS-level information includes validity information, and wherein routing the data packet based at least in part on the DNS-level information decoded from the network address comprises:

detecting that the validity information indicates that network packet is addressed to an invalid network address; and

discarding the network packet.

Clause 30. The computer-implemented method of Clause 26, wherein the DNS-level information includes validity information, and wherein routing the data packet based at least in part on the DNS-level information decoded from the network address comprises:

detecting that the validity information indicates that network packet is addressed to a valid network address; and

routing the network packet to a content server associated with the domain name.

Clause 31. The computer-implemented method of Clause 26, wherein the validity information indicates a time of the DNS request, and wherein detecting that the validity information indicates that network packet is addressed to the valid network address includes detecting that a period of time between a current time and the time of the DNS request falls with a threshold time-to-live value.

Clause 32. The computer-implemented method of Clause 26, wherein the validity information includes a digital signature associated with a public key, and wherein detecting that the validity information indicates that network packet is addressed to the valid network address includes verifying the digital signature using the public key.

Clause 33. Non-transitory computer-readable media comprising computer-executable instructions for encoding DNS-level information within network addresses that, when executed, cause a computing system to:

obtain one or more rules for encoding the DNS-level information into the network addresses and decoding the DNS-level information from the network addresses;

obtain a request for an encoded network address, wherein the request includes DNS-level information associated with a DNS request to resolve a domain name into the encoded network address;

encode the DNS-level information into the encoded network address according to at least the one or more rules;

return the encoded network address in response to the request;

receive a network request associated with the encoded network address; and

decode the encoded network address according to at least the one or more rules to result in the DNS-level information; and

respond to the network request based at least in part on the DNS-level information obtained by decoding the encoded network address.

Clause 34. The non-transitory computer-readable media of Clause 33, wherein the network request is at least one of a request to decode the encoded network address or a request to obtain content associated with the encoded network address.

Clause 35. The non-transitory computer-readable media of Clause 33, wherein the computer-executable instructions cause the computing system to respond to the network request based at least in part on the DNS-level information by at least one of: returning the DNS-level information in response to the network request; returning content associated with the network request; discarding the network request; or forwarding the network request to a content server associated with the DNS-level information.

Clause 36. The non-transitory computer-readable media of Clause 33, wherein the DNS-level information includes a domain name and an identifier of a security certificate associated with the domain name, and wherein the computer-executable instructions cause the computing system to respond to the network request at least partly by initiating a secure connection utilizing the security certificate.

Clause 37. The non-transitory computer-readable media of Clause 33, wherein the one or more rules comprise multiple rule versions, and wherein the DNS-level

information includes a version identifier associated with a rule version of the multiple rule versions utilized in encoding the encoded network address.

Clause 38. The non-transitory computer-readable media of Clause 37, wherein the computer-executable instructions cause the computing system to determine a rule version of the multiple rule versions to utilize in encoding the encoded network address based at least in part on a current time and a security level associated with the domain name.

Clause 39. The non-transitory computer-readable media of Clause 33, wherein the DNS-level information includes a domain name, and wherein the computer-executable instructions cause the computing system encode the DNS-level information into the encoded network address at least in part by:

hashing the domain name according to a hash function to result in a hash value; and

including the hash value as a portion of the encoded network address.

Clause 40. The non-transitory computer-readable media of Clause 39, wherein the hash function is a cryptographic hash function.

Clause 41. A system to provide digitally signed network addresses, the system comprising:

a domain name system (DNS) computing device configured with computer-executable instructions to:

obtain a request to resolve a domain name into a network address;

determine a portion of the network address based at least in part on the domain name;

hash the portion of the network address according to a cryptographic hash function to result in a hash value;

encrypt the hash value with a cryptographic private key to result in a digital signature;

generate the network address, wherein the network address includes at least the portion of the network address and the digital signature; and

return the network address in response to the request; and

a router computing device configured with computer-executable instructions to:

obtain a data packet addressed to the network address;

hash the portion of the network address according to the cryptographic hash function to result in a second hash value;

decrypt the digital signature with a cryptographic public key corresponding to the cryptographic private key to result in a decryption output;

compare the decryption output and the second hash value to determine a validity of the digital signature; and

route the data packet based at least in part on the validity of the digital signature.

Clause 42. The system of Clause 41, wherein the router computing device is configured to route the data packet based at least in part on the validity of the digital signature by at least one of routing the data packet to a content server when the digital signature is valid or discarding the data packet when the digital signature is invalid.

Clause 43. The system of Clause 41, wherein the network address is formatted as an Internet Protocol version 6 (IPv6) address.

Clause 44. The system of Clause 41, wherein the network address is formatted to include the portion as a first set of bits within the network address and the encrypted value as a second set of bits within the network address.

Clause 45. A computer-implemented method comprising:

obtaining a DNS request to resolve a domain name into a network address;

determining a portion of the network address based at least in part on the DNS request;

hashing the portion of the network address according to a cryptographic hash function to result in a hash value;

encrypting the hash value with a cryptographic private key to result in an encrypted value;

generating the network address, wherein the network address includes at least the portion of the network address and the encrypted value; and

returning the network address in response to the request.

Clause 46. The computer-implemented method of Clause 45 further comprising:

obtaining a data packet addressed to the network address;

hashing the portion of the network address according to the cryptographic hash function to result in a second hash value;

decrypting the encrypted value with a cryptographic public key to result in a decryption output;

comparing the decryption output and the second hash value to determine a validity of the encrypted value; and

routing the data packet based at least in part on the validity of the encrypted value.

Clause 47. The computer-implemented method of Clause 46, the method is performed without inspection of the content of the data packet.

Clause 48. The computer-implemented method of Clause 46, wherein routing the data packet based at least in part on the validity of the encrypted value comprises at least one of routing the data packet to a content server when the encrypted value is valid or discarding the data packet when the encrypted value is invalid.

Clause 49. The computer-implemented method of Clause 46, wherein the private key is included in a set of private keys, wherein individual private keys of the set of private keys are associated with individual time periods, wherein the cryptographic private key is selected according to a time of the encrypting, and wherein the cryptographic public key is selected according to a time of the decrypting.

Clause 50. The computer-implemented method of Clause 45, wherein hashing the portion of the network address according to the cryptographic hash function comprises hashing a combination of the portion and a routing prefix according to the cryptographic hash function.

Clause 51. The computer-implemented method of Clause 45, wherein the network address includes a first set of bits representing a routing prefix, a second set of bits representing the portion of the network address, and a third set of bits representing the encrypted value.

Clause 52. The computer-implemented method of Clause 45, wherein the portion of the network address represents DNS-level information associated with the DNS request, the DNS-level information including at least one of the domain name, security information associated with the domain name, timing information of the DNS request, or information specifying a source of the DNS request.

Clause 53. The computer-implemented method of Clause 45, wherein the portion of the network address includes a cryptographic public key.

Clause 54. The computer-implemented method of Clause 45 further comprising providing a cryptographic public key to an originating device of the DNS request, wherein the originating device is configured to:

obtain the network address;

hashing the portion of the network address according to the cryptographic hash function to result in a second hash value;

decrypting the encrypted value with the cryptographic public key to result in a decryption output;

comparing the decryption output and the second hash value to determine a validity of the network address.

Clause 55. Non-transitory computer-readable media comprising computer-executable instructions that, when executed, cause a computing system to:

obtain a request for a network address, wherein the request includes DNS-level information associated with a DNS request to resolve a domain name into the encoded network address;

determine a portion of the network address based at least in part on the DNS request;

hash the portion of the network address according to a cryptographic hash function to result in a hash value;

encrypt the hash value with a private key to result in an encrypted value;

generate the network address, wherein the network address includes at least the portion of the network address and the encrypted value; and

return the network address in response to the request.

Clause 56. The non-transitory computer-readable media of Clause 55, wherein the request for the network address is at least one of the DNS request or a request generated in response to the DNS request.

Clause 57. The non-transitory computer-readable media of Clause 55, wherein the computer-executable instructions cause the computing system to determine the portion of the network address based at least in part on the DNS request by referencing information mapping the network address to the portion.

Clause 58. The non-transitory computer-readable media of Clause 55, wherein the computer-executable instructions cause the computing system to determine the portion of the network address based at least in part on the DNS request by encoding the DNS-level information according to one or more encoding rules.

Clause 59. The non-transitory computer-readable media of Clause 55, wherein the network address further comprises a version identifier indicating at least one of the cryptographic hash function or bits of the network address representing the encrypted value.

Clause 60. The non-transitory computer-readable media of Clause 55, wherein the computer-executable instructions further cause the computing system to:

obtain a data packet addressed to the network address;

hash the portion of the network address according to the cryptographic hash function to result in a second hash value;

decrypt the encrypted value with a cryptographic public key to result in a decryption output;

compare the decryption output and the second hash value to determine a validity of the encrypted value; and

route the data packet based at least in part on the validity of the encrypted value.

Clause 61. The non-transitory computer-readable media of Clause 60, wherein the computer-executable instructions further cause the computing system to route the data packet based at least in part on the validity of the encrypted value by at least one of routing the data packet to a content server when the encrypted value is valid or discarding the data packet when the encrypted value is invalid.

Clause 62. The non-transitory computer-readable media of Clause 55, wherein the computer-executable instructions further cause the computing system to encrypt the network address according to a cryptographic public key.

[0062] It should be emphasized that many variations and modifications may be made to the above-described embodiments, the elements of which are to be understood as being among other acceptable examples. All such modifications and variations are intended to be included herein within the scope of this disclosure and protected by the following claims.

CLAIMS

WHAT IS CLAIMED IS:

1. A content delivery system configured to mitigate network attacks on a domain name, wherein the domain name is associated with content provided by the content delivery system, the content delivery system comprising:

a domain name system (DNS) computing device comprising a processor configured with computer-executable instructions to:

obtain one or more encoding rules for encoding DNS-level information into network addresses provided by the DNS computing device;

obtain a request to resolve a domain name into a network address;

encode, according to one or more encoding rules, the domain name into the network address; and

return the network address in response to the request;

a computing device comprising a processor configured with computer-executable instructions to:

obtain one or more decoding rules for decoding DNS-level information encoded into network addresses by the DNS computing device;

detect a malicious data packet addressed to the network address, wherein the malicious data packet forms at least a part of a network attack on the content delivery system;

decode, according to the one or more decoding rules, the domain name from the network address to which the malicious data packet is addressed; and

identify the domain name decoded from the network address to which the malicious data packet is addressed as a target of the network attack.

2. The content delivery system of Claim 1, wherein the network address is formatted as an Internet Protocol version 6 (IPv6) address.

3. The content delivery system of Claim 1, wherein the one or more encoding rules specify information to be included within individual bits of the network addresses.

4. The content delivery system of Claim 1, wherein the network address comprises a first portion corresponding to routing information on a publically addressable network, and a second portion including encoded DNS-level information.

5. The content delivery system of Claim 1, wherein the computing device is at least one of a content server computing device or a networking routing computing device.

6. A computer-implemented method for providing DNS-level information within encoded network addresses, the computer-implemented method comprising:

obtaining one or more rules for encoding DNS-level information into the encoded network addresses and decoding DNS-level information from the encoded network addresses, wherein the rules specify a format of the DNS-level information when encoded in the encoded network addresses and individual bits of the encoded network addresses to utilize in representing the DNS-level information;

receiving a DNS request to resolve a domain name into a network address;

using the one or more rules to encode DNS-level information associated with the DNS request into the network address;

returning the network address in response to the DNS request;

receiving a network packet addressed to the network address;

using the one or more rules to decode the DNS-level information from the network address; and

routing the data packet based at least in part on the DNS-level information decoded from the network address.

7. The computer-implemented method of Claim 6, wherein the DNS-level information includes at least one of the domain name, security information associated with the domain name, timing information of the DNS request, information specifying a source of the DNS request, or validity information indicating a validity of the network address generated based at least in part on the DNS request.

8. The computer-implemented method of Claim 6, wherein the DNS-level information includes the domain name, the computer-implemented method further comprising:

identifying the network packet as malicious and forming at least part of a network attack; and

identifying the domain name decoded from the network address to which the malicious network packet is addressed as a target of the network attack.

9. The computer-implemented method of Claim 6, wherein the DNS-level information includes validity information, and wherein routing the data packet based at least in part on the DNS-level information decoded from the network address comprises:

detecting that the validity information indicates that network packet is addressed to an invalid network address; and

discarding the network packet.

10. The computer-implemented method of Claim 6, wherein the DNS-level information includes validity information, and wherein routing the data packet based at least in part on the DNS-level information decoded from the network address comprises:

detecting that the validity information indicates that network packet is addressed to a valid network address; and

routing the network packet to a content server associated with the domain name.

11. The computer-implemented method of Claim 6, wherein the validity information indicates a time of the DNS request, and wherein detecting that the validity information indicates that network packet is addressed to the valid network address includes detecting that a period of time between a current time and the time of the DNS request falls with a threshold time-to-live value.

12. The computer-implemented method of Claim 6, wherein the validity information includes a digital signature associated with a public key, and wherein detecting that the validity information indicates that network packet is addressed to the valid network address includes verifying the digital signature using the public key.

13. A system comprising:

one or more processors; and

one or more memories having stored thereon computer-executable instructions for encoding DNS-level information within network addresses that, when executed on the one or more processors, cause the one more processors to:

obtain one or more rules for encoding the DNS-level information into the network addresses and decoding the DNS-level information from the network addresses;

obtain a request for an encoded network address, wherein the request includes DNS-level information associated with a DNS request to resolve a domain name into the encoded network address;

encode the DNS-level information into the encoded network address according to at least the one or more rules;

return the encoded network address in response to the request;

receive a network request associated with the encoded network address; and

decode the encoded network address according to at least the one or more rules to result in the DNS-level information; and

respond to the network request based at least in part on the DNS-level information obtained by decoding the encoded network address.

14. The system of Claim 13, wherein the network request is at least one of a request to decode the encoded network address or a request to obtain content associated with the encoded network address.

15. The system of Claim 13, wherein the computer-executable instructions cause the computing system to respond to the network request based at least in part on the DNS-level information by at least one of: returning the DNS-level information in response to the network request; returning content associated with the network request; discarding the network request; or forwarding the network request to a content server associated with the DNS-level information.

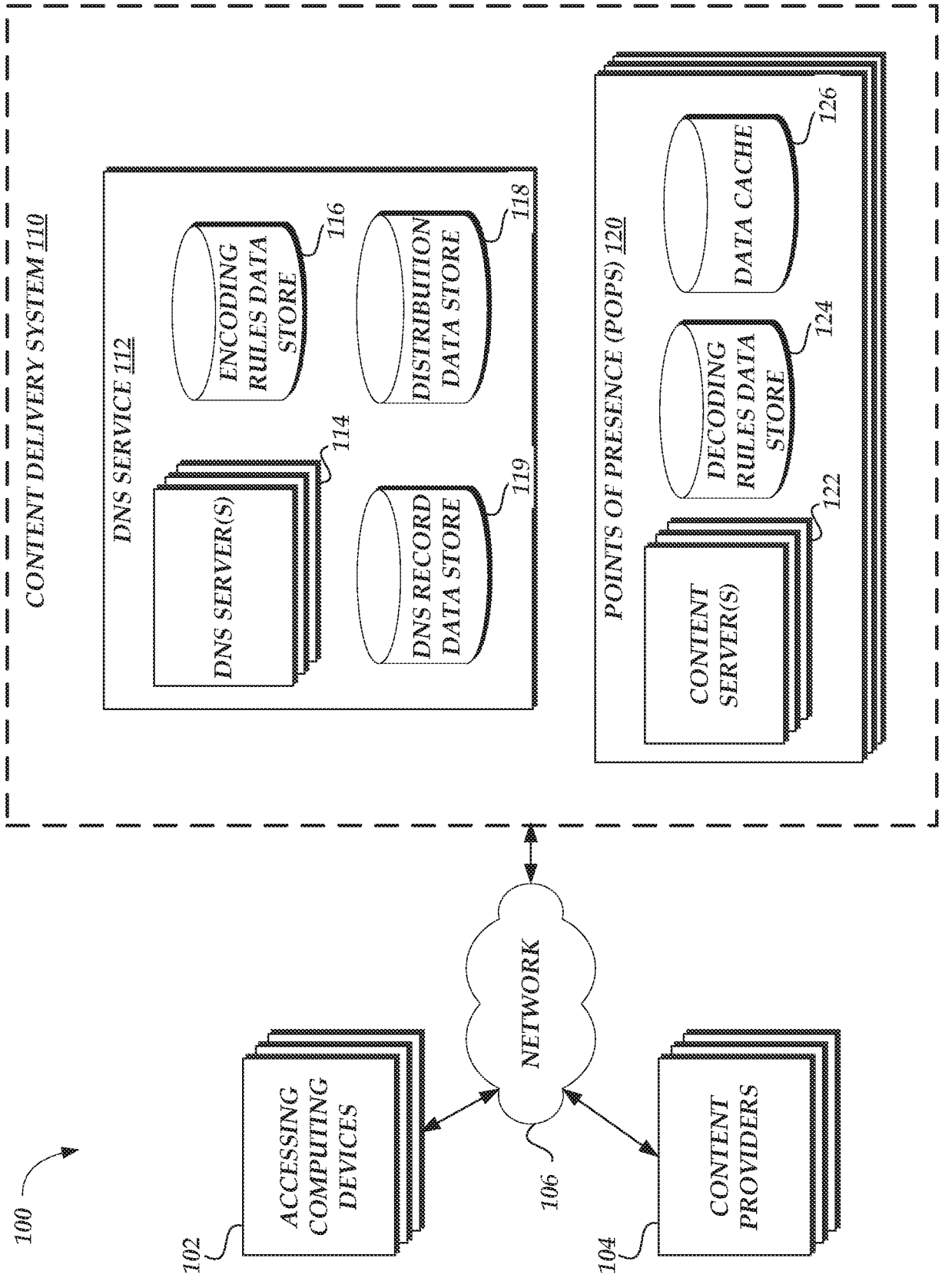
100
CONTENT DELIVERY SYSTEM 110
DNS SERVICE 112
DNS SERVER(S)
114
ENCODING RULES DATA STORE
116
DISTRIBUTION DATA STORE
118
DNS RECORD DATA STORE
119
POINTS OF PRESENCE (POPS) 120
CONTENT SERVER(S)
122
DECODING RULES DATA STORE
124
DATA CACHE
126
NETWORK
106
ACCESSING COMPUTING DEVICES
102
CONTENT PROVIDERS
104

Fig. 1

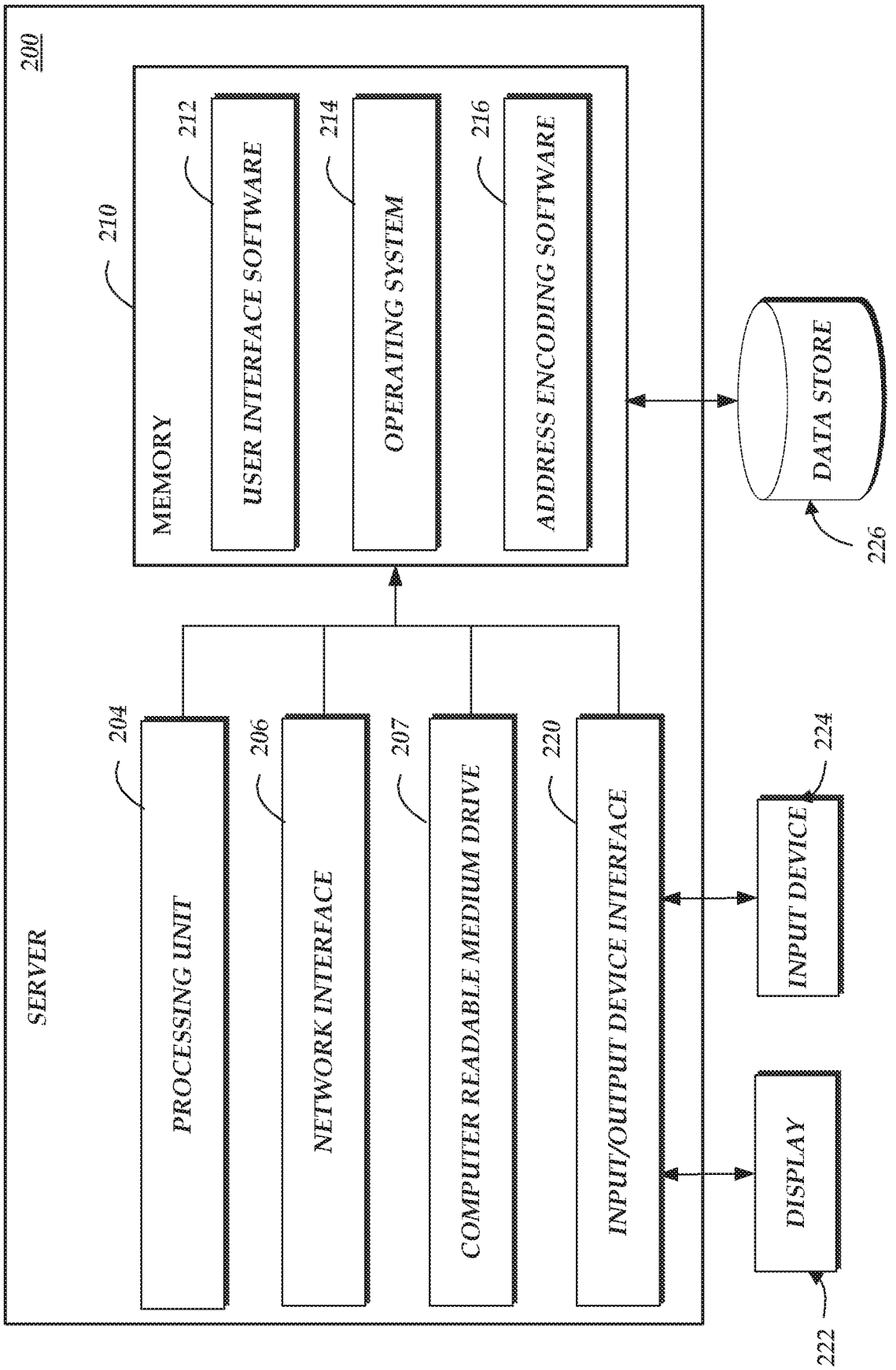
200
SERVER
MEMORY
210
212
USER INTERFACE SOFTWARE
214
OPERATING SYSTEM
216
ADDRESS ENCODING SOFTWARE
DATA STORE
226
204
PROCESSING UNIT
206
NETWORK INTERFACE
207
COMPUTER READABLE MEDIUM DRIVE
220
INPUT/OUTPUT DEVICE INTERFACE
INPUT DEVICE
224
DISPLAY
222

Fig. 2

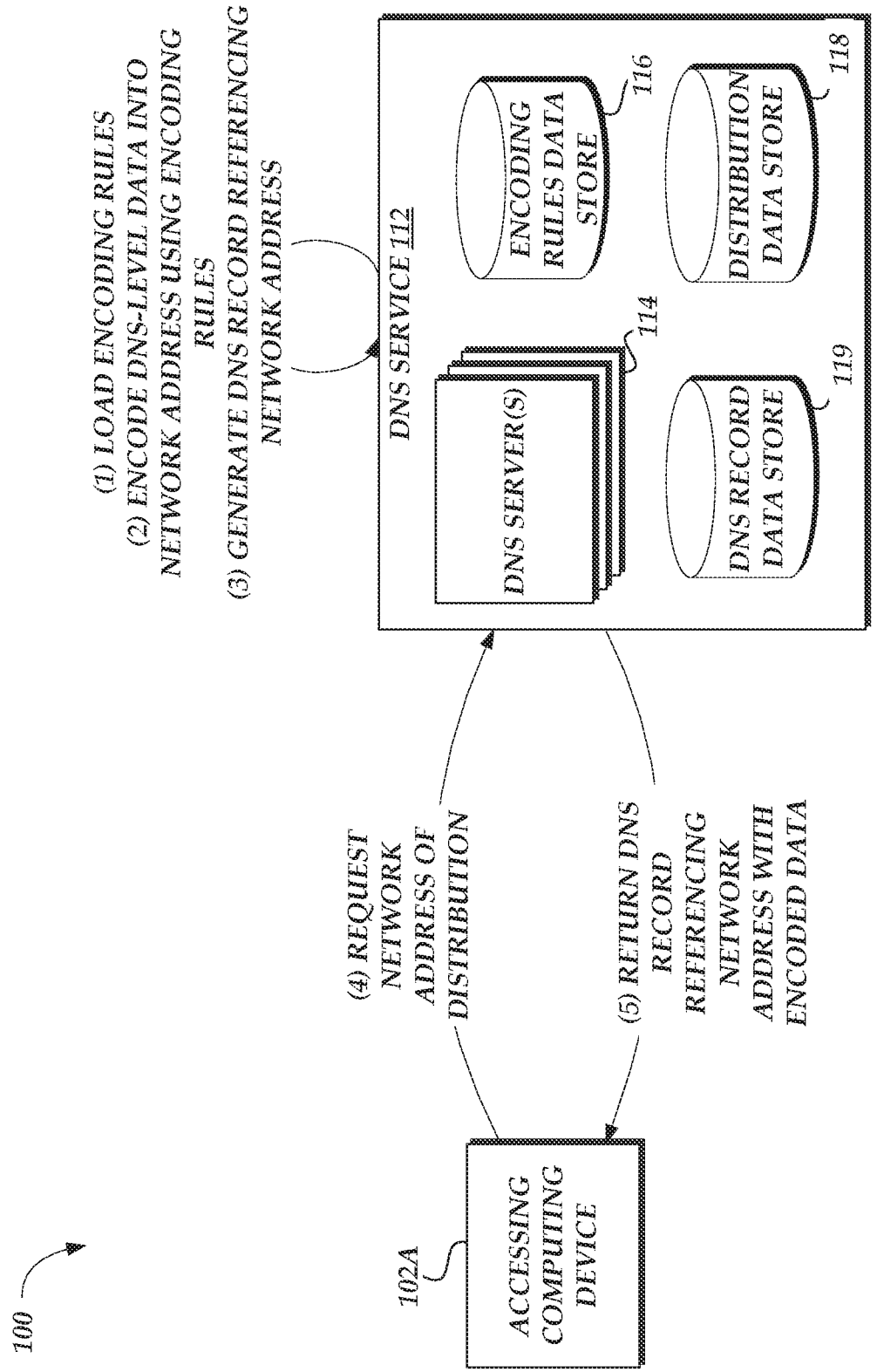
100
ACCESSING COMPUTING DEVICE
102A
(1) LOAD ENCODING RULES
(2) ENCODE DNS-LEVEL DATA INTO NETWORK ADDRESS USING ENCODING RULES
(3) GENERATE DNS RECORD REFERENCING NETWORK ADDRESS
DNS SERVICE 112
DNS SERVER(S)
114
ENCODING RULES DATA STORE
116
DNS RECORD DATA STORE
119
DISTRIBUTION DATA STORE
118
(4) REQUEST NETWORK ADDRESS OF DISTRIBUTION
(5) RETURN DNS RECORD REFERENCING NETWORK ADDRESS WITH ENCODED DATA

Fig. 3A

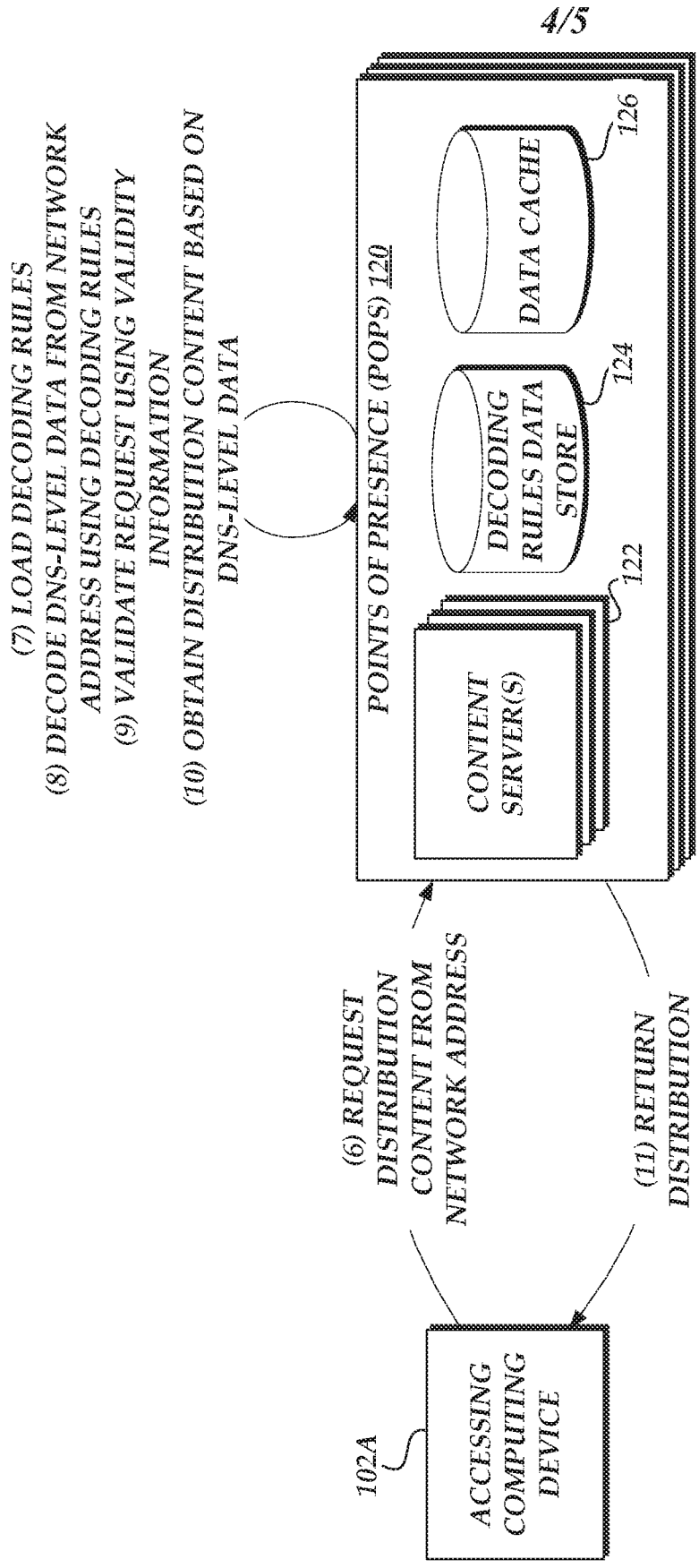
(7) LOAD DECODING RULES
(8) DECODE DNS-LEVEL DATA FROM NETWORK ADDRESS USING DECODING RULES
(9) VALIDATE REQUEST USING VALIDITY INFORMATION
(10) OBTAIN DISTRIBUTION CONTENT BASED ON DNS-LEVEL DATA
POINTS OF PRESENCE (POPS) 120
CONTENT SERVER(S)
122
DECODING RULES DATA STORE
124
DATA CACHE
126
(6) REQUEST DISTRIBUTION CONTENT FROM NETWORK ADDRESS
(11) RETURN DISTRIBUTION CONTENT
102A
ACCESSING COMPUTING DEVICE

Fig. 3B

EXAMPLE NETWORK ADDRESS ENCODING FORMAT (128 BITS)
XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX
GROUP 1
GROUP 2
GROUP 3
GROUP 4
GROUP 5
GROUP 6
GROUP 7
GROUP 8
(48 BITS)
(80 BITS)
ROUTING PREFIX
DNS-LEVEL ENCODED DATA
XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX
GROUP 4
GROUP 5
GROUP 6
GROUP 7
GROUP 8
(16 BITS)
(16 BITS)
(32 BITS)
(16 BITS)
RULE IDENTIFIER
VALIDITY INFORMATION
HINT INFORMATION
DISTRIBUTION IDENTIFIER

Fig. 4

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2017/055156

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/12
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P	US 9 584 328 B1 (GRAHAM-CUMMING JOHN [GB]) 28 February 2017 (2017-02-28) abstract column 6, line 7 - column 8, line 3	1-15
X,P	US 2017/155678 A1 (ARAÚJO JOÃO DIOGO TAVEIRA [PT] ET AL) 1 June 2017 (2017-06-01) abstract paragraphs [0027], [0031], [0052] - [0055], [0065], [0072], [0073]	1-15
X,P	US 2017/153980 A1 (ARAÚJO JOÃO DIOGO TAVEIRA [PT] ET AL) 1 June 2017 (2017-06-01) paragraphs [0030], [0032], [0053] - [0055], [0059], [0060]	1-15

-/--

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
6 December 2017	13/12/2017

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Goller, Wolfgang

1

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2017/055156

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2002/083198 A1 (KIM YONG-WOON [KR] ET AL) 27 June 2002 (2002-06-27)	6,7,9-15
A	abstract paragraphs [0004], [0005], [0008], [0010], [0023], [0033] - [0035], [0040], [0048]; claims 1-6	1-5,8
X	US 2009/043900 A1 (BARBER TIMOTHY P [US]) 12 February 2009 (2009-02-12) paragraphs [0012], [0040], [0041], [0046], [0049], [0053], [0056] - [0058], [0060], [0071] - [0073]	6,13-15

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2017/055156

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 9584328 B1	28-02-2017	US 9584328 B1	28-02-2017
		US 2017171232 A1	15-06-2017
US 2017155678 A1	01-06-2017	NONE	
US 2017153980 A1	01-06-2017	NONE	
US 2002083198 A1	27-06-2002	EP 1350365 A1	08-10-2003
		KR 20020054191 A	06-07-2002
		US 2002083198 A1	27-06-2002
		WO 02052794 A1	04-07-2002
US 2009043900 A1	12-02-2009	US 2009043900 A1	12-02-2009
		US 2011202665 A1	18-08-2011
		US 2012151024 A1	14-06-2012
		US 2012151072 A1	14-06-2012
		US 2013151726 A1	13-06-2013
		US 2014250241 A1	04-09-2014
		US 2017295133 A1	12-10-2017