US 20170249468A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2017/0249468 A1**

Wood et al. (43) **Pub. Date: Aug. 31, 2017**

(54) **METHOD AND SYSTEM FOR NAME ENCRYPTION AGREEMENT IN A CONTENT CENTRIC NETWORK**

(71) Applicant: **CISCO TECHNOLOGY, INC.**, San Jose, CA (US)

(72) Inventors: **Christopher A. Wood**, San Francisco, CA (US); **Glenn C. Scott**, Portola Valley, CA (US)

(73) Assignee: **CISCO TECHNOLOGY, INC.**, San Jose, CA (US)

(21) Appl. No.: **15/056,904**

(22) Filed: **Feb. 29, 2016**

Publication Classification

(51) **Int. Cl.**

G06F 21/60 (2006.01)

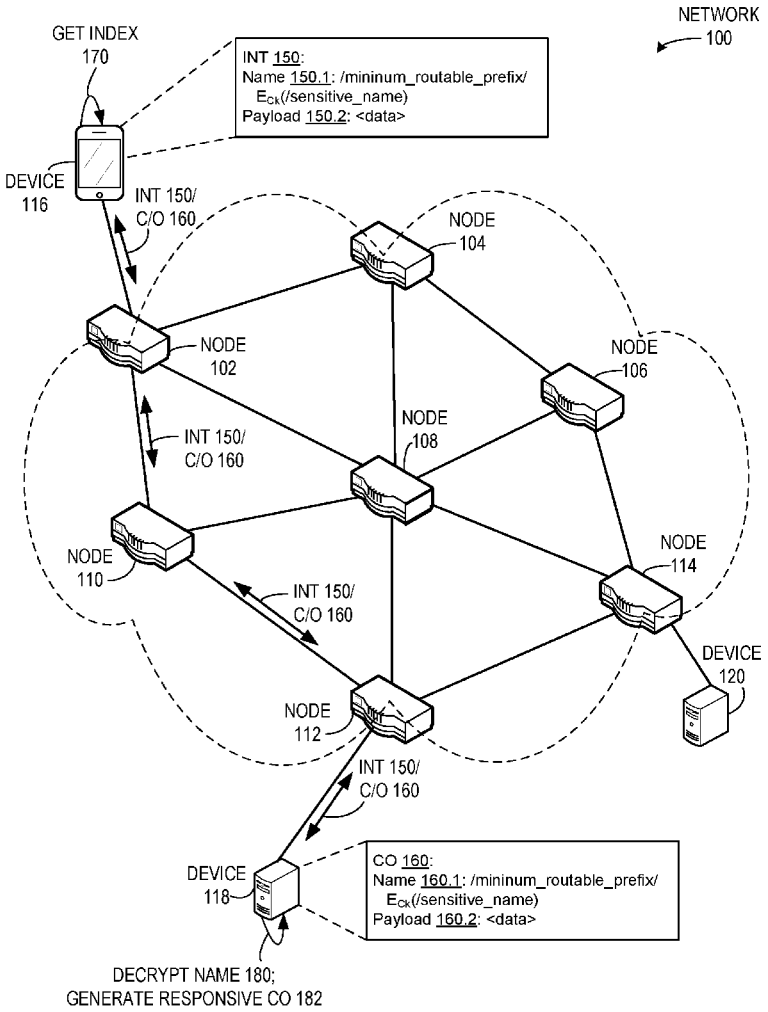
G06F 21/64 (2006.01)

(52) **U.S. Cl.**

CPC ***G06F 21/602*** (2013.01); ***G06F 21/645*** (2013.01)

(57) **ABSTRACT**

One embodiment provides a system that facilitates efficient name encryption in a CCN. During operation, the system determines, by a client computing device, an index for a name of an interest, wherein the name is a hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level, wherein the index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest. The system encrypts one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix. The system transmits the interest based on the encrypted name, thereby facilitating efficient name encryption in a CCN.



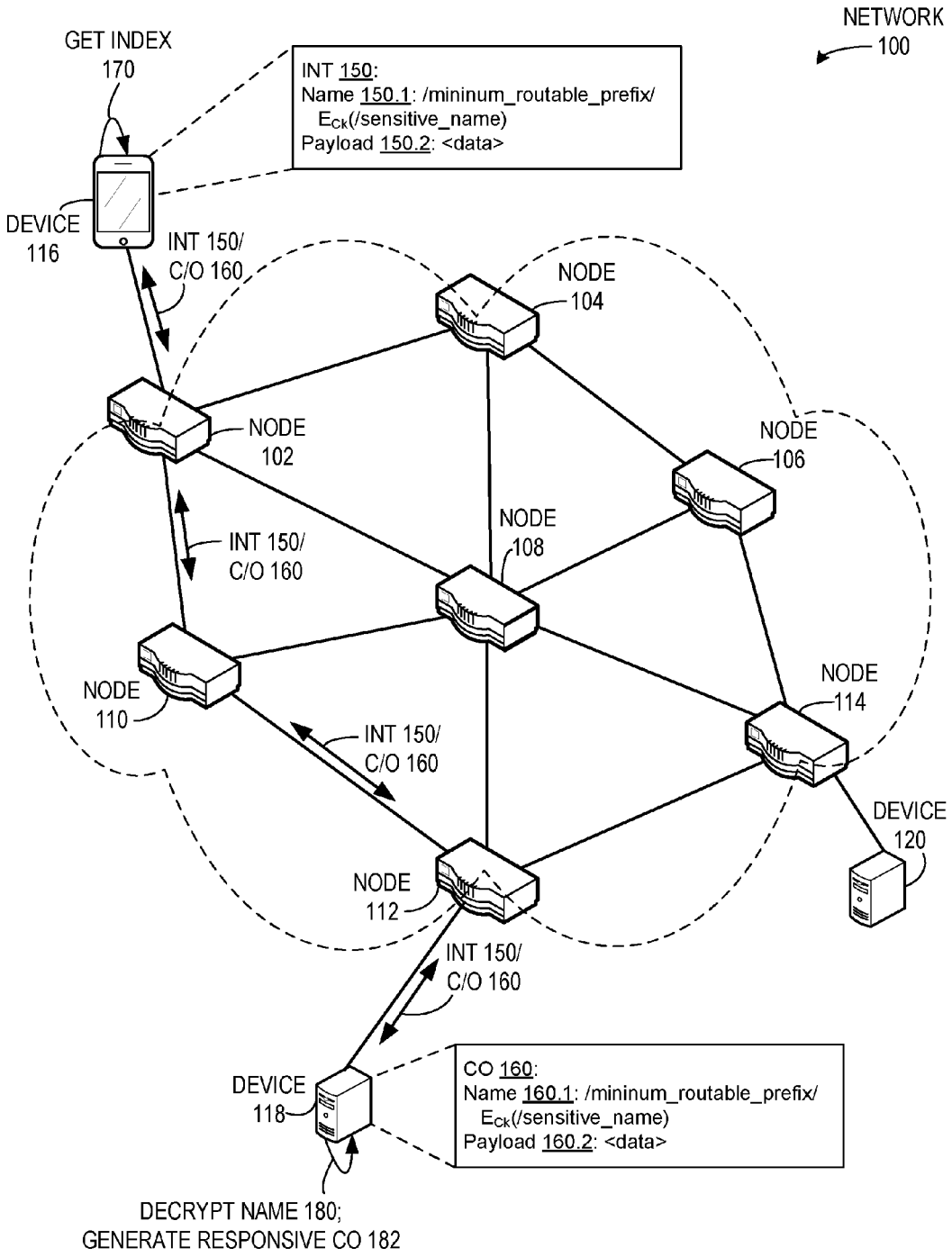

NETWORK
100
GET INDEX
170
INT 150:
Name 150.1: /mininum_routable_prefix/
E_{Ck}(/sensitive_name)
Payload 150.2: <data>
DEVICE
116
INT 150/
C/O 160
NODE
104
NODE
102
NODE
106
NODE
108
INT 150/
C/O 160
NODE
110
NODE
114
INT 150/
C/O 160
DEVICE
120
NODE
112
INT 150/
C/O 160
DEVICE
118
CO 160:
Name 160.1: /mininum_routable_prefix/
E_{Ck}(/sensitive_name)
Payload 160.2: <data>
DECRYPT NAME 180;
GENERATE RESPONSIVE CO 182

FIG. 1

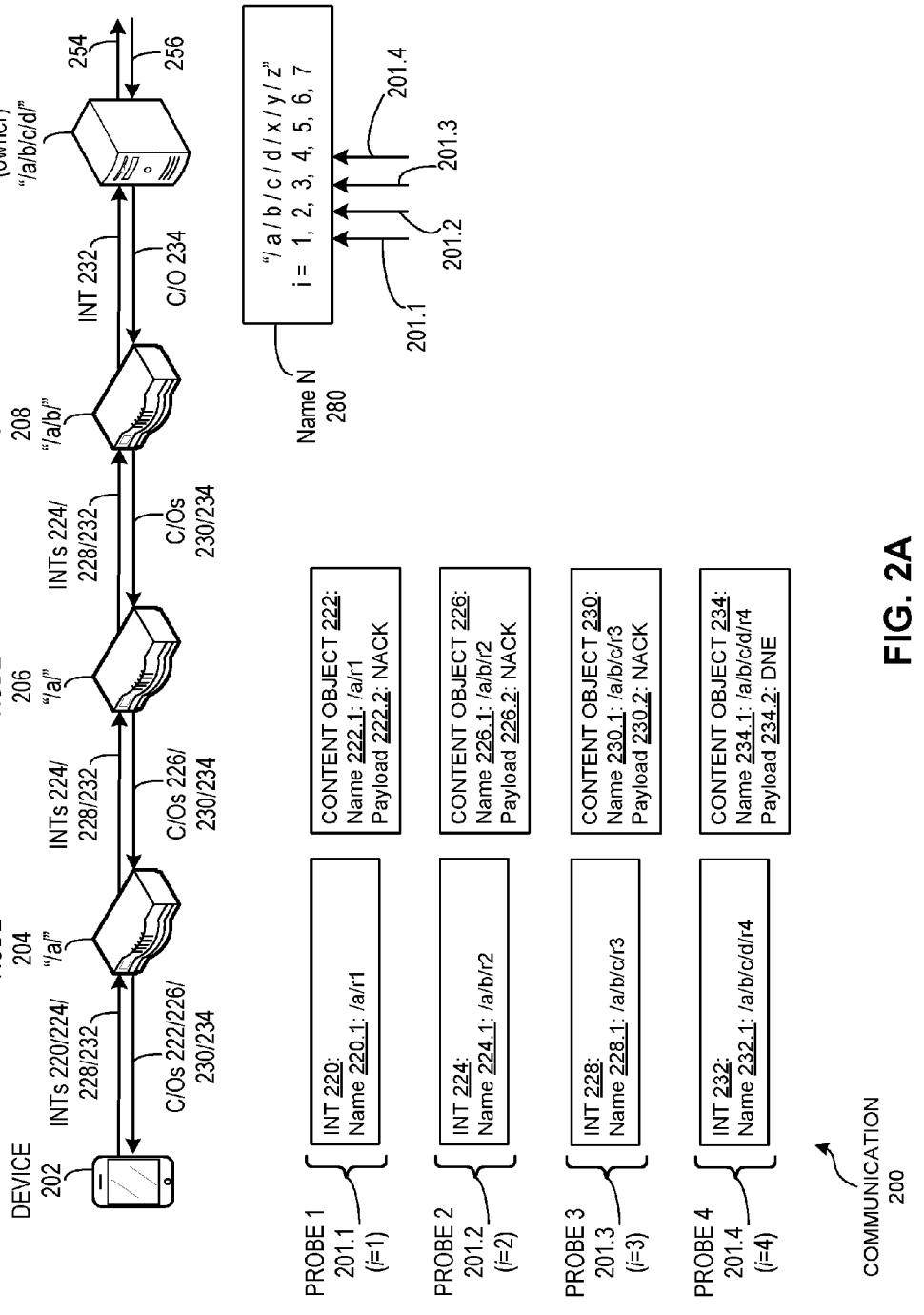
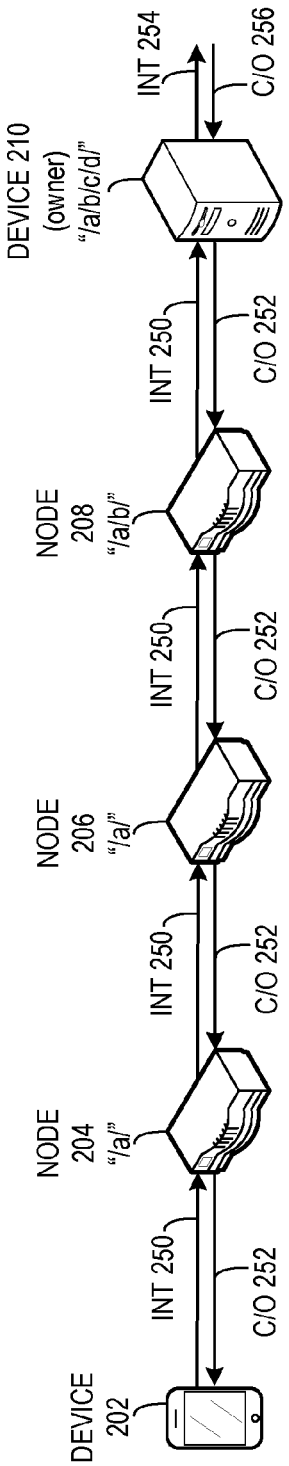

DEVICE 202
INTs 220/224/ 228/232
C/Os 222/226/ 230/234
NODE 204 "/a/"
INTs 224/ 228/232
C/Os 226/ 230/234
NODE 206 "/a/"
INTs 224/ 228/232
C/Os 230/234
NODE 208 "/a/b/"
INT 232
C/O 234
DEVICE 210 (owner) "/a/b/c/d/"
254
256
Name N 280
"/ a / b / c / d / x / y / z" i = 1, 2, 3, 4, 5, 6, 7
201.1
201.2
201.3
201.4
PROBE 1 201.1 (i=1)
INT 220: Name 220.1: /a/r1
CONTENT OBJECT 222: Name 222.1: /a/r1 Payload 222.2: NACK
PROBE 2 201.2 (i=2)
INT 224: Name 224.1: /a/b/r2
CONTENT OBJECT 226: Name 226.1: /a/b/r2 Payload 226.2: NACK
PROBE 3 201.3 (i=3)
INT 228: Name 228.1: /a/b/c/r3
CONTENT OBJECT 230: Name 230.1: /a/b/c/r3 Payload 230.2: NACK
PROBE 4 201.4 (i=4)
INT 232: Name 232.1: /a/b/c/d/r4
CONTENT OBJECT 234: Name 234.1: /a/b/c/d/r4 Payload 234.2: DNE
COMMUNICATION 200

FIG. 2A



ENCRYPTED INTEREST 250/ CONTENT OBJECT 252

INT 250:
Name 250.1: /a/b/c/d/Enc_202$_K$(/x/y/z)

CONTENT OBJECT 252:
Name 252.1: /a/b/c/d/Enc_202$_K$(/x/y/z)
Payload 252.2: <data>

DECRYPTED INTEREST 254/ CONTENT OBJECT 256

INT 254:
Name 254.1: /a/b/c/d/x/y/z

CONTENT OBJECT 256:
Name 256.1: /a/b/c/d/x/y/z
Payload 256.2: <data>

COMMUNICATION 240

FIG. 2B

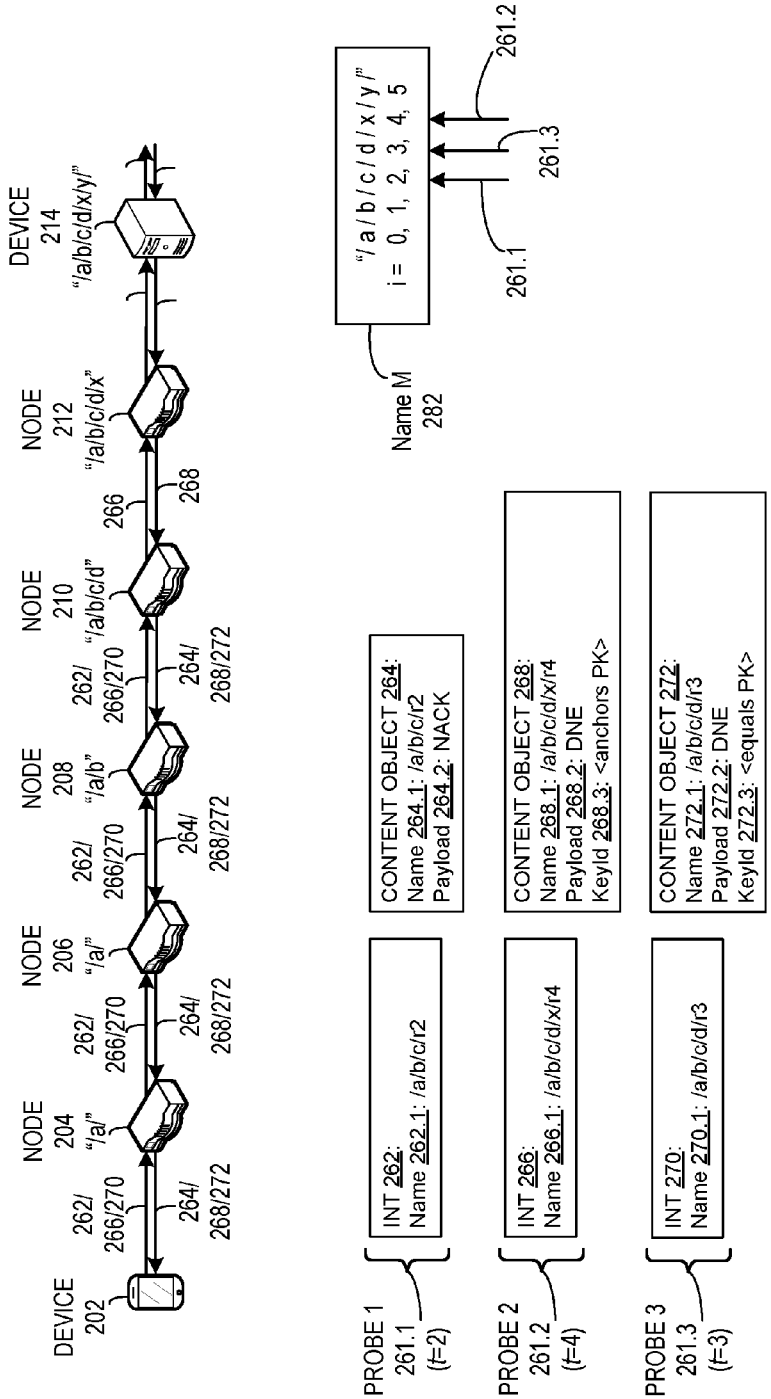

DEVICE 202
NODE 204 "/a/"
NODE 206 "/a/"
NODE 208 "/a/b"
NODE 210 "/a/b/c/d"
NODE 212 "/a/b/c/d/x"
DEVICE 214 "/a/b/c/d/x/y/"
262/266/270
264/268/272
266
268
Name M 282
"/a/b/c/d/x/y/" i = 0, 1, 2, 3, 4, 5
261.1
261.2
261.3
PROBE 1 261.1 (t=2)
INT 262: Name 262.1: /a/b/c/r2
CONTENT OBJECT 264: Name 264.1: /a/b/c/r2 Payload 264.2: NACK
PROBE 2 261.2 (t=4)
INT 266: Name 266.1: /a/b/c/d/x/r4
CONTENT OBJECT 268: Name 268.1: /a/b/c/d/x/r4 Payload 268.2: DNE Keyld 268.3: <anchors PK>
PROBE 3 261.3 (t=3)
INT 270: Name 270.1: /a/b/c/d/r3
CONTENT OBJECT 272: Name 272.1: /a/b/c/d/r3 Payload 272.2: DNE Keyld 272.3: <equals PK>
COMMUNICATION 260

FIG. 2C

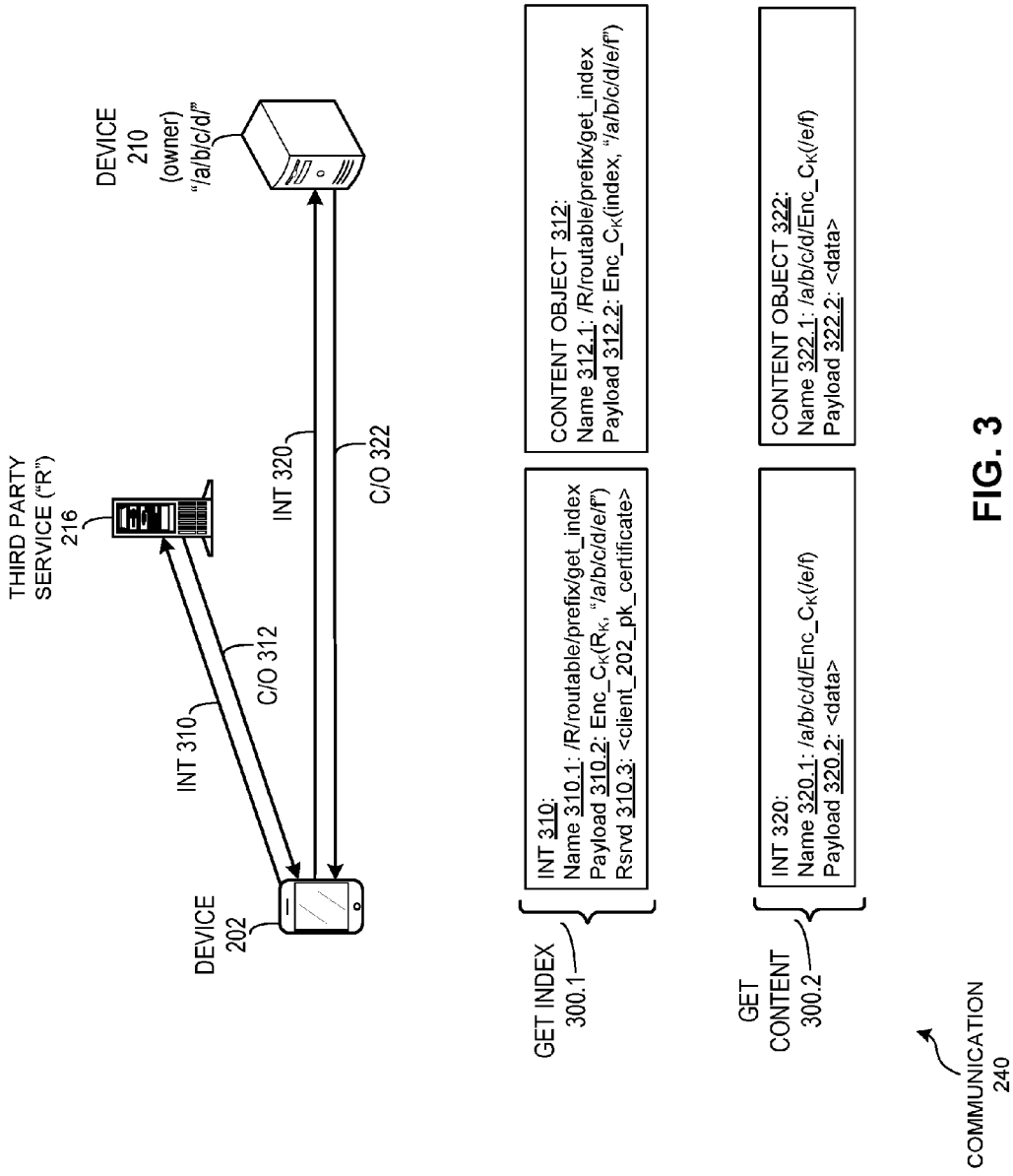

DEVICE
210
(owner)
"/a/b/c/d/"
THIRD PARTY
SERVICE ("R")
216
INT 320
C/O 322
INT 310
C/O 312
DEVICE
202
INT 310:
Name 310.1: /R/routable/prefix/get_index
Payload 310.2: Enc_CK(RK, "/a/b/c/d/e/f")
Rsrvd 310.3: <client_202_pk_certificate>
CONTENT OBJECT 312:
Name 312.1: /R/routable/prefix/get_index
Payload 312.2: Enc_CK(index, "/a/b/c/d/e/f")
GET INDEX
300.1
INT 320:
Name 320.1: /a/b/c/d/Enc_CK(/e/f)
Payload 320.2: <data>
CONTENT OBJECT 322:
Name 322.1: /a/b/c/d/Enc_CK(/e/f)
Payload 322.2: <data>
GET
CONTENT
300.2
COMMUNICATION
240

FIG. 3

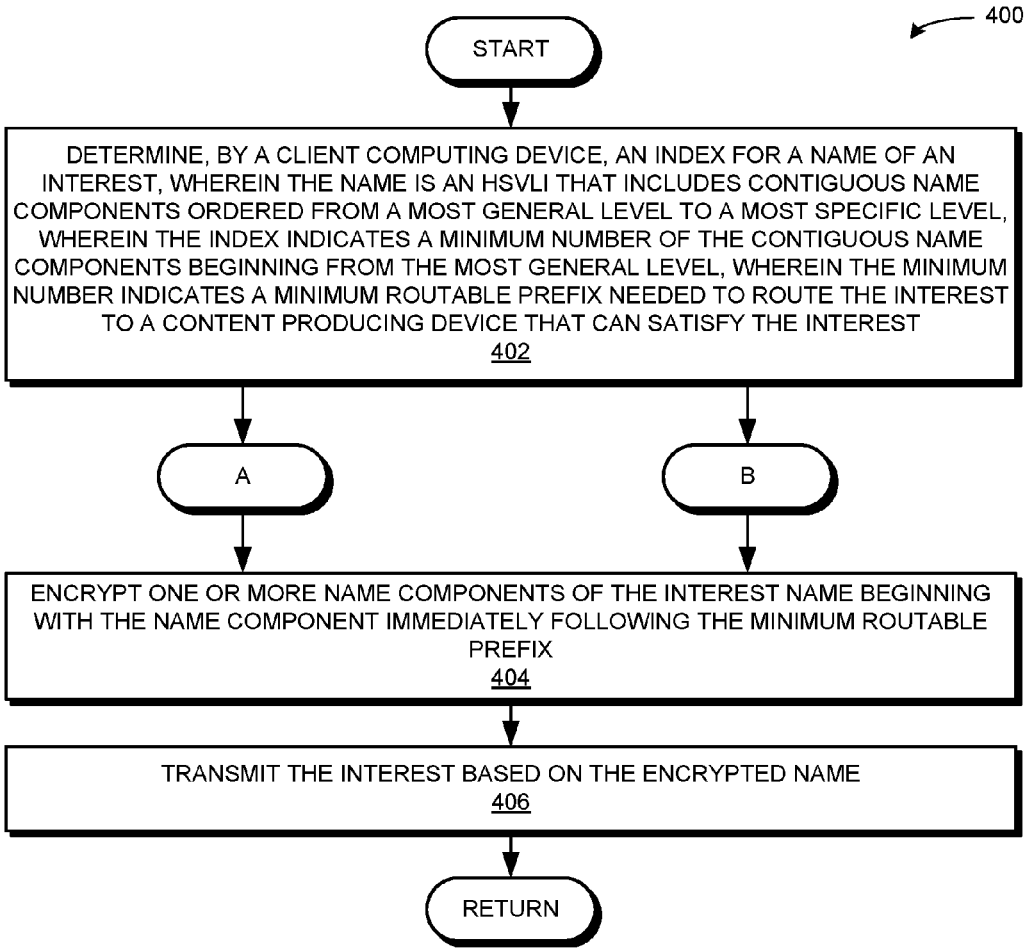

400
START
DETERMINE, BY A CLIENT COMPUTING DEVICE, AN INDEX FOR A NAME OF AN INTEREST, WHEREIN THE NAME IS AN HSVLI THAT INCLUDES CONTIGUOUS NAME COMPONENTS ORDERED FROM A MOST GENERAL LEVEL TO A MOST SPECIFIC LEVEL, WHEREIN THE INDEX INDICATES A MINIMUM NUMBER OF THE CONTIGUOUS NAME COMPONENTS BEGINNING FROM THE MOST GENERAL LEVEL, WHEREIN THE MINIMUM NUMBER INDICATES A MINIMUM ROUTABLE PREFIX NEEDED TO ROUTE THE INTEREST TO A CONTENT PRODUCING DEVICE THAT CAN SATISFY THE INTEREST
402
A
B
ENCRYPT ONE OR MORE NAME COMPONENTS OF THE INTEREST NAME BEGINNING WITH THE NAME COMPONENT IMMEDIATELY FOLLOWING THE MINIMUM ROUTABLE PREFIX
404
TRANSMIT THE INTEREST BASED ON THE ENCRYPTED NAME
406
RETURN

FIG. 4A

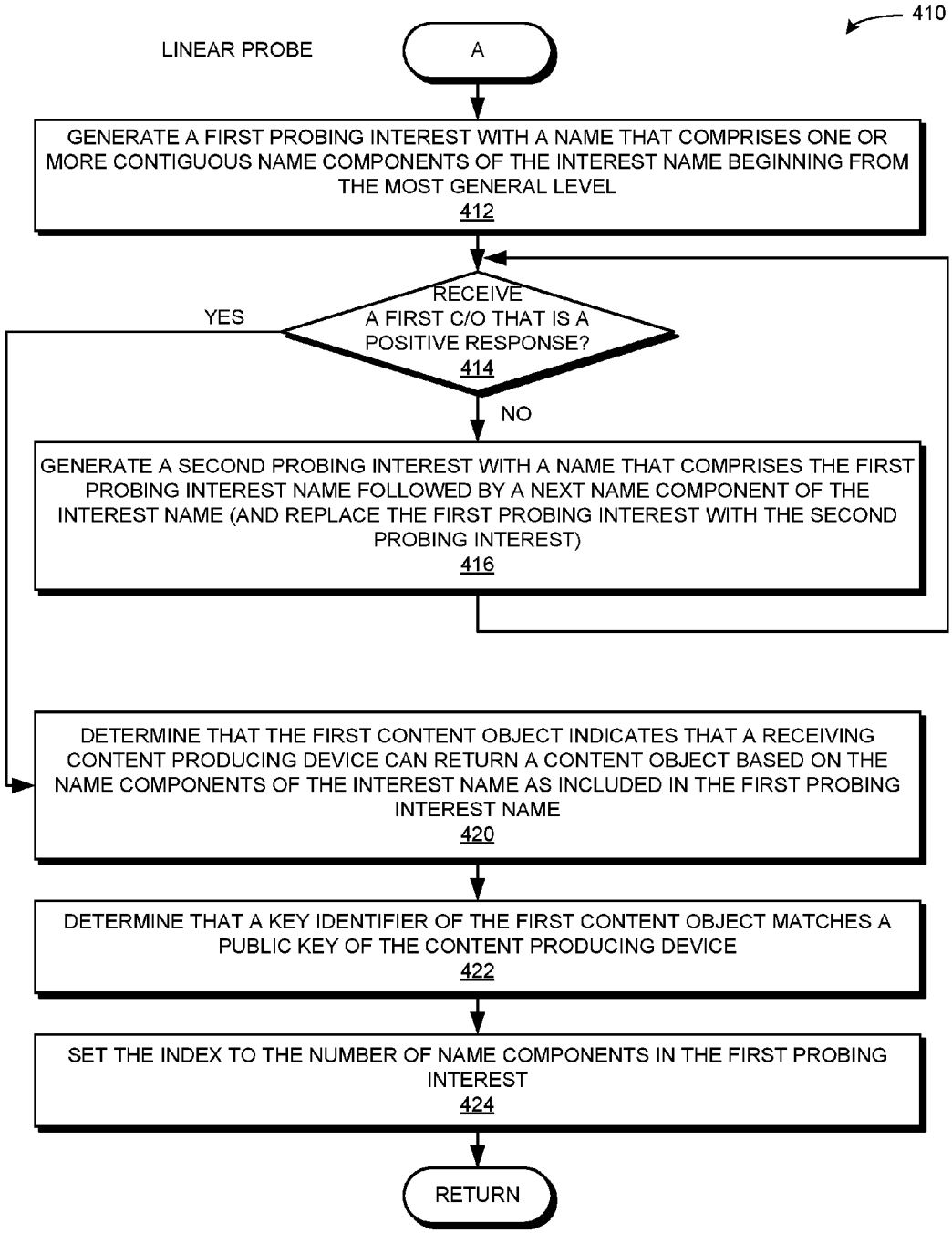

410
LINEAR PROBE
A
GENERATE A FIRST PROBING INTEREST WITH A NAME THAT COMPRISES ONE OR MORE CONTIGUOUS NAME COMPONENTS OF THE INTEREST NAME BEGINNING FROM THE MOST GENERAL LEVEL
412
RECEIVE A FIRST C/O THAT IS A POSITIVE RESPONSE?
414
YES
NO
GENERATE A SECOND PROBING INTEREST WITH A NAME THAT COMPRISES THE FIRST PROBING INTEREST NAME FOLLOWED BY A NEXT NAME COMPONENT OF THE INTEREST NAME (AND REPLACE THE FIRST PROBING INTEREST WITH THE SECOND PROBING INTEREST)
416
DETERMINE THAT THE FIRST CONTENT OBJECT INDICATES THAT A RECEIVING CONTENT PRODUCING DEVICE CAN RETURN A CONTENT OBJECT BASED ON THE NAME COMPONENTS OF THE INTEREST NAME AS INCLUDED IN THE FIRST PROBING INTEREST NAME
420
DETERMINE THAT A KEY IDENTIFIER OF THE FIRST CONTENT OBJECT MATCHES A PUBLIC KEY OF THE CONTENT PRODUCING DEVICE
422
SET THE INDEX TO THE NUMBER OF NAME COMPONENTS IN THE FIRST PROBING INTEREST
424
RETURN

FIG. 4B

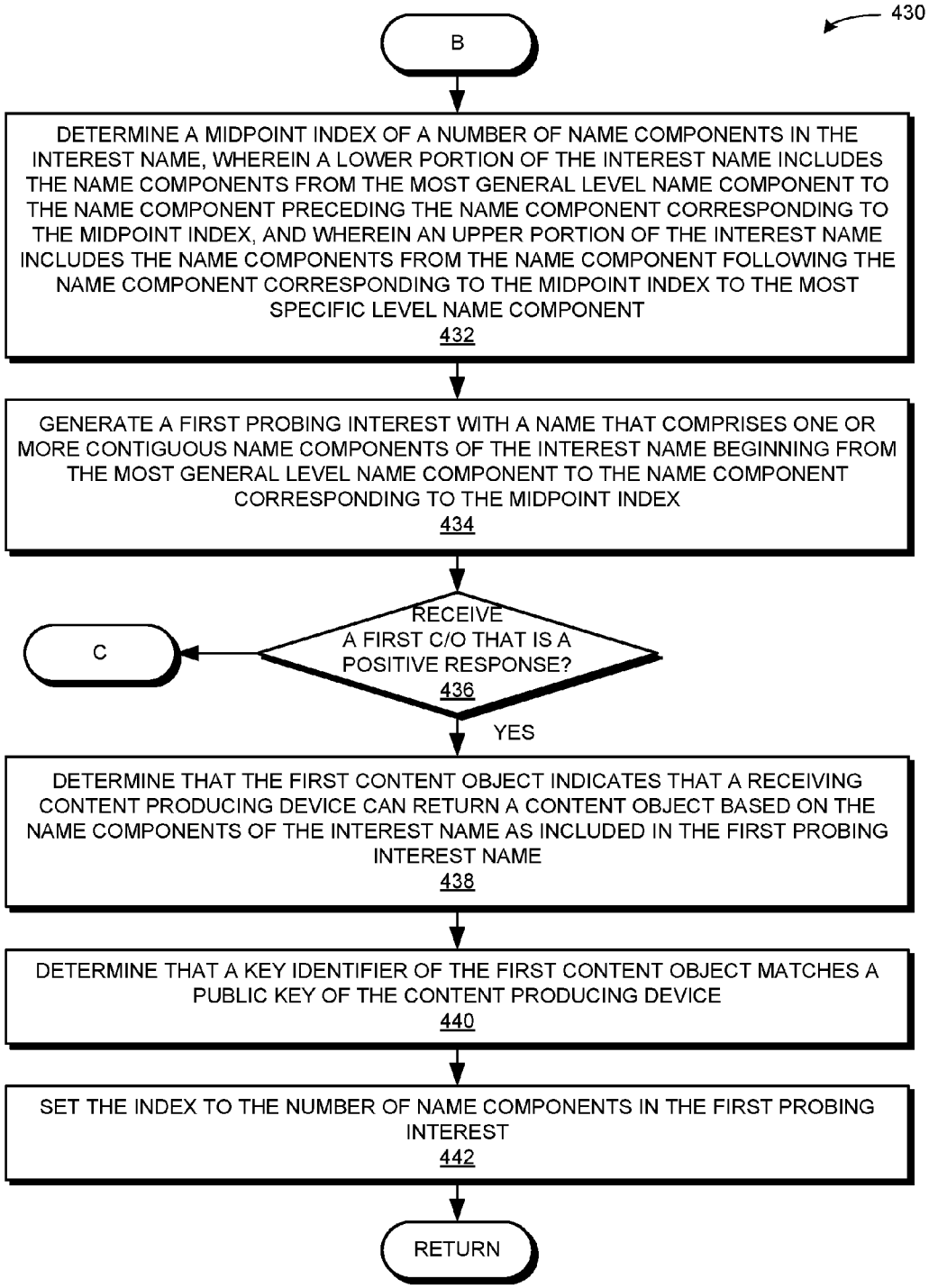

430
B
DETERMINE A MIDPOINT INDEX OF A NUMBER OF NAME COMPONENTS IN THE INTEREST NAME, WHEREIN A LOWER PORTION OF THE INTEREST NAME INCLUDES THE NAME COMPONENTS FROM THE MOST GENERAL LEVEL NAME COMPONENT TO THE NAME COMPONENT PRECEDING THE NAME COMPONENT CORRESPONDING TO THE MIDPOINT INDEX, AND WHEREIN AN UPPER PORTION OF THE INTEREST NAME INCLUDES THE NAME COMPONENTS FROM THE NAME COMPONENT FOLLOWING THE NAME COMPONENT CORRESPONDING TO THE MIDPOINT INDEX TO THE MOST SPECIFIC LEVEL NAME COMPONENT
432
GENERATE A FIRST PROBING INTEREST WITH A NAME THAT COMPRISES ONE OR MORE CONTIGUOUS NAME COMPONENTS OF THE INTEREST NAME BEGINNING FROM THE MOST GENERAL LEVEL NAME COMPONENT TO THE NAME COMPONENT CORRESPONDING TO THE MIDPOINT INDEX
434
RECEIVE A FIRST C/O THAT IS A POSITIVE RESPONSE?
436
C
YES
DETERMINE THAT THE FIRST CONTENT OBJECT INDICATES THAT A RECEIVING CONTENT PRODUCING DEVICE CAN RETURN A CONTENT OBJECT BASED ON THE NAME COMPONENTS OF THE INTEREST NAME AS INCLUDED IN THE FIRST PROBING INTEREST NAME
438
DETERMINE THAT A KEY IDENTIFIER OF THE FIRST CONTENT OBJECT MATCHES A PUBLIC KEY OF THE CONTENT PRODUCING DEVICE
440
SET THE INDEX TO THE NUMBER OF NAME COMPONENTS IN THE FIRST PROBING INTEREST
442
RETURN

FIG. 4C

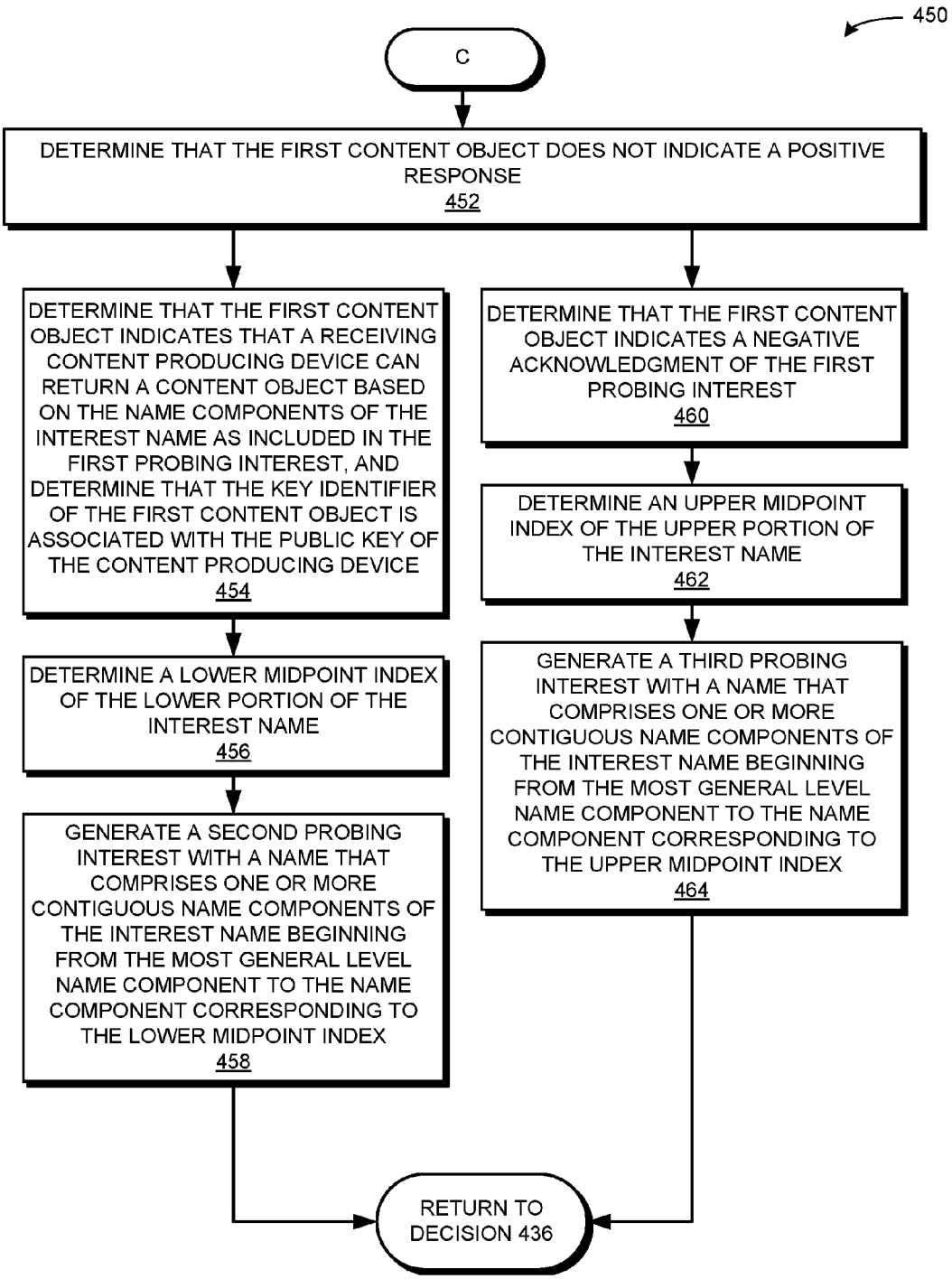

450
C
DETERMINE THAT THE FIRST CONTENT OBJECT DOES NOT INDICATE A POSITIVE RESPONSE
452
DETERMINE THAT THE FIRST CONTENT OBJECT INDICATES THAT A RECEIVING CONTENT PRODUCING DEVICE CAN RETURN A CONTENT OBJECT BASED ON THE NAME COMPONENTS OF THE INTEREST NAME AS INCLUDED IN THE FIRST PROBING INTEREST, AND DETERMINE THAT THE KEY IDENTIFIER OF THE FIRST CONTENT OBJECT IS ASSOCIATED WITH THE PUBLIC KEY OF THE CONTENT PRODUCING DEVICE
454
DETERMINE A LOWER MIDPOINT INDEX OF THE LOWER PORTION OF THE INTEREST NAME
456
GENERATE A SECOND PROBING INTEREST WITH A NAME THAT COMPRISES ONE OR MORE CONTIGUOUS NAME COMPONENTS OF THE INTEREST NAME BEGINNING FROM THE MOST GENERAL LEVEL NAME COMPONENT TO THE NAME COMPONENT CORRESPONDING TO THE LOWER MIDPOINT INDEX
458
DETERMINE THAT THE FIRST CONTENT OBJECT INDICATES A NEGATIVE ACKNOWLEDGMENT OF THE FIRST PROBING INTEREST
460
DETERMINE AN UPPER MIDPOINT INDEX OF THE UPPER PORTION OF THE INTEREST NAME
462
GENERATE A THIRD PROBING INTEREST WITH A NAME THAT COMPRISES ONE OR MORE CONTIGUOUS NAME COMPONENTS OF THE INTEREST NAME BEGINNING FROM THE MOST GENERAL LEVEL NAME COMPONENT TO THE NAME COMPONENT CORRESPONDING TO THE UPPER MIDPOINT INDEX
464
RETURN TO DECISION 436

FIG. 4D

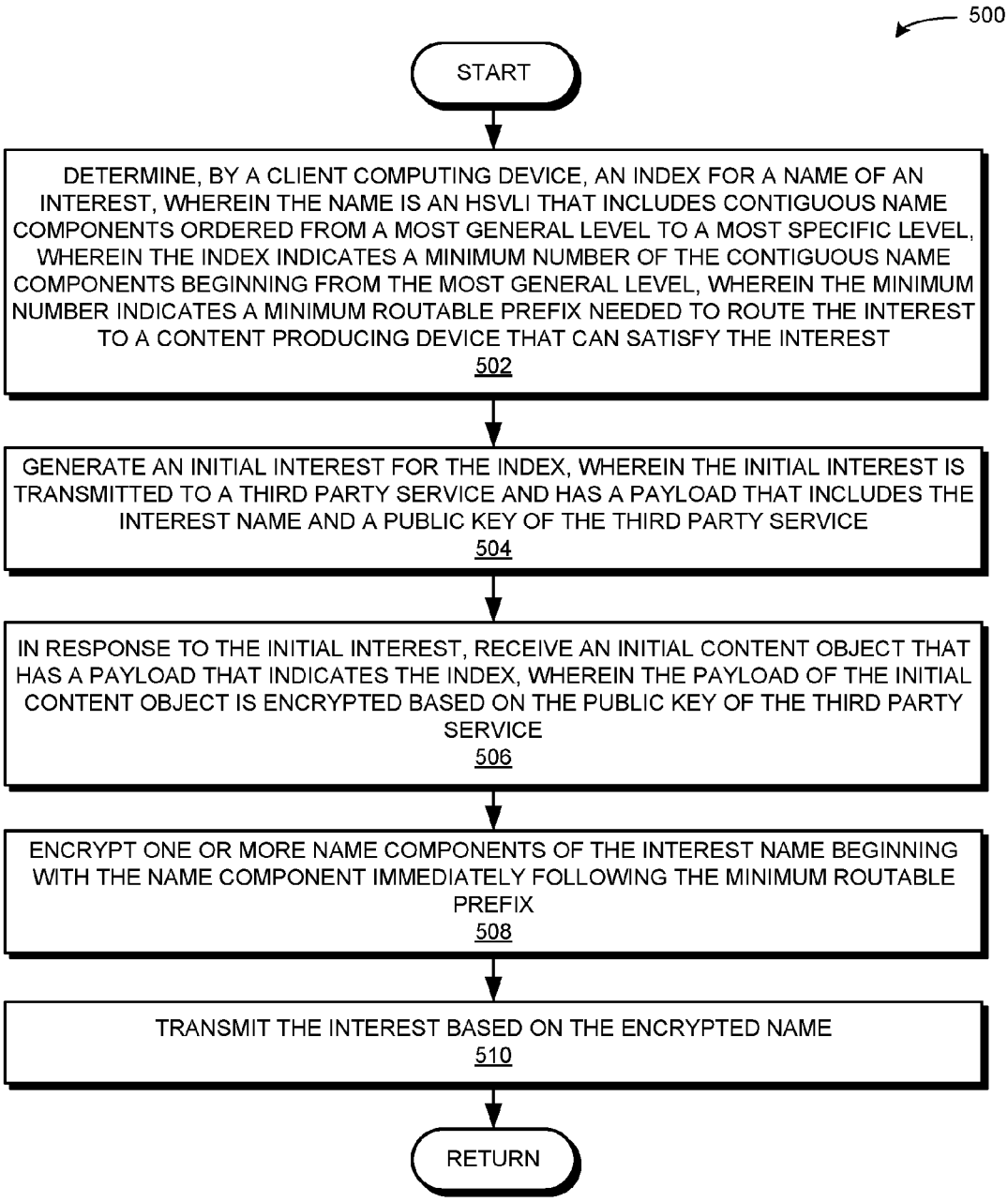

500
START
DETERMINE, BY A CLIENT COMPUTING DEVICE, AN INDEX FOR A NAME OF AN INTEREST, WHEREIN THE NAME IS AN HSVLI THAT INCLUDES CONTIGUOUS NAME COMPONENTS ORDERED FROM A MOST GENERAL LEVEL TO A MOST SPECIFIC LEVEL, WHEREIN THE INDEX INDICATES A MINIMUM NUMBER OF THE CONTIGUOUS NAME COMPONENTS BEGINNING FROM THE MOST GENERAL LEVEL, WHEREIN THE MINIMUM NUMBER INDICATES A MINIMUM ROUTABLE PREFIX NEEDED TO ROUTE THE INTEREST TO A CONTENT PRODUCING DEVICE THAT CAN SATISFY THE INTEREST
502
GENERATE AN INITIAL INTEREST FOR THE INDEX, WHEREIN THE INITIAL INTEREST IS TRANSMITTED TO A THIRD PARTY SERVICE AND HAS A PAYLOAD THAT INCLUDES THE INTEREST NAME AND A PUBLIC KEY OF THE THIRD PARTY SERVICE
504
IN RESPONSE TO THE INITIAL INTEREST, RECEIVE AN INITIAL CONTENT OBJECT THAT HAS A PAYLOAD THAT INDICATES THE INDEX, WHEREIN THE PAYLOAD OF THE INITIAL CONTENT OBJECT IS ENCRYPTED BASED ON THE PUBLIC KEY OF THE THIRD PARTY SERVICE
506
ENCRYPT ONE OR MORE NAME COMPONENTS OF THE INTEREST NAME BEGINNING WITH THE NAME COMPONENT IMMEDIATELY FOLLOWING THE MINIMUM ROUTABLE PREFIX
508
TRANSMIT THE INTEREST BASED ON THE ENCRYPTED NAME
510
RETURN

FIG. 5

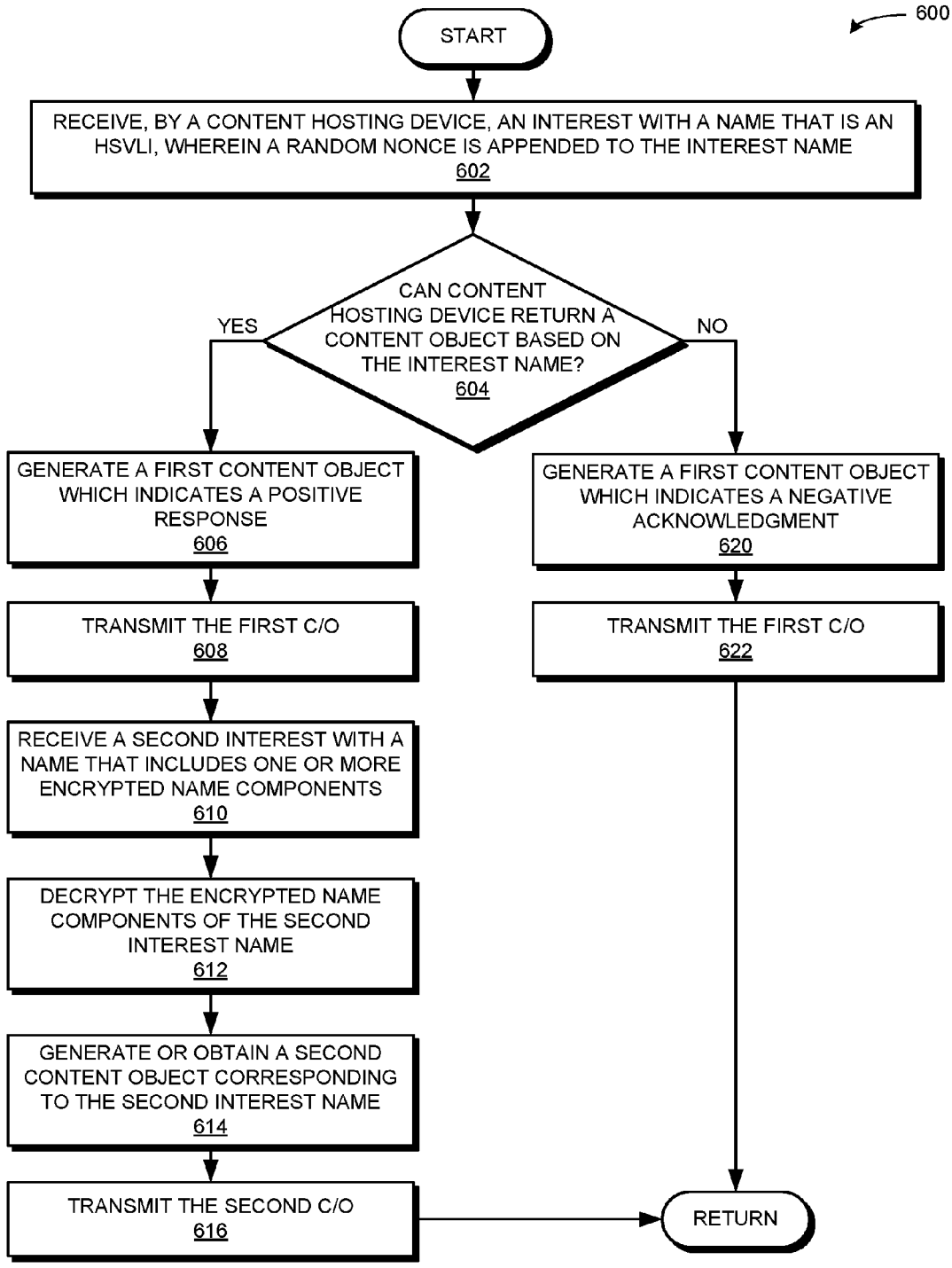

600
START
RECEIVE, BY A CONTENT HOSTING DEVICE, AN INTEREST WITH A NAME THAT IS AN HSVLI, WHEREIN A RANDOM NONCE IS APPENDED TO THE INTEREST NAME
602
CAN CONTENT HOSTING DEVICE RETURN A CONTENT OBJECT BASED ON THE INTEREST NAME?
604
YES
NO
GENERATE A FIRST CONTENT OBJECT WHICH INDICATES A POSITIVE RESPONSE
606
TRANSMIT THE FIRST C/O
608
RECEIVE A SECOND INTEREST WITH A NAME THAT INCLUDES ONE OR MORE ENCRYPTED NAME COMPONENTS
610
DECRYPT THE ENCRYPTED NAME COMPONENTS OF THE SECOND INTEREST NAME
612
GENERATE OR OBTAIN A SECOND CONTENT OBJECT CORRESPONDING TO THE SECOND INTEREST NAME
614
TRANSMIT THE SECOND C/O
616
GENERATE A FIRST CONTENT OBJECT WHICH INDICATES A NEGATIVE ACKNOWLEDGMENT
620
TRANSMIT THE FIRST C/O
622
RETURN

FIG. 6

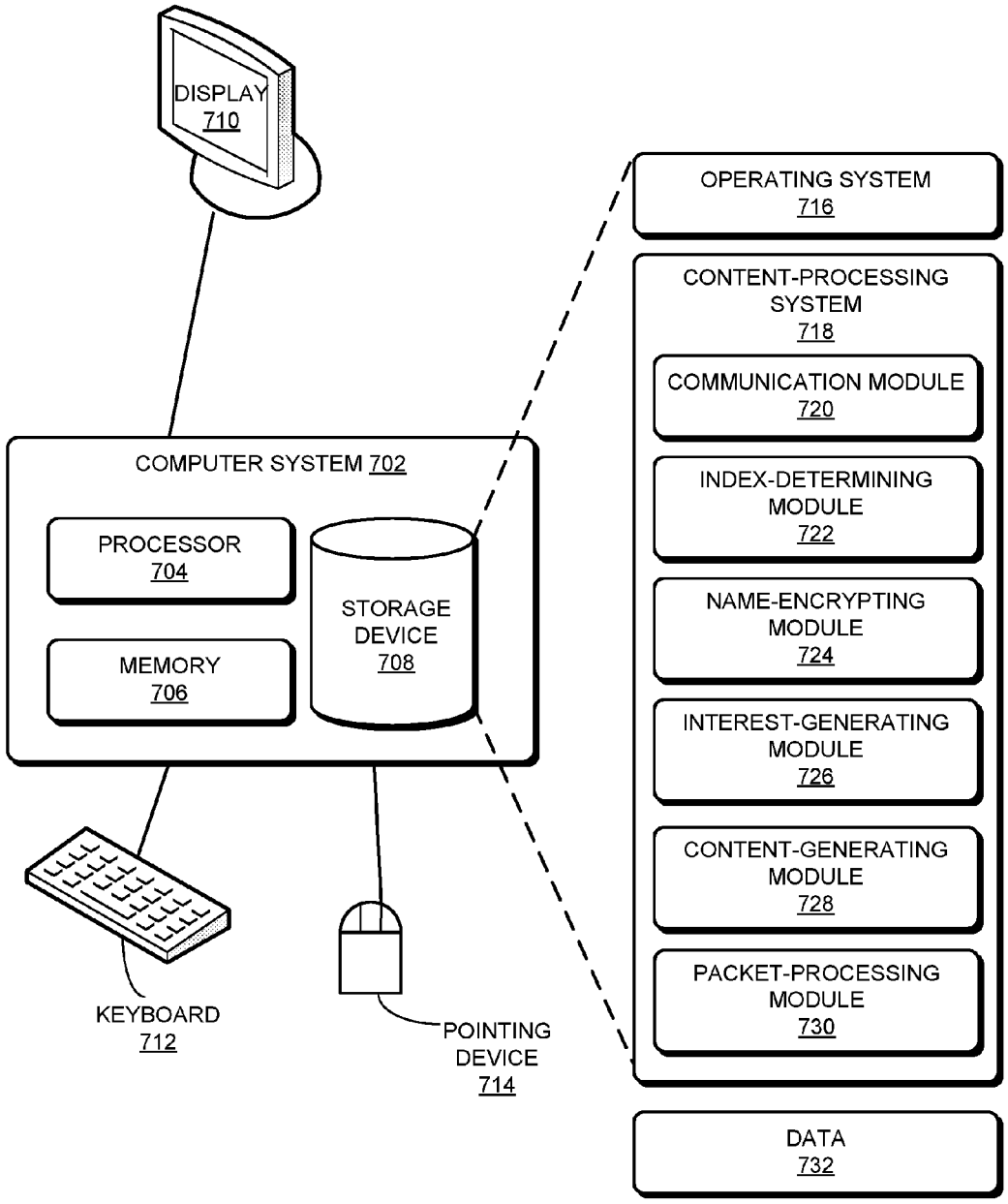

DISPLAY
710
OPERATING SYSTEM
716
CONTENT-PROCESSING SYSTEM
718
COMMUNICATION MODULE
720
INDEX-DETERMINING MODULE
722
NAME-ENCRYPTING MODULE
724
INTEREST-GENERATING MODULE
726
CONTENT-GENERATING MODULE
728
PACKET-PROCESSING MODULE
730
COMPUTER SYSTEM 702
PROCESSOR
704
MEMORY
706
STORAGE DEVICE
708
KEYBOARD
712
POINTING DEVICE
714
DATA
732

FIG. 7

METHOD AND SYSTEM FOR NAME ENCRYPTION AGREEMENT IN A CONTENT CENTRIC NETWORK

RELATED APPLICATIONS

[0001] The subject matter of this application is related to the subject matter in the following applications:

[0002] U.S. patent application Ser. No. 13/847,814 (Attorney Docket No. PARC-20120537-US-NP), entitled "ORDERED-ELEMENT NAMING FOR NAME-BASED PACKET FORWARDING," by inventor Ignacio Solis, filed 20 Mar. 2013 (hereinafter "U.S. patent application Ser. No. 13/847,814");

[0003] U.S. patent application Ser. No. 12/338,175 (Attorney Docket No. PARC-20080626-US-NP), entitled "CONTROLLING THE SPREAD OF INTERESTS AND CONTENT IN A CONTENT CENTRIC NETWORK," by inventors Van L. Jacobson and Diana K. Smetters, filed 18 Dec. 2008 (hereinafter "U.S. patent application Ser. No. 12/338,175"); and

[0004] U.S. patent application Ser. No. 14/947,810 (Attorney Docket No. PARC-20150245US01), entitled "TRANSPARENT ENCRYPTION IN A CONTENT CENTRIC NETWORK," by inventor Christopher A. Wood, filed 20 Nov. 2015 (hereinafter "U.S. patent application Ser. No. 14/947,810");

the disclosures of which are herein incorporated by reference in their entirety.

BACKGROUND

[0005] Field

[0006] This disclosure is generally related to distribution of digital content. More specifically, this disclosure is related to a method and system for name encryption agreement which allows a consumer to determine an index in a CCN name at which to begin encryption, based on a minimum routable prefix necessary for the interest to reach a producer in a content centric network.

[0007] Related Art

[0008] The proliferation of the Internet and e-commerce continues to create a vast amount of digital content. Content centric network (CCN) architectures have been designed to facilitate accessing and processing such digital content. A CCN includes entities, or nodes, such as network clients, forwarders (e.g., routers), and content producers, which communicate with each other by sending interest packets for various content items and receiving content object packets in return. CCN interests and content objects are identified by their unique names, which are typically hierarchically structured variable length identifiers (HSVLI). An HSVLI can include contiguous name components ordered from a most general level to a most specific level.

[0009] A CCN data packet (such as an interest or content object) is routed based on its name. Some name components may be used by an intermediate node to route a CCN interest, while other name components may be used by a content producer to satisfy a request based on private user information or application-specific data. In the latter case, the meaningfulness of the name components may reveal information regarding the requested content and may result in a breach of user privacy or security. A consumer may encrypt the interest name, but a sufficient number of name components must remain unencrypted for routing purposes. This "minimum routable prefix" is the maximal name length (e.g., maximum number of name components) needed to route an interest to a content producer who can satisfy the content request.

[0010] While a CCN brings many desired features to a network, some issues remain unsolved for a consumer in determining the minimum routable prefix for an interest name.

SUMMARY

[0011] One embodiment provides a system that facilitates efficient name encryption in a CCN. During operation, the system determines, by a client computing device, an index for a name of an interest, wherein the name is a hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level, wherein the index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest. The system encrypts one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix. The system transmits the interest based on the encrypted name, thereby facilitating efficient name encryption in a CCN.

[0012] In some embodiments, the system generates a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level. In response to receiving a first content object which indicates a positive response of the first probing interest, the system sets the index to a number of name components in the first probing interest name. In response to receiving a second content object which indicates a negative acknowledgment of the first probing interest, the system generates a second probing interest with a name that comprises the first probing interest name followed by a next contiguous name component of the interest name.

[0013] In some embodiments, the system determines that the first content object indicates a positive response of the first probing interest. The system determines that the first content object indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest name. The system also determines that a key identifier of the first content object matches a public key of the content producing device.

[0014] In some embodiments, the system appends a first random nonce to the first probing interest name. The system also appends a second random nonce to the second probing interest name.

[0015] In some embodiments, the system determines a midpoint index of a number of name components in the interest name, wherein a lower portion of the interest name includes the name components from the most general level name component to the name component preceding the name component corresponding to the midpoint index, and wherein an upper portion of the interest name includes the name components from the name component following the name component corresponding to the midpoint index to the most specific level name component. The system generates a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to

the name component corresponding to the midpoint index. In response to receiving a first content object which indicates a positive response of the first probing interest, the system sets the index to a number of name components in the first probing interest name.

[0016] In some embodiments, in response to receiving a second content object which indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest, and in response to determining that a key identifier of the second content object is associated with a public key of the content producing device, the system determines a lower midpoint index of the lower portion. The system generates a second probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the lower midpoint index. In response to receiving a third content object which indicates a negative acknowledgment of the first probing interest, the system determines an upper midpoint index of the upper portion, and generates a third probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the upper midpoint index.

[0017] In some embodiments, the system appends a first random nonce to the first probing interest name, appends a second random nonce to the second probing interest name, and appends a third random nonce to the third probing interest name.

[0018] In some embodiments, the system generates one or more probing interests based on a number of number components for the interest name and further based on one or more of: a linear search; a binary search; and a number of collapsed name prefixes in a forwarding information base, wherein a collapsed name prefix indicates a plurality of name components with a same forwarding information in the forwarding information base.

[0019] In some embodiments, the system generates an initial interest for the index, wherein the initial interest is transmitted to a third party service and has a payload that includes the interest name and a public key of the third party service, wherein the payload of the initial interest is encrypted based on a public key of the client computing device, wherein the initial interest indicates the public key of the client computing device. In response to the initial interest, the system receives an initial content object that has a payload that indicates the index, wherein the payload of the initial content object is encrypted based on the public key of the third party service.

[0020] In some embodiments, the interest name includes one or more nested and encrypted names suffixes, and a name suffix comprises one or more contiguous name components of the interest name. The system determines a second index for a nested and encrypted name suffix, wherein the second index indicates a minimum number of contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy a nested interest with a name which includes the nested and encrypted name suffix. The system encrypts the name components following the name components corresponding to the second index.

BRIEF DESCRIPTION OF THE FIGURES

[0021] FIG. **1** illustrates an exemplary environment which facilitates efficient name encryption in a content centric network, in accordance with an embodiment of the present invention.

[0022] FIG. **2**A illustrates an exemplary communication which facilitates efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention.

[0023] FIG. **2**B illustrates an exemplary communication which facilitates efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention.

[0024] FIG. **2**C illustrates an exemplary communication which facilitates efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention.

[0025] FIG. **3** illustrates an exemplary communication which facilitates efficient name encryption in a content centric network, including communication with a third party service, in accordance with an embodiment of the present invention.

[0026] FIG. **4**A presents a flow chart illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, in accordance with an embodiment of the present invention.

[0027] FIG. **4**B presents a flow chart illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention.

[0028] FIG. **4**C presents a flow chart illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention.

[0029] FIG. **4**D presents a flow chart illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention.

[0030] FIG. **5** presents a flow chart illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, including communication with a third party service, in accordance with an embodiment of the present invention.

[0031] FIG. **6** presents a flow chart illustrating a method by a content-hosting device for facilitating efficient name encryption in a content centric network, in accordance with an embodiment of the present invention.

[0032] FIG. **7** illustrates an exemplary computer system that facilitates efficient name encryption in a content centric network, in accordance with an embodiment of the present invention.

[0033] In the figures, like reference numerals refer to the same figure elements.

DETAILED DESCRIPTION

[0034] The following description is presented to enable any person skilled in the art to make and use the embodiments, and is provided in the context of a particular application and its requirements. Various modifications to the disclosed embodiments will be readily apparent to those

skilled in the art, and the general principles defined herein may be applied to other embodiments and applications without departing from the spirit and scope of the present disclosure. Thus, the present invention is not limited to the embodiments shown, but is to be accorded the widest scope consistent with the principles and features disclosed herein.

Overview

[0035] Embodiments of the present invention solve the problem of efficiently encrypting a CCN name by providing a system which allows a consumer to determine the minimum routable prefix of a CCN name, which indicates the index in the name at which to begin encryption. A CCN data packet (e.g., an interest or a content object) is routed based on its name, which can include multiple name components. Some of the name components may be used for routing purposes, while other name components may contain sensitive user information or application-specific data. A consumer may encrypt the interest name, but a sufficient number of name components must remain unencrypted in order for the interest to be routed to a producer that can satisfy the interest or serve the requested content. Embodiments of the present system allow a consumer to determine this sufficient number of unencrypted name components, which is also known as the minimum routable prefix. The minimum routable prefix can correspond to an index in the CCN name, where the index indicates the position of a particular name component in the hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level.

[0036] The consumer can discover the index based on three different methods: 1) a name-based negotiation protocol; 2) a route-based negotiation protocol; and 3) an explicit negotiation protocol. Name-based negotiation (the first method) can be based on a linear probing method or a binary probing method. The consumer can send probing interests with an increasing number of name components until a positive response is returned. For example, based on the linear probing method, given a name N of "/a/b/c/d/x/y/z," and a random nonce rx, the consumer can transmit a probing interest with the name "laid" and if a negative response is received, the consumer can transmit another probing interest with the name "/a/b/r2." The consumer can continue sending probing interests, each with an additional name component, until it receives a positive response. The positive response can indicate the minimum routable prefix needed to properly route the interest, and thus can indicate the index within the name N at which the consumer may begin encryption. The name-based negotiation protocol using linear probing is described below in relation to FIGS. **2**A and **2**B. The consumer can also perform the name-based negotiation protocol based on binary probing, which is described below in relation to FIG. **2**C.

[0037] Route-based negotiation (the second method) is an extension of name-based negotiation. The routing algorithm can account for the number of prefixes truncated or collapsed during publication. Each CCN node has a forwarding information base ("FIB"), which is a table with entries of name prefixes and corresponding outgoing interfaces. The FIB is used to route interests based on longest-prefix matches of their names. A FIB entry usually contains one name prefix and its corresponding outgoing interfaces. If two or more name prefixes correspond to the same outgoing interface, the CCN node may collapse or truncate the entries into one entry.

[0038] In the explicit negotiation protocol (the third method), a CCN producer delegates the negotiation to a third party service which is known to a consumer. The consumer can send an explicit request to the third party service for the index. A detailed description of the protocol based on explicit negotiation with the third party service is described below in relation to FIG. **3**.

[0039] Thus, the system facilitates efficient name encryption in a CCN by allowing a consumer to discover the minimum routable prefix for an interest, which indicates a maximum number of name components needed to route the interest to a producer. The minimum routable prefix also indicates the index at which the consumer may begin encrypting the name.

[0040] In CCN, each piece of content is individually named, and each piece of data is bound to a unique name that distinguishes the data from any other piece of data, such as other versions of the same data or data from other sources. This unique name allows a network device to request the data by disseminating a request or an interest that indicates the unique name, and can obtain the data independent from the data's storage location, network location, application, and means of transportation. The following terms are used to describe the CCN architecture:

[0041] Content Object (or "content object"): A single piece of named data, which is bound to a unique name. Content Objects are "persistent," which means that a Content Object can move around within a computing device, or across different computing devices, but does not change. If any component of the Content Object changes, the entity that made the change creates a new Content Object that includes the updated content, and binds the new Content Object to a new unique name.

[0042] Unique Names: A name in a CCN is typically location independent and uniquely identifies a Content Object. A data-forwarding device can use the name or name prefix to forward a packet toward a network node that generates or stores the Content Object, regardless of a network address or physical location for the Content Object. In some embodiments, the name may be a hierarchically structured variable-length identifier (HSVLI). The HSVLI can be divided into several hierarchical components, which can be structured in various ways. For example, the individual name components parc, home, ccn, and test.txt can be structured in a left-oriented prefix-major fashion to form the name "/parc/home/ccn/test.txt." Thus, the name "/parc/home/ccn" can be a "parent" or "prefix" of "/parc/home/ccn/test.txt." Additional components can be used to distinguish between different versions of the content item, such as a collaborative document. The HSVLI can also include contiguous name components ordered from a most general level to a most specific level.

[0043] In some embodiments, the name can include an identifier, such as a hash value that is derived from the Content Object's data (e.g., a checksum value) and/or from elements of the Content Object's name. A description of a hash-based name is described in U.S. patent application Ser. No. 13/847,814, which is herein incorporated by reference. A name can also be a flat label. Hereinafter, "name" is used to refer to any name for a piece of data in a name-data network, such as a hierarchical name or name prefix, a flat

name, a fixed-length name, an arbitrary-length name, or a label (e.g., a Multiprotocol Label Switching (MPLS) label).

[0044] Interest (or "interest"): A packet that indicates a request for a piece of data, and includes a name (or a name prefix) for the piece of data. A data consumer can disseminate a request or Interest across an information-centric network, which CCN/NDN routers can propagate toward a storage device (e.g., a cache server) or a data producer that can provide the requested data to satisfy the request or Interest.

[0045] The methods disclosed herein are not limited to CCN networks and are applicable to other architectures as well. A description of a CCN architecture is described in U.S. patent application Ser. No. 12/338,175, which is herein incorporated by reference.

Exemplary Network and Communication

[0046] FIG. **1** illustrates an exemplary environment which facilitates efficient name encryption in a content centric network, in accordance with an embodiment of the present invention. A network **100** can include a consumer or content requesting device **116**, producers or content producing devices **118** and **120**, and a router or other forwarding device at nodes **102**, **104**, **106**, **108**, **110**, **112**, and **114**. A node can be a computer system, an end-point representing users, and/or a device that can generate interests or originate content. A node can also be an edge router (e.g., CCN nodes **102**, **104**, **112**, and **114**) or a core router (e.g., intermediate CCN routers **106**, **108**, and **110**). Network **100** can be a content centric network.

[0047] During operation, consumer or client computing device **116** can determine, for a name N of "/a/b/c/d/x/y/z," an index at which device **116** may begin encrypting the name N (get index function **170**, described in detail below in relation to FIGS. **2**A-**2**C and **3**). This index may be referred to as the "split index." The split index can indicate "3" as the "minimum_routable_prefix," which also indicates the remainder of the name N as the "sensitive_name" that can be encrypted. In other words, the split index can indicate the name prefix of the name N through the name component whose position index is equal to "3" (e.g., "a/b/c/d"), and can also indicate the name components following the name component whose position index is equal to 3 that can be encrypted (e.g., "/x/y/z"). Device **116** can generate an interest **150** with a name **150**.**1** of "/minimum_routable_prefix/E_{Ck}(/sensitive_name)," where "Ck" is the public key of consumer or device **116**. Interest **150** can also include an optional payload **150**.**2** with a value of "<data>."

[0048] Interest **150** can travel through network **100** via nodes **102**, **110**, and **112**, before reaching producer or content producing device **118**. Device **118** can serve content or satisfy requests for content with the prefix of "/a/b/c/d" or "minimum_routable_prefix." Assume that device **118** is in possession of or has a way to retrieve the public key of device **116**. Device **118** can decrypt the encrypted portion of name **150**.**1** of interest **150** (function **180**), and generate a content object **160** corresponding to the name "/minimum_routable_prefix/sensitive_data" (function **182**). Device **118** can replace a name **160**.**1** in content object **160** with the original partially encrypted name (e.g., name **150**.**1** with a value of "/minimum_routable_prefix/E_{Ck}(/sensitive_name)"), and transmits content object **160** to device **118** on a reverse path (e.g., via nodes **112**, **110**, and **102**).

Name-Based Negotiation Based on Linear Probing

[0049] A consumer can determine the split index (which indicates the minimum routable prefix) for a given name using a name-based negotiation by sending probing interests with an increasing number of name components. FIG. **2**A illustrates an exemplary communication **200** which facilitates efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention. Device **202** can be a consumer or a client computing device. Nodes **204** and **206** can be intermediate nodes or routers or content-hosting devices that can forward an interest with the prefix "/a." Node **208** can be an intermediate node or router or a content-hosting device that can forward an interest for the prefix "/a/b." Node **210** can be an intermediate node or router or a content-hosting device that can serve content for the prefix "/a/b/c/d." For the sake of illustration, nodes **204**-**208** are depicted as intermediate routers, and device **210** is depicted as a server, but any of entities **204**-**210** can be an intermediate router or a content-hosting device that can serve content (as described above).

[0050] Assume that a name N **280** hasp name components, N_1-N_p, e.g., for a name N of "/a/b/c/d/x/y/z," p is equal to 7. The determined split index i indicates that all components N_j where j is greater than i may be encrypted. In addition, the consumer can generate for each probe interest a random nonce rx that is appended to the name for a respective probing interest.

[0051] During operation, a consumer or a client computing device **202** can send a set of probes **200**.**1**-**200**.**4** to determine the split index i. For example, device **202** can generate and transmit an interest **220** with a name **220**.**1** of "/a/r1," which is a probing interest to determine whether the split index i=1. Interest **220** can travel to a node **204**, which can determine based on its local FIB to forward interest **220** to node **206**. Node **206** can determine based on its local FIB that no route exists for name **220**.**1**. Node **206** can return a negative acknowledgment to device **202** in the form of a content object **222** with a name **222**.**1** of "laid" and a payload **222**.**2** with a value of "NACK."

[0052] Device **202** can receive the NACK of content object **222**, and determine to send another probing interest with an additional name component. Device **202** can generate and transmit an interest **224** with a name **224**.**1** of "/a/b/r2," which is a probing interest to determine whether the split index i=2. Interest **224** can reach node **204**, which forwards interest **224** to node **206**, which in turn forwards interest **224** to node **208**. Node **208** can determine based on its local FIB that no route exists for name **224**.**1**. Node **208** can return a negative acknowledgment to device **202** in the form of a content object **226** with a name **226**.**1** of "/a/b/r2" and a payload **226**.**2** with a value of "NACK."

[0053] Device **202** can receive the NACK of content object **226**, and determine to send another probing interest with an additional name component. Device **202** can generate and transmit an interest **228** with a name **228**.**1** of "/a/b/c/r3," which is a probing interest to determine whether the split index i=3. Interest **228** can reach node **204**, which forwards interest **228** to node **206**, which in turn forwards interest **228** to node **208**. Node **208** can determine based on its local FIB that no route exists for name **228**.**1**. Node **208** can return a negative acknowledgment to device **202** in the form of a content object **230** with a name **230**.**1** of "/a/b/c/r3" and a payload **230**.**2** with a value of "NACK."

[0054] Finally, device **202** can receive the NACK of content object **230**, and determine to send another probing interest with an additional name component. Device **202** can generate and transmit an interest **232** with a name **232.1** of "/a/b/c/d/r4," which is a probing interest to determine whether the split index i=4. Interest **232** can reach node **204**, which forwards interest **232** to node **206**, which in turn forwards interest **232** to node **208**, which in turn forwards interest **232** to device **210**. Device **210** can determine that it can serve content under the prefix "/a/b/c/d," but that the content corresponding to name **232.1** does not exist ("DNE"). Device **210** can return a positive acknowledgment to device **202** in the form of a content object **234** with a name **234.1** of "/a/b/c/d/r4" and a payload **234.2** with a value of "DNE."

[0055] Device **202**, in possession of a positive acknowledgment from probes **201.1**-**201.4**, can determine that content object **234** indicates that a content producing device can return a content object with the minimum routable prefix of "/a/b/c/d." Device **202** can also determine that the key identifier of content object **234** matches the key identifier of the public key of content producing device **210**. This allows device **202** to determine that the minimum routable prefix for the name N of "/a/b/c/d/x/y/z" is "/a/b/c/d," and that the split index i is equal to 4 (or 3, when the index count begins at zero instead of at one).

[0056] FIG. **2**B illustrates an exemplary communication **240** which facilitates efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention. FIG. **2**B corresponds to FIG. **2**A. Device **202** can use the determined split index i=4 to encrypt the name components of name N starting from the name component following the name component at index **4** (e.g., after the minimum routable prefix name of "/a/b/c/d"). Device **202** can generate and transmit an encrypted interest **250** with a name **250.1** of "/a/b/c/d/E_{Ck}(/x/y/z)," which interest travels via nodes **204**, **206**, and **208** until it reaches device **210**. Device **210** can decrypt the encrypted portion of name **250.1** based on a public key of device **202**, and generate a responsive content object **252** with a payload **252.2** of "<data>" that corresponds to the unencrypted name. Device **210** can further replace the unencrypted name with a name **252.1** of "a/b/c/d/E_{Ck}(/x/y/z)," which matches name **250.1** of interest **250**. Device **210** can then return content object **252** to device **202** along a reverse path.

[0057] Device **210** can also obtain the content corresponding to the decrypted name from a different entity in the network. Thus, device **210** can generate and transmit an interest **254** with a name **254.1** of "/a/b/c/d/x/y/z," and receive a responsive content object **256** with a name **256.1** of "/a/b/c/d/x/y/z" and a payload **256.2** of "<data>." Device **210** can subsequently create a content object **252** as described above (by replacing name **256.1** with name **252.1**), and return **252** to device **202** along the reverse path.

[0058] An example of pseudocode for a linear probe function is provided herein:

```
def LinearProbe (N, low, high):
    for i = low to high do
        Ri := GenerateRandomNonce( )
        Probe := [N1, ..., Ni].Append(Ri)
        Content Object = RequestInterestWithName(Probe)
        if (ContentObject == DNE and
            ContentObject.KeyId == KeyId(pk))
                return i
        end
    done
return −1 // error
```

-continued

[0059] Thus, the consumer can perform a name-based negotiation using a linear probing method to determine the split index for subsequent encryption of an interest name. Using only the name N, the consumer can call the function as:

$$\text{split\_index}=\text{LinearProbe}(N,0,\text{len}(N)-1) \qquad \text{Function (1)}$$

The term "len(N)" is equal to the number of name components in N, and the function LinearProbe( ) is performed on a zero-based index count.

Name-Based Negotiation Based on Binary Probing

[0060] FIG. **2**C illustrates an exemplary communication **260** which facilitates efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention. FIG. **2**C includes device **202** and nodes **204**-**210**, which correspond to the same entities depicted in FIG. **2**A. FIG. **2**C additionally includes a node **212** which can be an intermediate node or router or a content-hosting device that can forward an interest with the prefix "/a/b/c/d/x," and a node or device **214** which can be an intermediate node or router or a content-hosting device that can serve content for the prefix "/a/b/c/d/x/y."

[0061] Assume that a name **282** hasp name components, M_1-M_p, e.g., for a name M **282** of "/a/b/c/d/x/y," p is equal to 6. Note that the index count shown for name M **282** is a zero-based count, i.e., the index number begins from zero, which is different from the index count shown for name N **280** of FIG. **2**A, which begins from "1." During operation, a consumer or a client computing device **202** can send a set of probes **261.1**-**261.3** to determine the split index. Device **202** can determine a midpoint target index of t=2 for the name components M_1-M_p, where a lower portion of the name consists of the name components from M_1-M_{t-1}, and an upper portion of the name consists of the name components from M_{t+1} to M_p. Device **202** can generate and transmit an interest **262** with a name **262.1** of "/a/b/c/r2." Interest **262** can reach node **204**, which forwards interest **262** to node **206**, which in turn forwards interest **262** to node **208**. Node **208** can determine based on its local FIB that no route exists for name **262.1**. Node **208** can return a negative acknowledgment to device **202** in the form of a content object **264** with a name **264.1** of "/a/b/c/r3" and a payload **264.2** with a value of "NACK."

[0062] Based on the NACK of content object **264**, device **202** can determine to continue the binary probe on the upper portion of the name. Device **202** can determine a new midpoint target index t=4 of the upper portion (and again determine a new lower and upper portion of the name based on the new midpoint target index). Device **202** can generate and transmit an interest **266** with a name **266.1** of "/a/b/c/d/x/r4." Interest **266** can reach node **204**, which forwards interest **266** to node **206**, which in turn forwards interest **266** to node **208**, which in turn forwards interest **266** to node **210**, which in turn forwards interest **266** to node **212**. Node **212**

can determine that it can serve content under the prefix "/a/b/c/d/x," but that the content corresponding to name **266.1** does not exist ("DNE"). Node or device **212** can return a positive acknowledgment to device **202** in the form of a content object **268** with a name **268.1** of "/a/b/c/d/x/r4" and a payload **234.2** with a value of "DNE." Node **212** can further include in content object **268** a KeyId **268.3** which indicates that its KeyId anchors or is associated with the public key of a producer of content for a number of name components less than t+1 (e.g., which allows consumer or client computing device **202** to determine that the key identifier for content object **268** is associated with the key identifier of a public key of a content producing device that can serve the requested content).

[0063] Upon receiving content object **268**, device **202** can determine from the DNE of payload **268.2** and the anchor indication of KeyId **268.3** to continue the binary probe search on the (new) lower portion of the name. Device **202** can determine an updated midpoint target index t=3 of the (new) lower portion (and, if necessary, determine an updated lower and upper portion of the name based on the updated midpoint target index). Device **202** can generate and transmit an interest **270** with a name **270.1** of "/a/b/c/d/r3." Interest **270** can reach node **204**, which forwards interest **270** to node **206**, which in turn forwards interest **270** to node **208**, which in turn forwards interest **270** to node **210**. Node or device **210** can determine that it can serve content under the prefix "/a/b/c/d/x," but that the content corresponding to name **270.1** does not exist ("DNE"). Device **210** can return a positive acknowledgment to device **202** in the form of a content object **272** with a name **272.1** of "/a/b/c/d/x/r4" and a payload **272.2** with a value of "DNE." Device **210** can further include in content object **272** a KeyId **272.3** which indicates that its KeyId matches the public key of a producer of content for a number of name components equal to t (e.g., which allows consumer or client computing device **202** to determine that the key identifier for content object **272** matches the key identifier of a public key of a content producing device that can serve the requested content).

[0064] Device **202**, in possession of a positive acknowledgment from probes **261.1**-**261.3**, can determine that content object **272** indicates that a content producing device can return a content object with the minimum routable prefix of "/a/b/c/d." Device **202** can also determine that the key identifier of content object **272** matches the key identifier of the public key of the content producing device. This allows device **202** to determine that the minimum routable prefix for the name N of "/a/b/c/d/x/y/z" is "/a/b/c/d," and that the split index i is equal to 3 (in the case of a zero-based index count). Device **202** can subsequently send an encrypted interest **250**, as shown in relation to communication **240** of FIG. **2**B.

[0065] An example of pseudocode for a binary probe function is provided herein:

```
def BinaryProbe(N, low, high):
    i := ((high - low) / 2)
    visited = [ ]
    while len(visited) < (high - low) do
        Ri := GenerateRandomNonce( )
        Probe := [N1, ..., Ni, ..., N(low + i)]. Append(Ri)
        ContentObject = RequestInterestWithName(Probe)
        if (KeyId(pk) anchors ContentObject.KeyId and
            ContentObject == DNE) then
                visited.Append(i + low); i := i - (i / 2)
        elseif (ContentObject.KeyId == KeyId(pk) and
            ContentObject == DNE)
                return i + low
        else // != KeyId or a NACK (P cannot serve probe)
                visited.Append(i + low); i := i - (i / 2)
    end
    return -1 // error
```

-continued

[0066] Thus, the consumer can perform a name-based negotiation using a binary probing method to determine the split index for subsequent encryption of an interest name. Using only the name N, the consumer can call the function as:

$$split\_index=BinaryProbe(N,0,len(N)-1) \qquad \text{Function (2)}$$

The term "len(N)" is equal to the number of name components in N, and the function BinaryProbe( ) is performed on a zero-based index count.

[0067] The consumer can also generate and transmit nested probing interests within each other. The probing interests can be sent and processed similar to onion routing, where each gateway or decrypting node acts as an application-layer gateway or forwarder for the nested probe on behalf of the original issuer. For example, for an interest with a name N of "/a/b/c/MARK</d/e/f/MARK</g/h/i/>>," where "MARK" indicates that the following suffix is encrypted, the consumer can send out probing interests that corresponds to each "layer" of the name. A first probing interest may include a name NO of "/a/b/c/MARK</d/e/f/MARK</g/h/i/>>," which can return a first index that corresponds to a minimum routable prefix for the outer layer. A second probing interest may include a name N1 of "/a/b/c/d/e/f/MARK<g/h/i/>," which can return a second index that corresponds to a minimum routable prefix for that respective layer. Finally, a third probing interest may include a name N2 of "/a/b/c/d/e/f/g/h/i/," which can return a third index that corresponds to a minimum routable prefix for that respective layer. In addition, name prefixes may be inherited (as described in the example above), or name prefixes may not be inherited, e.g.: N0="/a/b/c/MARK</d/e/f/MARK</g/h/i/>>"; N1="/d/e/f/MARK</g/h/i/"; and N2="/g/h/i."

Route-Based Negotiation

[0068] An extension of the name-based negotiation protocol is route-based negotiation, where the routing algorithm takes into account the number of prefixes that were truncated or collapsed during publication. Recall that a CCN node has a forwarding information base ("FIB") which is a table with entries of name prefixes and corresponding outgoing interfaces. If two or more name prefixes correspond to the same outgoing interface, the CCN node may collapse or truncate the entries into one entry. A local FIB can contain the minimum number of hops to the nearest anchor for a given prefix. For example, in FIG. **2**A, if the local FIB for node **204** has a first entry for the name prefix "/a" which corresponds to an interface(s) associated with node **206**, and a second entry for the name prefix "/a/b" which also corresponds to an interface(s) associated with node **208**, then node **204** can collapse or truncate the names prefixes in its FIB. If 1* indicates the total number of name prefixes in the

name (taking into account the collapsed name prefixes under this extension to the protocol), the consumer can call the function as:

split_index=LinearProbe(N,1*,len(N)−1); or Function (3)

split_index=BinaryProbe(N,1*,len(N)−1). Function (4)

Explicit Negotiation Protocol

[0069] FIG. **3** illustrates an exemplary communication **300** which facilitates efficient name encryption in a content centric network, including communication with a third party service, in accordance with an embodiment of the present invention. FIG. **3** includes a consumer or client computing device **202**, a content-hosting or content-producing device **210**, and a third party service ("R") **216** to which device **210** has delegated responsibility in the negotiation protocol for determining the split index. Device **202** can perform an explicit request for the split index for a target name N. Device can **202** can also possess "Rk," the public key of R **216**. For example, for a target name N of "/a/b/c/d/e/f," device **202** can perform a get index **300.1** function by sending an interest **310** with a name **310.1** of "/R/routable/prefix/get index," a payload **310.2** of "E_{Ck}(Rk, "/a/b/c/d/e/f)," and a reserved field **310.3** which indicates a public key certificate of device **202** (e.g., "<client_202_pk_certificate>"). Third party service R **216** can receive interest **310**, decrypt the encrypted payload **310.2** to obtain the target name N, and determine the appropriate split index i. R **216** can generate and transmit a content object **312** with a name **312.1** of "/R/routable/prefix/get_index" and a payload **312.2** of "E_{Ck}(i, "/a/b/c/d/e/f)."

[0070] Upon discovering the split index i, device **202** can retrieve content via a get content **300.2** function by generating and transmitting an interest **320** with a name **320.1** of "/a/b/c/d/E_{Ck}(/e/f)" and an optional payload **320.2** of "<data>." Device **210** can receive interest **320**, decrypt the encrypted portion of the name **320.1**, generate a responsive content object **322** with a name **322.1** which matches the encrypted name **320.1** of interest **320** and includes a payload **322.2** with a value of "<data>." Device **210** can transmit content object **322** to device **202** along a reverse path.

Role of Client-Computing Device in Facilitating Efficient Name Encryption

[0071] FIG. **4**A presents a flow chart **400** illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, in accordance with an embodiment of the present invention. During operation, the system determines, by a client computing device, an index for a name of an interest, wherein the name is an HSVLI that includes contiguous name components ordered from a most general level to a most specific level (operation **402**). The index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest. The operation can continue based on a linear probe (as indicated at Label A of FIG. **4**B) or based on a binary probe (as indicated at Label B of FIG. **4**C). The system encrypts one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix (operation **404**). The system transmits the interest based on the encrypted name (operation **406**).

[0072] FIG. **4**B presents a flow chart **410** illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a linear probing method, in accordance with an embodiment of the present invention. During operation, the system generates a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level (operation **412**). The system can generate and append a random nonce to a probing interest, as described above in relation to FIG. **2**A. The system determines whether it receives a first content object that indicates a positive response (decision **414**). If it does, the system determines that the first content object indicates that a content producing device can return a content object based on the name components of the interest name as included in the first probing interest (operation **420**). The system further determines that a key identifier of the first content object matches a public key of the content producing device (operation **422**). The system then sets the index to the number of name components in the first probing interest (operation **424**). The positive response can also be indicated with any other indicator, such as a notification flag or a reserved field or bit.

[0073] If the system determines that it receives a first content object that is not a positive response (i.e., the first content object indicates a negative response) (decision **414**), the system generates a second probing interest with a name that comprises the first probing interest name followed by a next name component of the interest name (operation **416**). The system can replace the first probing interest with the second probing interest (for purposes of looping), and the operation returns to decision **414**, where the system determines whether it receives a first content object that is a positive response to the first probing interest (i.e., the second probing interest previously generated in operation **416**). The operations continue until a positive response is received, and the system performs operations **420**, **422**, and **424** as described above. If the operation reaches the end of the name (e.g., processes all name components) and does not return the index, the operation can return an error (not shown).

[0074] FIG. **4**C presents a flow chart **430** illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention. During operation, the system determines a midpoint index of the number of name components in the interest name (operation **432**). A lower portion of the interest name includes the name components from the most general level name component to the name component preceding the name component corresponding to the midpoint index, and an upper portion of the interest name includes the name components from the name component following the name component corresponding to the midpoint index to the most specific level name component. The system generates a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level to the name component corresponding to the midpoint index (operation **434**). The system can generate and append a random nonce to a probing interest, as described above in relation to FIG. **2**A.

[0075] The system determines whether it receives a first content object that indicates a positive response (decision **436**). If it does, the system determines that the first content object indicates that a content producing device can return a content object based on the name components of the interest name as included in the first probing interest (operation **438**). The system further determines that a key identifier of the first content object matches a public key of the content producing device (operation **440**). The system then sets the index to the number of name components in the first probing interest (operation **442**). The positive response can also be indicated with any other indicator, such as a notification flag or a reserved field or bit.

[0076] If the system determines that it receives a first content object that is not a positive response (i.e., the first content object indicates a negative response) (decision **436**), the operation continues as indicated at label C of FIG. **4**D. FIG. **4**D presents a flow chart **450** illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, based on a binary probing method, in accordance with an embodiment of the present invention. The system determines that the first content object does not indicate a positive response (operation **452**), and the operation can continue as depicted by operation **454** or by operation **460**. The system can determine that the first content object indicates that a content producing device can return a content object based on the interest name as included in the first probing interest, and can further determine that the key identifier of the first content object is associated with the public key of the content producing device (operation **454**). The system can determine a lower midpoint index of the lower portion of the interest name (operation **456**). The system can generate a second probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to the name component corresponding to the lower midpoint index (operation **458**). The system can replace the first probing interest with the second probing interest (for purposes of recursion), and the operation returns to decision **436**, where the system determines whether it receives a first content object that is a positive response to the first probing interest (i.e., the second probing interest previously generated in operation **458**). This begins the binary probe search again on the lower portion of the interest name.

[0077] Alternatively, after operation **452**, the system can determine that the first content object indicates a negative acknowledgment of the first probing interest (operation **460**). The system can determine an upper midpoint index of the upper portion of the interest name (operation **462**). The system can generate a third probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to the name component corresponding to the upper midpoint index (operation **464**). As described above, the system can replace the first probing interest with the second probing interest (for purposes of recursion), and the operation returns to decision **436**, which begins the binary probe search again on the upper portion of the interest name.

[0078] FIG. **5** presents a flow chart **500** illustrating a method by a client computing device for facilitating efficient name encryption in a content centric network, including communication with a third party service, in accordance with an embodiment of the present invention. During operation, the system determines, by a client computing device, an index for a name of an interest, wherein the name is an HSVLI that includes contiguous name components ordered from a most general level to a most specific level (operation **502**). The index indicates a minimum number of the contiguous name components beginning from the most general level, wherein the minimum number indicates a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest. The system generates an initial interest for the index, wherein the initial interest is transmitted to a third party service and has a payload that includes the interest name and a public key of the third party service (operation **504**). The payload of the initial interest is encrypted based on a public key of the client computing device, and the initial interest indicates the public key of the client computing device.

[0079] In response to the initial interest, the system receives an initial content object that has a payload that indicates the index, wherein the payload of the initial content object is encrypted based on the public key of the third party service (operation **506**). The system encrypts one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix (operation **508**). Subsequently, the system transmits the interest based on the encrypted name (operation **510**).

Role of Content-Hosting Device in Facilitating Efficient Name Encryption

[0080] FIG. **6** presents a flow chart **600** illustrating a method by a content-hosting device for facilitating efficient name encryption in a content centric network, in accordance with an embodiment of the present invention. During operation, the system receives, by a content-hosting device, an interest with a name that is an HSVLI, wherein a random nonce is appended to the interest name (operation **602**). The content-hosting device determines whether it can return a content object based on the interest name (decision **604**). For example, if the device can serve content under the prefix "/a/b/c" and if the interest name is "/a/b/c/<nonce>," the device can determine that it can serve content under the prefix "/a/b/c" but that the content object with the name of "/a/b/c/<nonce>" does not exist. The device can generate a first content object which indicates a positive response (e.g., "does not exist" or "DNE") (operation **606**).

[0081] The device can transmit the first content object (operation **608**). The device can subsequently receive a second interest with a name that includes one or more encrypted name components (operation **612**). The device can decrypt the encrypted name components of the second interest name (operation **612**). The device can generate or obtain a second content object corresponding to the decrypted second interest name (operation **614**). The device can replace the decrypted second interest name with the partially encrypted second interest name in the second content object (not shown), and transmit the second content object (operation **616**).

[0082] If the content-hosting device determines that it cannot return a content object based on the interest name (decision **604**), the device generates a first content object which indicates a negative acknowledgement ("NACK") (operation **620**). The device then transmits the first content object (operation **622**).

Exemplary Computer Systems

[0083] FIG. **7** illustrates an exemplary computer system that facilitates efficient name encryption in a content centric network, in accordance with an embodiment of the present invention. Computer system **702** includes a processor **704**, a memory **706**, and a storage device **708**. Memory **706** can include a volatile memory (e.g., RAM) that serves as a managed memory, and can be used to store one or more memory pools. Furthermore, computer system **702** can be coupled to a display device **710**, a keyboard **712**, and a pointing device **714**. Storage device **708** can store an operating system **716**, a content-processing system **718**, and data **730**.

[0084] Content-processing system **718** can include instructions, which when executed by computer system **702**, can cause computer system **702** to perform methods and/or processes described in this disclosure. Specifically, content-processing system **718** may include instructions for sending and/or receiving data packets to/from other network nodes across a computer network, such as a content centric network (communication module **720**). A data packet can include an interest packet or a content object packet with a name which is an HSVLI that includes contiguous name components ordered from a most general level to a most specific level, and the name can include a random nonce appended to the end of the name (e.g., as a last name component).

[0085] Further, content-processing system **718** can include instructions for determining an index for a name of an interest, wherein the index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest (index-determining module **722**). Content-processing system **718** can include instructions for encrypting one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix (name-encrypting module **724**). Content-processing system **718** can include instructions for transmitting the interest based on the encrypted name (communication module **720**).

[0086] Content-processing system **718** can also include instructions for generating a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level (interest-generating module **726**). Content-processing system **718** can also include instructions for, in response to receiving a first content object which indicates a positive response of the first probing interest (communication module **720**), setting the index to a number of name components in the first probing interest name (index-determining module **722**). Content-processing system **718** can also include instructions for, in response to receiving a second content object which indicates a negative acknowledgment of the first probing interest (communication module **720**), generating a second probing interest with a name that comprises the first probing interest name followed by a next contiguous name component of the interest name (interest-generating module **726**).

[0087] Content-processing system **718** can also include instructions for determining a midpoint index of a number of name components in the interest name (index-determining module **722**). Content-processing system **718** can also include instructions for generating a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to the name component corresponding to the midpoint index (interest-generating module **726**). Content-processing system **718** can also include instructions for, in response to receiving a first content object which indicates a positive response of the first probing interest (communication module **720**), setting the index to a number of name components in the first probing interest name (index-determining module **722**).

[0088] Content-processing system **718** can also include instructions for determining a lower midpoint index of a lower portion of the interest name (index-determining module **722**). Content-processing system **718** can also include instructions for generating a second probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the lower midpoint index. Content-processing system **718** can also include instructions for, in response to receiving a third content object which indicates a negative acknowledgment of the first probing interest (communication module **728**), determining an upper midpoint index of the upper portion (index-determining module **722**) and generating a third probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the upper midpoint index (interest-generating module **726**).

[0089] Content-processing system **718** can also include instructions for generating an initial interest for the index, wherein the initial interest is transmitted to a third party service and has a payload that includes the interest name and a public key of the third party service (interest-generating module **726**). Content-processing system **718** can also include instructions for, in response to the initial interest, receiving an initial content object that has a payload that indicates the index, wherein the payload of the initial content object is encrypted based on the public key of the third party service (communication module **720**).

[0090] Content-processing system **718** can also include instructions for receiving a first interest with a name that is an HSVLI, wherein a random nonce is appended to the first interest name (communication module **720**). Content-processing system **718** can include instructions for, in response to determining that the system can return a content object based on the first interest name (packet-processing module **730**), generating a first content object which indicates a positive response (content-generating module **728**). Content-processing system **718** can also include instructions for, in response to determining that the content-hosting device cannot return a content object based on the first interest name (packet-processing module **730**), generating a second content object which indicates a negative acknowledgment of the first interest (content-generating module **728**).

[0091] Data **732** can include any data that is required as input or that is generated as output by the methods and/or processes described in this disclosure. Specifically, data **732** can store at least: an interest or a content object packet; a name; a name that is an HSVLI that includes contiguous name components ordered from a most general level to a most specific level; an index that corresponds to a position of a name component in the HSVLI; an index which is a split index that indicates a minimum routable prefix; a routable

prefix; one or more encrypted name components; a probing interest with a random nonce appended as a last name component; an indicator of a positive response; an indicator of a negative response or acknowledgment ("NACK"); a key identifier of a content object; a public key or associated key identifier of a consumer, a third party service, or a content-hosting or content-producing device; a midpoint index which is an index corresponding to a midpoint of a total number of name components in an interest name; a lower portion of an interest name which includes name components from the most general level name component to the name component preceding the name component corresponding to the midpoint index; an upper portion of an interest name which includes name components from the name component following the name component corresponding to the midpoint index to the most specific level name component; a lower midpoint index of the lower portion; an upper midpoint index of the upper portion; and an indicator of a number of collapsed name prefixes in a forwarding information base.

[0092] The data structures and code described in this detailed description are typically stored on a computer-readable storage medium, which may be any device or medium that can store code and/or data for use by a computer system.

[0093] The computer-readable storage medium includes, but is not limited to, volatile memory, non-volatile memory, magnetic and optical storage devices such as disk drives, magnetic tape, CDs (compact discs), DVDs (digital versatile discs or digital video discs), or other media capable of storing computer-readable media now known or later developed.

[0094] The methods and processes described in the detailed description section can be embodied as code and/or data, which can be stored in a computer-readable storage medium as described above. When a computer system reads and executes the code and/or data stored on the computer-readable storage medium, the computer system performs the methods and processes embodied as data structures and code and stored within the computer-readable storage medium.

[0095] Furthermore, the methods and processes described above can be included in hardware modules. For example, the hardware modules can include, but are not limited to, application-specific integrated circuit (ASIC) chips, field-programmable gate arrays (FPGAs), and other programmable-logic devices now known or later developed. When the hardware modules are activated, the hardware modules perform the methods and processes included within the hardware modules.

[0096] The foregoing descriptions of embodiments of the present invention have been presented for purposes of illustration and description only. They are not intended to be exhaustive or to limit the present invention to the forms disclosed. Accordingly, many modifications and variations will be apparent to practitioners skilled in the art. Additionally, the above disclosure is not intended to limit the present invention. The scope of the present invention is defined by the appended claims.

What is claimed is:

1. A computer system for facilitating efficient name encryption, the system comprising:

a processor; and

a storage device storing instructions that when executed by the processor cause the processor to perform a method, the method comprising:

determining, by a client computing device, an index for a name of an interest, wherein the name is a hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level, wherein the index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest;

encrypting one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix; and

transmitting the interest based on the encrypted name, thereby facilitating efficient name encryption in a content centric network.

2. The computer system of claim **1**, wherein determining the index comprises:

generating a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level;

in response to receiving a first content object which indicates a positive response of the first probing interest, setting the index to a number of name components in the first probing interest name; and

in response to receiving a second content object which indicates a negative acknowledgment of the first probing interest, generating a second probing interest with a name that comprises the first probing interest name followed by a next contiguous name component of the interest name.

3. The computer system of claim **2**, wherein the method further comprises determining that the first content object indicates a positive response of the first probing interest, which comprises:

determining that the first content object indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest name; and

determining that a key identifier of the first content object matches a public key of the content producing device.

4. The computer system of claim **2**, wherein the method further comprises:

appending a first random nonce to the first probing interest name; and

appending a second random nonce to the second probing interest name.

5. The computer system of claim **1**, wherein determining the index further comprises:

determining a midpoint index of a number of name components in the interest name, wherein a lower portion of the interest name includes the name components from the most general level name component to the name component preceding the name component corresponding to the midpoint index, and wherein an upper portion of the interest name includes the name components from the name component following the name component corresponding to the midpoint index to the most specific level name component;

generating a first probing interest with a name that comprises one or more contiguous name components of the

interest name beginning from the most general level name component to the name component corresponding to the midpoint index; and

in response to receiving a first content object which indicates a positive response of the first probing interest, setting the index to a number of name components in the first probing interest name.

6. The computer system of claim **5**, wherein determining the index further comprises:

in response to receiving a second content object which indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest, and in response to determining that a key identifier of the second content object is associated with a public key of the content producing device:

determining a lower midpoint index of the lower portion; and

generating a second probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the lower midpoint index; and

in response to receiving a third content object which indicates a negative acknowledgment of the first probing interest:

determining an upper midpoint index of the upper portion; and

generating a third probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the upper midpoint index.

7. The computer system of claim **5**, wherein the method further comprises:

appending a first random nonce to the first probing interest name;

appending a second random nonce to the second probing interest name; and;

appending a third random nonce to the third probing interest name.

8. The computer system of claim **1**, wherein determining the index further comprises generating one or more probing interests based on a number of number components for the interest name and further based on one or more of:

a linear search;

a binary search; and

a number of collapsed name prefixes in a forwarding information base, wherein a collapsed name prefix indicates a plurality of name components with a same forwarding information in the forwarding information base.

9. The computer system of claim **1**, wherein determining the index further comprises:

generating an initial interest for the index, wherein the initial interest is transmitted to a third party service and has a payload that includes the interest name and a public key of the third party service, wherein the payload of the initial interest is encrypted based on a public key of the client computing device, wherein the initial interest indicates the public key of the client computing device; and

in response to the initial interest, receiving an initial content object that has a payload that indicates the index, wherein the payload of the initial content object is encrypted based on the public key of the third party service.

10. The computer system of claim **1**, wherein the interest name includes one or more nested and encrypted names suffixes, wherein a name suffix comprises one or more contiguous name components of the interest name, and wherein the method further comprises:

determining a second index for a nested and encrypted name suffix, wherein the second index indicates a minimum number of contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy a nested interest with a name which includes the nested and encrypted name suffix; and

encrypting the name components following the name components corresponding to the second index.

11. A computer-implemented method for facilitating efficient name encryption, the method comprising:

determining, by a client computing device, an index for a name of an interest, wherein the name is a hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level, wherein the index indicates a minimum number of the contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy the interest;

encrypting one or more name components of the interest name beginning with the name component immediately following the minimum routable prefix; and

transmitting the interest based on the encrypted name, thereby facilitating efficient name encryption in a content centric network.

12. The method of claim **11**, wherein determining the index comprises:

generating a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level;

in response to receiving a first content object which indicates a positive response of the first probing interest, setting the index to a number of name components in the first probing interest name;

in response to receiving a second content object which indicates a negative acknowledgment of the first probing interest, generating a second probing interest with a name that comprises the first probing interest name followed by a next contiguous name component of the interest name;

appending a first random nonce to the first probing interest name; and

appending a second random nonce to the second probing interest name.

13. The method of claim **12**, further comprising determining that the first content object indicates a positive response of the first probing interest, which comprises:

determining that the first content object indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest name; and

determining that a key identifier of the first content object matches a public key of the content producing device.

14. The method of claim **1**, wherein determining the index further comprises:

determining a midpoint index of a number of name components in the interest name, wherein a lower portion of the interest name includes the name components from the most general level name component to the name component preceding the name component corresponding to the midpoint index, and wherein an upper portion of the interest name includes the name components from the name component following the name component corresponding to the midpoint index to the most specific level name component;

generating a first probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to the name component corresponding to the midpoint index; and

in response to receiving a first content object which indicates a positive response of the first probing interest, setting the index to a number of name components in the first probing interest name.

15. The method of claim **14**, wherein determining the index further comprises:

in response to receiving a second content object which indicates that a receiving content producing device can return a content object based on the name components of the interest name as included in the first probing interest, and in response to determining that a key identifier of the second content object is associated with a public key of the content producing device:

determining a lower midpoint index of the lower portion; and

generating a second probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the lower midpoint index;

in response to receiving a third content object which indicates a negative acknowledgment of the first probing interest:

determining an upper midpoint index of the upper portion; and

generating a third probing interest with a name that comprises one or more contiguous name components of the interest name beginning from the most general level name component to a name component corresponding to the upper midpoint index;

appending a first random nonce to the first probing interest name;

appending a second random nonce to the second probing interest name; and;

appending a third random nonce to the third probing interest name.

16. The method of claim **11**, wherein determining the index further comprises generating one or more probing interests based on a number of number components for the interest name and further based on one or more of:

a linear search;

a binary search; and

a number of collapsed name prefixes in a forwarding information base, wherein a collapsed name prefix indicates a plurality of name components with a same forwarding information in the forwarding information base.

17. The method of claim **11**, wherein determining the index further comprises:

generating an initial interest for the index, wherein the initial interest is transmitted to a third party service and has a payload that includes the interest name and a public key of the third party service, wherein the payload of the initial interest is encrypted based on a public key of the client computing device, wherein the initial interest indicates the public key of the client computing device; and

in response to the initial interest, receiving an initial content object that has a payload that indicates the index, wherein the payload of the initial content object is encrypted based on the public key of the third party service.

18. The method of claim **11**, wherein the interest name includes one or more nested and encrypted names suffixes, wherein a name suffix comprises one or more contiguous name components of the interest name, and wherein the method further comprises:

determining a second index for a nested and encrypted name suffix, wherein the second index indicates a minimum number of contiguous name components beginning from the most general level that represent a minimum routable prefix needed to route the interest to a content producing device that can satisfy a nested interest with a name which includes the nested and encrypted name suffix; and

encrypting the name components following the name components corresponding to the second index.

19. A computer system for facilitating efficient content exchange, the system comprising:

a processor; and

a storage device storing instructions that when executed by the processor cause the processor to perform a method, the method comprising:

receiving, by a content-hosting device, a first interest with a name that is a hierarchically structured variable length identifier that includes contiguous name components ordered from a most general level to a most specific level, wherein a random nonce is appended to the first interest name;

in response to determining that the content-hosting device can return a content object based on the first interest name, generating a first content object which indicates a positive response; and

in response to determining that the content-hosting device cannot return a content object based on the first interest name, generating a second content object which indicates a negative acknowledgment of the first interest.

20. The computer system of claim **19**, wherein the method further comprises:

receiving a second interest with a partially encrypted name that includes one or more encrypted name components;

decrypting the partially encrypted name for the second interest;

determining a second content object corresponding to the decrypted name for the second interest;

replacing, in the second content object, the decrypted name for the second interest with the partially encrypted name for the second interest; and

transmitting the second content object.

* * * * *