



- (51) International Patent Classification: *H04L 12/721* (2013.01) *H04L 12/26* (2006.01)
New York 11801 (US). **BROCKNERS, Frank**; Gutenbergstrasse 26, 50823 Köln (DE).
- (21) International Application Number: PCT/US2019/052168
- (22) International Filing Date: 20 September 2019 (20.09.2019)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 16/138,650 21 September 2018 (21.09.2018) US
- (71) Applicant: **CISCO TECHNOLOGY, INC.** [US/US]; 170 West Tasman Drive, San Jose, California 95134 (US).
- (72) Inventors: **FILSFILS, Clarence**; 66 rue du Duc, 1150 Belgium (BE). **ALI, Zafar**; 422 S. Broadway, #5, Hicksville, (74) Agent: **MCKNIGHT, Brian**; POLSINELLI, THREE EMBARCADERO CENTER, SUITE 2400, SAN FRANCISCO, California 94111 (US).
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(54) Title: SCALABLE AND PROGRAMMABLE MECHANISM FOR TARGETED IN-SITU OAM IMPLEMENTATION IN SEGMENT ROUTING NETWORKS

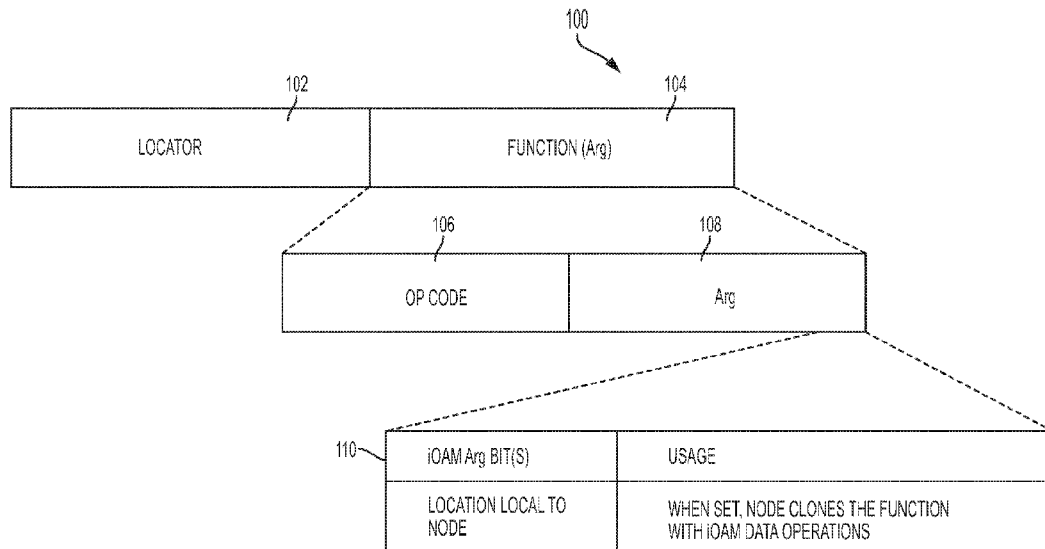


FIG. 1

(57) Abstract: The present technology provides a system and method for implementing targeted collection of in-situ Operation, Administration and Maintenance data from select nodes in a Segment Routing Domain. The selection is programmable and is implemented by setting an iOAM bit in the function arguments field of a Segment Identifier. In this way only the nodes associated with local Segment Identifiers (Function field of a Segment Identifier) with an iOAM argument bit are directed to generate iOAM data. The iOAM data generated by target nodes may be stored in TLV field of the segment routing header. The Segment Routing packet is then decapsulated at a Segment Routing egress node and the Header information with the collected iOAM data is sent to a controller entity for further processing, analysis and/or monitoring.



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

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

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

SCALABLE AND PROGRAMMABLE MECHANISM FOR TARGETED IN-SITU OAM IMPLEMENTATION IN SEGMENT ROUTING NETWORKS

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of, and priority to, U.S. Non-Provisional Patent Application No. 16/138,650, filed on September 21, 2018, entitled "SCALABLE AND PROGRAMMABLE MECHANISM FOR TARGETED IN-SITU OAM IMPLEMENTATION IN SEGMENT ROUTING NETWORKS," the content of which is hereby expressly incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] The present technology pertains to in-situ collection of operational and telemetry information in a data network. More specifically it is directed to a scalable implementation of in-situ Operation, Administration and Maintenance data probing and collection in a Segment Routing domain.

BACKGROUND

[0003] The ever increasing complexity and diversity of application traffic having a diverse range of performance requirements has meant that Service Level Agreement (SLA) may now require routine verification of services provided, for example, by providing a proof of transit showing that the traffic indeed follows the suggested application-specific path. Operation, Administration, and Maintenance (OAM) operations can be implemented to obtain telemetry for all user traffic. However, this approach does not provide real-time results and significantly increases the number of messages or traffic propagating in a network, which may already be highly loaded or suffering performance issues.

[0004] In-situ Operation, Administration, and Maintenance (iOAM) records operational and telemetry information in the packet while the packet traverses a path between two points in the network. One of the challenges associated with the "In-situ OAM" mechanisms is the difficulty to implement it in hardware without performance penalties. Enabling "In-situ OAM" probes on all transit nodes can potentially affect the timing of the very stream targeted for monitoring and

troubleshooting. Furthermore, the nodes need to have an indication somewhere in the packet header that hardware needs to check for each packet in the network, regardless of whether "In-situ OAM" operation is enabled or not, thus causing additional hardware performance penalties.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] In order to describe the manner in which the above-recited and other advantages and features of the disclosure can be obtained, a more particular description of the principles briefly described above will be rendered by reference to specific embodiments thereof which are illustrated in the appended drawings. Understanding that these drawings depict only exemplary embodiments of the disclosure and are not therefore to be considered to be limiting of its scope, the principles herein are described and explained with additional specificity and detail through the use of the accompanying drawings in which:

[0006] Fig. 1 illustrates a Segment Identifier Format encoding an in-situ OAM probe, in accordance to some embodiments of the present technology.

[0007] Fig. 2 illustrates two example segment routing headers, one with and the other without an in-situ OAM probe, in accordance to some embodiments of the present technology.

[0008] Fig. 3 illustrates a segment routing header with a targeted (node-specific) in-situ OAM probe included therein, in accordance to some embodiments of the present technology.

[0009] Fig. 4 illustrates targeted iOAM data collection from selected Segment Routing capable nodes, in accordance to some embodiments of the present technology.

[0010] Fig. 5 illustrates a segment routing header transporting in-situ OAM data in its Type, Length, Value (TLV) fields which is decapsulated by the egress Segment Routing node and sent to a controller entity, in accordance to some embodiments of the present technology.

[0011] Fig. 6 illustrates a flowchart for an example method of implementing targeted iOAM probing in Segment Routing context, in accordance to some embodiments of the present technology

[0012] Fig. 7 illustrates an example network device in accordance to some embodiments of the present technology.

[0013] Fig. 8 illustrates an example architecture of a computing device, in accordance to some embodiments of the present technology.

DESCRIPTION OF EXAMPLE EMBODIMENTS

[0014] Additional features and advantages of the disclosure will be set forth in the description which follows, and in part will be obvious from the description, or can be learned by practice of the herein disclosed principles. The features and advantages of the disclosure can be realized and obtained by means of the instruments and combinations particularly pointed out in the appended claims. These and other features of the disclosure will become more fully apparent from the following description and appended claims, or can be learned by the practice of the principles set forth herein.

OVERVIEW

[0015] Systems, methods and computer-readable storage media are disclosed for scalable and targeted collection of in-situ Operation, Administration, and Maintenance (iOAM) data in a programmable way in a Segment Routing context. In some examples, a method can involve encoding an iOAM instruction as a local argument in the function field of one or more Segment Identifiers (SID) selected from a listing of Segment Identifiers (SID list) specified in the segment routing header of a packet. The one or more SIDs in the SID list of the segment routing header, which feature an iOAM argument bit in their respective function fields, can correspond to one or more Segment Routing nodes selected for iOAM data collection. In some examples, this may be achieved by setting an iOAM bit in the function argument field of one or more Segment Identifier in the Segment Identifier list.

[0016] The method can further involve sending the packet to the one or more segment routing nodes based on the segment routing header, receiving a packet containing the iOAM data from the one or more Segment Routing nodes selected for iOAM data collection, and processing the iOAM data from the one or more Segment Routing nodes selected for iOAM data collection. According to some examples, the iOAM data from the one or more targeted Segment Routing nodes can be inserted into one or more Type, Length, Value (TLV) fields of the segment routing header of the packet. An egress Segment Routing node can extract the Segment Routing header,

which includes the collected iOAM data from the selected Segment Routing nodes, and send the information to a controller entity for further processing, analysis and/or monitoring. The egress segment routing node may forward the user data packet (e.g., a remaining portion of the Segment Routing encapsulated packet) towards its intended destination. Alternatively the one or more Segment Routing nodes selected for iOAM data collection may insert the generated iOAM data into a duplicate copy of the Segment Routing header. The duplicate copy with the iOAM information included therein is sent to a controller entity using a collector mechanism. The targeted/tapped Segment Routing nodes may then forward the Segment Routing packet with the header-embedded iOAM probes onto the next hop along the Segment Routing Path specified in the SID list.

[0017] In some examples, a Segment Routing ingress router that encapsulates the incoming packet with the segment routing header may be used to encode iOAM probe(s) in the function field (or the local SID) of one or more Segment Identifiers selected from the entries in the SID list of the segment routing header. In other examples, selecting target segment routing nodes for iOAM data collection may be performed by a Segment Routing Policy Headend router serving as a controller entity for both selective iOAM probing and iOAM data collection from probed Segment Routing nodes. The encapsulating Segment Routing ingress router and/or the controller entity may programmably change, for example in a round robin fashion, the one or more Segment Routing nodes selected from the SID list of the segment routing header for iOAM probing.

EXAMPLE EMBODIMENTS

[0018] Disclosed are systems, methods, and non-transitory computer-readable storage media for scalable programmable in-situ OAM implementation in a Segment Routing context. Various embodiments of the disclosure are discussed in detail below. While specific implementations are discussed, it should be understood that this is done for illustration purposes only. A person skilled in the relevant art will recognize that other components and configurations may be used without parting from the spirit and scope of the disclosure.

[0019] Operations, administration and management (OAM) refer to a set of processes, activities, tools and standards involved with operating, administering, managing and maintaining telecommunication systems and computer networks/hardware. It is often involved with fault

management and performance monitoring and may include measurements such as frame loss ratio, frame delay, frame delay variation, continuity checks to assist with Service Level Agreement (SLA) and capacity planning.

[0020] OAM functionality generally involves a set of network management tools and functions that provide network fault indication, fault localization, performance information, and data and diagnostic functions. These operations may involve automatic monitoring of environment, detecting and determining faults and alerting administrators, collecting performance statistics, accounting data for the purpose of billing, capacity planning based on usage data and maintaining system reliability. As such, OAM functionality enables effective fault detection, verification, isolation and notification in carrier networks.

[0021] In-situ OAM (iOAM) can provide real-time telemetry for individual data packets and flows. iOAM can include operational and telemetry data and metadata embedded within live user traffic (packets that originate and terminate at the application layer). In iOAM, operational information may be recorded in the packet as it traverses a path between two points in the network. As described herein, "In-situ OAM" can be implemented in a IPv6 Segment Routed (SRv6) network by carrying appropriate data fields in the Type Length Value (TLV) fields of a segment routing header (SRH). A bit may be defined in the segment routing header that when set enables in-situ OAM data collection.

[0022] The present technology describes methods, systems, for selective probing and collection of iOAM data from programmably selected target nodes in a scalable fashion. The present technology obviates the need to monitor a bit in each incoming packet, as the instruction to perform iOAM function is encoded in the argument field of the SRv6 SID function. This way only nodes for which the local SID has the iOAM argument set will insert the iOAM data. A controller entity (SR policy headend) or a SR encapsulating ingress node may programmably change the iOAM target nodes or iOAM tapping points in order to construct the entire picture or model of how data is traveling in the network, thus providing scalable and programmable in-situ OAM data collection.

[0023] Segment Routing (SR) allows a node to steer a packet through a controlled set of instructions, called segments, by prepending a segment routing header (SRH) to the packet. A segment can represent any (forwarding) instruction, topological or service-based. Segment

Routing allows for steering of a flow through any path (topological or service/application based) while maintaining per-flow state only at the ingress node of the SR domain. Segments can be derived from different components: IGP, BGP, Services, Contexts, Locators, etc. The list of segment defining an end-to-end forwarding path of the flow packets is called the Segment List and is encoded in the SRH of the packet.

[0024] In the IPv6 Segment Routing architecture, a Segment Identifier (SID) may be represented as an IPv6 address modeled as a Locator and a Function. The Locator, as represented by the most significant bits of the address, is used to route the packet to the associated segment (i.e., the node corresponding to the segment). The Function, as represented by the least significant bits of the address, may be used to identify the action to be performed by the segment (i.e., the node corresponding to the segment). Optionally, the function bits may include local arguments, which are encoded in the last bits of the address. The specific address format (i.e., number of bits allocated to each field) is entirely flexible as it may be defined locally by the parent node. SID reachability is made possible by advertising the locator prefix within the routing protocol domain.

[0025] Treatment of OAM operation as a SID function, as disclosed by some embodiments of the present technology, enables the implementation of a programmable in-situ OAM. Consequently, instead of only providing a global end-to-end behavior, service providers may control OAM features on a node-by-node basis, enabling specific OAM operations to be performed on selected node(s). The iOAM enabled Segment Identifier includes an iOAM argument bit(s) in the Function field of the Segment Identifier. This may include flipping a bit in the appropriate argument portion of the Segment Identifier's Function field.

[0026] Fig. 1 illustrates an example format for a Segment Identifier (SID) 100 carrying an iOAM instruction. The Segment Identifier 100 includes a routable Locator field 102. As stated above, the locator information is encoded by the first most significant bits of the Segment Identifier and represents an address of a particular Segment Routing node (parent node of the local SID) and it is therefore used for routing in a Segment Routing domain. The remaining SID bits constitute the Function field 104 which identifies the function that is executed locally on a particular node, specified by the locator bits. The Function field 104 further comprises a portion 106 for identifying the type of operation to be performed (Op Code) and a portion 108 for storing one or

more parameters/arguments that may be required for performing the operation identified by the Op Code (i.e., arguments passed to the function). Presence of iOAM argument bit(s) 110 in the argument portion 108 of the Function field 104 prompt the targeted Segment Routing node to take a specific action such as cloning the Function (104) with iOAM data operations.

[0027] Segment Routing deployments can be used to deliver customized services with stringent performance requirements, details of which may be explicitly set forth in a service level agreements (SLA). Ensuring that such service-level guarantees are met may require routine monitoring to verify that a forwarding path across the network is in compliance with the implemented Segment Routing policy and the provisions of the associated Service Level Agreement. To address this requirement, iOAM probing may be implemented to verify a particular Segment Routing policy by monitoring the live data as it is steered across the Segment Routing path. However, implementing this in hardware amount to enabling in-situ OAM probe on all transit nodes which may potentially affect the timing of the actual traffic stream being probed. Therefore some performance penalties may result from the performance measurement operation itself. Additionally, the aforementioned Hardware implemented iOAM probing scheme may further incur additional hardware performance penalties as the examination of header information is performed for all incoming packets, regardless of whether the in-situ OAM is enabled or not.

[0028] In the context of Segment Routing based IPv6 (SRv6) networks, iOAM data probing and collection may involve provisioning iOAM data-fields in the Type Length Value (TLV) field of the segment routing header. Moreover iOAM probing operation may involve setting a bit (i.e., O-flag) defined in the segment routing header, which when set indicates that iOAM data collection is enabled. This approach, however, is also susceptible to performance constraints discussed above.

[0029] Generally, in-situ OAM (iOAM) data collection is expected to be deployed in a specific domain rather than on the overall Internet. The part of the network which employs iOAM is referred to as the iOAM domain. In-situ OAM data is added to a packet upon entering the iOAM-domain and is removed from the packet when exiting the domain. Within the iOAM-domain, the iOAM data may be updated by network nodes that the packet traverses. The device which adds an iOAM data container to the packet to capture iOAM data is called the iOAM

encapsulating node whereas the device which removes the iOAM data container is referred to as the iOAM decapsulating node. Nodes within the domain which are aware of iOAM data and read and/or write or process the iOAM data are called iOAM transit nodes. Restricting the proliferation boundary of iOAM in this way serves to contain the iOAM signaling and data transport traffic along with the resulting processing load within the iOAM domain thus keeping it away from the rest of the network.

[0030] The present technology enables selective collection of iOAM data from target nodes in a programmable fashion. Embodiments of the present technology obviate the need to monitor "a" bit (iOAM flag) in each incoming packet, as the instruction to perform iOAM function is encoded in the argument field of the SRv6 SID function. This way only nodes for which the local SID has the iOAM argument set will insert iOAM data. According to some embodiments an SRv6 Policy headend/controller entity may programmably change the iOAM tapping points (devices/nodes selected for iOAM collection) to construct a comprehensive picture of how data is traveling in the network.

[0031] Some aspects involve a programmable iOAM implementation that enables a user/operator to specifically select the node from which to collect the desired iOAM data. The iOAM data may be injected into the header of the data packet by the specified node as it forwards the packet onto its next hop. Alternatively a duplicate copy of the packet with the iOAM information inserted therein may be sent to a device, such as a controller entity, using appropriate collector mechanism such as Netflow/IPFIX. An example implementation of programmable iOAM is illustrated in Fig. 2.

[0032] Fig. 2 illustrates an example Segment Routing path/policy 202 where traffic coming from a User Equipment (i.e., smart phone) denoted as Node 0 is steered into a Segment Routing policy at gNodeB (Node 1). The Segment Routing policy steers the traffic via User Plane Function 1 (Node 2) to the Traffic Engineering or service chaining node (Node 3) and finally terminates the flow at User Plane Function 2 (Node 4) which is the end point of the Segment Routing policy 202. The ingress Segment Routing Node (Node 1) encapsulates the incoming flow packet from the User Equipment (Node 0) into a SRv6 packet 203. The SRv6 packet 203 comprises an outer IPv6 header 204 which further contains the segment routing header 208. The original user data packet 210 is left unmodified as the payload. The Source Address (SA) of the packet is the

ingress node, the Destination Address (DA) is set as the first segment of the path, which corresponds to Node 2 in the example Segment Routing path 202.

[0033] With reference to notations included in the segment routing header in Fig. 2; Node SIDs are represented as an alphabetic letter followed by the node number. For example, Node 2, 3 and 4 correspond to SIDs A2, A3 and A4, respectively. Notation A behind the node number indicate that the node/router is Segment Routing capable. Notation B behind the node number indicate that the node is a classic IPv6 transit node.

[0034] The segment routing header (SRH) 208 includes a Segment Identifier (SID) list which corresponds to a list of segments that define the steering path of the packets (i.e., Segment Routing Path 202 in Fig. 1). The Segment List is encoded starting from the last segment of the path (i.e., the first element of the segment list (A4) corresponds to the last segment of the path (Node 4), the second element (A3::C34) contains the penultimate segment of the path (Node 3) and so on). The identifier C34 attached to A3 (Locator SID portion) specifies a function to be performed locally at Node 3. The function or operation denoted by C34 may include switching the packet onto a specific outgoing interface or adjacency link that connects to Node 4 (last segment of the path and the egress node of the Segment Routing policy domain 202). Therefore the SID list representing the Segment Routing path 202 is expressed in the segment routing header 208 as (A2::, A3::C34, A4::).

[0035] Segments Left (SL) parameter encoded in the segment routing header 208 represents a pointer of the active segment in the Segment List, and is decremented at each segment. Therefore, the encapsulating node (Node 1) sets the numerical value of the SL parameter, in the Segment Routing header 208, to 2. This identifies Node 2 as the next segment along the Segment Routing path. The O-bit in the segment routing header 208 represents an OAM flag which when set indicates that the present packet is an operations and management (OAM) packet.

[0036] According to embodiments of the present technology any nodes across the Segment Routing path may be tapped for iOAM data collection. The Segment Routing (SRv6) packet 212 in Fig. 2 corresponds to a scenario whereby a user/operator wishes to collect iOAM data, on the Segment Routing policy path 202, only at the point of flow transit through the User Plane Function 1 (UPF1) at Node 2. In order to enable iOAM data tapping only at User Plane Function 1(UPF1) from Node 2 (i.e., to collect iOAM data only from node 2), the ingress Segment

Routing Node (Node 1) modifies the Segment Identifier (SID) list in the segment routing header 216 to <A4::, A3::C34, A2(1)::>. The referenced portion 217 in the IPv6 header 218 indicates that SR capable Node 2 has iOAM argument bit enabled. The Segment Routing (SRv6) packet 212 is such that the iOAM argument bit is only enabled on Node 2 as denoted by its modified SID Function notations A2(1). A2(1) is a clone of A2::SID with iOAM data collection bit enabled via a bit in the argument field of the A2::SID function. Such a probe as illustrated by 212 will collect iOAM data only from Node 2.

[0037] An example for enabling iOAM data tapping only at the Traffic Engineering or service Node 3 is illustrated in Fig. 3. The example corresponds to the same Segment Routing Path 202 as Fig. 2. The encapsulated SRv6 packet 303 in Fig. 3 corresponds to a scenario whereby a user/operator wishes to collect iOAM data on the Segment Routing policy path 202 only from the Traffic Engineering or Service Node 3. In order to enable iOAM data tapping only at Node 3 (i.e., to collect iOAM data only from Node 3) the ingress Segment Routing Node (Node 1) modifies the Segment Identifier (SID) list in the segment routing header 308 to (A2::, A3::C34(1), A4::). The referenced portion 317 of the segment routing header 308 indicates that SR capable Node 3 has iOAM argument bit enabled. The Segment Routing (SRv6) packet 303 is such that the iOAM argument bit is only enabled on Node 3 as denoted by its modified SID Function notations A3::C34(1). The modified/augmented SID of Node 3, A3::C34(1), which includes an iOAM probe in the encoded SID is a clone of the regular SID of Node 3, A3::C34, with the only difference being that in the former case iOAM data collection is enabled via a bit in the argument field of the A3::C34 SID function. Such a probe as illustrated by 303 will collect iOAM data only from Node 3.

[0038] According to some embodiment of the present technology, the modified SID with local iOAM probe functionality results from insertion of iOAM probe in the argument field of the locally significant portion (i.e. SID Function field) of a Segment Identifier. In some embodiments, the ingress Segment Routing node may implement iOAM data collection from multiple nodes by performing a round robin targeting of the nodes across the SID list to collect data in a scalable fashion. A controller entity may then run analytics routine and operations on the iOAM partial data to build a holistic view. The procedure is applicable for all underlay and overlay SRv6 SID types.

[0039] Due to the programmable nature of iOAM SID, as described by some embodiments, iOAM data collection may also be specified and implemented based on a local decision at a node. Specifically, an iOAM SID may implement iOAM data transport using "forward and punt" technique used by Netflow collector. In this case a copy of the packet is exported from the "tapping" node (iOAM collection node) to a controller entity with the requested iOAM information inserted therein. This case is depicted in Fig. 4.

[0040] With reference to Fig. 4, at 402 some of the User generated packets sourced from User Equipment (Node 1) are marked for insertion of iOAM probe. The iOAM augmentation of the SID of the marked packets is executed at gNodeB (Node 2) where the user data packet 401 is encapsulated with segment routing header 404. The O-bit in the Segment routing Header 404 is set to 1 and the hop limit (HL) is set to 64. As such, Node 2 implements a forward and punt mechanism. At each segment routing node the enabled O-bit (O-bit = 1) of the segment routing header causes a time-stamped copy of the packet to be punted (405) and processed elsewhere.

[0041] As described above, a segment routing header (SRH) can be used to steer packets through paths with given properties (e.g., bandwidth or latency) and through various network functions (e.g., firewalling). The list of segments present in the segment routing header thus specifies the network policy that applies to the packet. Each segment routing header contains at least a list of segments and a Segments Left pointer that references the current active segment (a value of 0 refers to the last segment). In addition, an segment routing header can optionally include extended information encoded as Type-Length-Value (TLV) fields. Another use of TLV is as a global argument field for passing additional information between locally executed SID Functions. An iOAM augmented SID may implement iOAM data transport inside of the data packet by using the TLV fields of the segment routing header. As such iOAM data records may be transported in the respective Type Length Value (TLV) data fields of the segment routing header until the flow is terminated at a Segment Routing egress router. The egress router will decapsulated the user data and send the segment routing header including the iOAM data inserted into TLV fields to a controller entity. This case is illustrated in Fig. 5.

[0042] Fig. 5 illustrates a Segment Routing packet 502 with iOAM probes enabled for the selected Segment Routing Nodes. The iOAM data generated at each iOAM tapping point (i.e., Segment Routing nodes with an iOAM argument bit enabled in the function filed of the

respective Segment Identifier) is collected in TLV fields 503 of the segment routing header 504. Node 4 is the egress node of the Segment Routing path 202. Therefore, Node 4 will decapsulated the Segment Routing packet 502 and export the segment routing header 504 with the collected iOAM data in the corresponding TLV fields to a controller entity, using, for example a Netflow collector. The User data packet 505 is then transported to the intended destination.

[0043] For clarity of explanation, in some instances the present technology may be presented as including individual functional blocks including functional blocks comprising devices, device components, steps or routines in a method embodied in software, or combinations of hardware and software.

[0044] An example flow chart 100, as presented in Fig. 6. illustrates targeted iOAM probing in Segment Routing context in accordance to some embodiments of the present technology. With reference to Fig. 6, at step 602 one or more Segment Routing (SR) nodes for iOAM data tapping to targeted iOAM data collection are identified. The iOAM instruction is encoded as a local argument in the function field of one or more Segment Identifiers (SIDs) corresponding to the one or more selected SR nodes at step 604.

[0045] Each SR capable node maintains a “My Local SID Table”. The table contains all the local segments explicitly instantiated at the node. Each entry of the “My Local SID Table” indicates the function associated with the local SID. As the SR packet travels the network, the Locator and Function are copied by each SR node to the destination address field of the SR header. When the SID inside of the SR header matches Local SID table of the SR capable node, the node executes a function encoded in the right part (Function field) of the SID. The next SID is placed into the SR Header destination field and the Segment Left Value is decreased by 1 accordingly.

[0046] Referring back to the Flowchart 100 in Fig. 6, at step 606, Destination address field of the SR header is compared with the SID of the SR Node the packet is forwarded to. In accordance to some embodiments of the technology, if there is a match (Locator portion of the destination SID matches the Locator portion of the Node SID and the function field of the destination SID is carrying an iOAM probe as argument), the operation moves to step 608 whereby the iOAM function is executed and the iOAM data is inserted inside TLV fields of the SR Header. If, however, the result of the comparison at step 606 returns no match, the packet is forwarded to

the next hop along the SR path and step 606 is repeated until a match is encountered, at which point the operation moves to step 608 as explained above.

[0047] At step 610 the operation verifies whether the current SR node is the SR Egress node. If the SR node does not correspond to the SR Egress node, the operation moves back to step 607 and the packet is forwarded to the next hop along the SR path and the process is repeated. However, if the comparison at step 610 reveals that the current SR node is the SR Egress router, the operation moves to 612 whereby the egress router decapsulates the header information and sends the header information along with the iOAM data embedded therein to a controller entity for further analysis and/or monitoring.

[0048] According to some embodiments of the present invention, after verifying a match at step 606, a duplicate copy of the packet including the requested iOAM data is generated by the SR node and sent to the controller entity for further analysis and/or monitoring. The operation is then moved to step 607, whereby the original packet is forwarded to the next hop along the SR path.

[0049] The disclosure now turns to FIGs. 7 and 8, which illustrate example architectures of computing and network devices, such as client computers, switches, routers, controllers, servers, and so forth.

[0050] FIG. 7 illustrates a computing system architecture 900 including components in electrical communication with each other using a connection 905, such as a bus. System 900 includes a processing unit (CPU or processor) 910 and a system connection 905 that couples various system components including the system memory 915, such as read only memory (ROM) 920 and random access memory (RAM) 925, to the processor 910. The system 900 can include a cache of high-speed memory connected directly with, in close proximity to, or integrated as part of the processor 910. The system 900 can copy data from the memory 915 and/or the storage device 930 to the cache 912 for quick access by the processor 910. In this way, the cache can provide a performance boost that avoids processor 910 delays while waiting for data. These and other modules can control or be configured to control the processor 910 to perform various actions. Other system memory 915 may be available for use as well. The memory 915 can include multiple different types of memory with different performance characteristics. The processor 910 can include any general purpose processor and a hardware or software service, such as service 1 932, service 2 934, and service 3 936 stored in storage device 930, configured to control the

processor 910 as well as a special-purpose processor where software instructions are incorporated into the actual processor design. The processor 910 may be a completely self-contained computing system, containing multiple cores or processors, a bus, memory controller, cache, etc. A multi-core processor may be symmetric or asymmetric.

[0051] To enable user interaction with the computing device 900, an input device 945 can represent any number of input mechanisms, such as a microphone for speech, a touch-sensitive screen for gesture or graphical input, keyboard, mouse, motion input, speech and so forth. An output device 935 can also be one or more of a number of output mechanisms known to those of skill in the art. In some instances, multimodal systems can enable a user to provide multiple types of input to communicate with the computing device 900. The communications interface 940 can generally govern and manage the user input and system output. There is no restriction on operating on any particular hardware arrangement and therefore the basic features here may easily be substituted for improved hardware or firmware arrangements as they are developed.

[0052] Storage device 930 is a non-volatile memory and can be a hard disk or other types of computer readable media which can store data that are accessible by a computer, such as magnetic cassettes, flash memory cards, solid state memory devices, digital versatile disks, cartridges, random access memories (RAMs) 925, read only memory (ROM) 920, and hybrids thereof.

[0053] The storage device 930 can include services 932, 934, 936 for controlling the processor 910. Other hardware or software modules are contemplated. The storage device 930 can be connected to the system connection 905. In one aspect, a hardware module that performs a particular function can include the software component stored in a computer-readable medium in connection with the necessary hardware components, such as the processor 910, connection 905, output device 935, and so forth, to carry out the function.

[0054] FIG. 8 illustrates an example network device 1000 suitable for performing switching, routing, assurance, and other networking operations. Network device 1000 includes a central processing unit (CPU) 1004, interfaces 1002, and a connection 1010 (*e.g.*, a PCI bus). When acting under the control of appropriate software or firmware, the CPU 1004 is responsible for executing packet management, error detection, and/or routing functions. The CPU 1004 preferably accomplishes all these functions under the control of software including an operating

system and any appropriate applications software. CPU 1004 may include one or more processors 1008, such as a processor from the INTEL X106 family of microprocessors. In some cases, processor 1008 can be specially designed hardware for controlling the operations of network device 1000. In some cases, a memory 1006 (e.g., non-volatile RAM, ROM, TCAM, etc.) also forms part of CPU 1004. However, there are many different ways in which memory could be coupled to the system. In some cases, the network device 1000 can include a memory and/or storage hardware, such as TCAM, separate from CPU 1004. Such memory and/or storage hardware can be coupled with the network device 1000 and its components via, for example, connection 1010.

[0055] The interfaces 1002 are typically provided as modular interface cards (sometimes referred to as "line cards"). Generally, they control the sending and receiving of data packets over the network and sometimes support other peripherals used with the network device 1000. Among the interfaces that may be provided are Ethernet interfaces, frame relay interfaces, cable interfaces, DSL interfaces, token ring interfaces, and the like. In addition, various very high-speed interfaces may be provided such as fast token ring interfaces, wireless interfaces, Ethernet interfaces, Gigabit Ethernet interfaces, ATM interfaces, HSSI interfaces, POS interfaces, FDDI interfaces, WIFI interfaces, 3G/4G/5G cellular interfaces, CAN BUS, LoRA, and the like. Generally, these interfaces may include ports appropriate for communication with the appropriate media. In some cases, they may also include an independent processor and, in some instances, volatile RAM. The independent processors may control such communications intensive tasks as packet switching, media control, signal processing, crypto processing, and management. By providing separate processors for the communications intensive tasks, these interfaces allow the master microprocessor 1004 to efficiently perform routing computations, network diagnostics, security functions, etc.

[0056] Although the system shown in FIG. 8 is one specific network device of the present disclosure, it is by no means the only network device architecture on which the concepts herein can be implemented. For example, an architecture having a single processor that handles communications as well as routing computations, etc., can be used. Further, other types of interfaces and media could also be used with the network device 1000.

[0057] Regardless of the network device's configuration, it may employ one or more memories or memory modules (including memory 1006) configured to store program instructions for the general-purpose network operations and mechanisms for roaming, route optimization and routing functions described herein. The program instructions may control the operation of an operating system and/or one or more applications, for example. The memory or memories may also be configured to store tables such as mobility binding, registration, and association tables, etc. Memory 1006 could also hold various software containers and virtualized execution environments and data.

[0058] The network device 1000 can also include an application-specific integrated circuit (ASIC), which can be configured to perform routing, switching, and/or other operations. The ASIC can communicate with other components in the network device 1000 via the connection 1010, to exchange data and signals and coordinate various types of operations by the network device 1000, such as routing, switching, and/or data storage operations, for example.

[0059] In some embodiments the computer-readable storage devices, mediums, and memories can include a cable or wireless signal containing a bit stream and the like. However, when mentioned, non-transitory computer-readable storage media expressly exclude media such as energy, carrier signals, electromagnetic waves, and signals per se.

[0060] Methods according to the above-described examples can be implemented using computer-executable instructions that are stored or otherwise available from computer readable media. Such instructions can comprise, for example, instructions and data which cause or otherwise configure a general purpose computer, special purpose computer, or special purpose processing device to perform a certain function or group of functions. Portions of computer resources used can be accessible over a network. The computer executable instructions may be, for example, binaries, intermediate format instructions such as assembly language, firmware, or source code. Examples of computer-readable media that may be used to store instructions, information used, and/or information created during methods according to described examples include magnetic or optical disks, flash memory, USB devices provided with non-volatile memory, networked storage devices, and so on.

[0061] Devices implementing methods according to these disclosures can comprise hardware, firmware and/or software, and can take any of a variety of form factors. Typical examples of

such form factors include laptops, smart phones, small form factor personal computers, personal digital assistants, and so on. Functionality described herein also can be embodied in peripherals or add-in cards. Such functionality can also be implemented on a circuit board among different chips or different processes executing in a single device, by way of further example.

[0062] The instructions, media for conveying such instructions, computing resources for executing them, and other structures for supporting such computing resources are means for providing the functions described in these disclosures.

[0063] Although a variety of examples and other information was used to explain aspects within the scope of the appended claims, no limitation of the claims should be implied based on particular features or arrangements in such examples, as one of ordinary skill would be able to use these examples to derive a wide variety of implementations. Further and although some subject matter may have been described in language specific to examples of structural features and/or method steps, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to these described features or acts. For example, such functionality can be distributed differently or performed in components other than those identified herein. Rather, the described features and steps are disclosed as examples of components of systems and methods within the scope of the appended claims.

CLAIMS

1. A method comprising:

encoding an in-situ Operation, Administration, and Maintenance (iOAM) instruction as a local argument in a function field of one or more segment identifiers selected from a plurality of segment identifiers in a segment identifier list of a segment routing header in a first packet, wherein the one or more segment identifiers correspond to one or more segment routing nodes selected for iOAM data collection;

sending the first packet to the one or more segment routing nodes based on the segment routing header;

receiving a second packet containing the iOAM data from the one or more segment routing nodes selected for iOAM data collection; and

processing the iOAM data from the one or more segment routing nodes selected for iOAM data collection.

2. The method of claim 1, wherein the encoding is implemented by setting one or more argument bits in the function field of the one or more segment identifiers.

3. The method of claim 1 or 2, wherein the iOAM data provided by the one or more Segment Routing nodes is inserted into one or more Type, Length, Value (TLV) fields of the segment routing header.

4. The method of claim 3, wherein the iOAM data is decapsulated by a Segment Routing egress node and sent to a controller entity.

5. The method of any preceding claim, wherein each of the one or more Segment Routing nodes selected for iOAM data collection sends a copy of the segment routing header containing iOAM data to a controller using a collector mechanism.

6. The method of claim 5, wherein the collector mechanism corresponds to a Netflow or IPFIX collector mechanism.

7. The method of any preceding claim, further comprising: varying, using a controller entity, the one or more segment routing nodes selected for iOAM data collection to optimize iOAM data collection without increasing iOAM traffic.

8. The method of claim 7, wherein the controller entity is an Segment Routing Policy Headend router.

9. At least one non-transitory computer-readable storage medium comprising instructions stored therein, which when executed by one or more processors, cause the processors to:

encode an in-situ Operation, Administration, and Maintenance (iOAM) instruction as a local argument in a function field of one or more segment identifiers selected from a plurality of segment identifiers in a segment identifier list of a segment routing header in a first packet, wherein the one or more segment identifiers correspond to one or more segment routing nodes selected for iOAM data collection;

send the first packet to the one or more segment routing nodes based on the segment routing header;

receive a second packet containing the iOAM data from the one or more segment routing nodes selected for iOAM data collection; and

process the iOAM data provided by the one or more segment routing nodes tapped for iOAM data collection.

10. The at least one non-transitory computer-readable medium of claim 9, wherein the encoding is implemented by setting one or more argument bits in the function field of the one or more segment identifiers.

11. The at least one non-transitory computer-readable medium of claim 9 or 10, wherein the iOAM data provided by the one or more Segment Routing nodes is inserted into one or more Type, Length, Value (TLV) fields of the segment routing header.

12. The at least one non-transitory computer-readable medium of claim 11, wherein the iOAM data is decapsulated by a Segment Routing egress node and sent to a controller entity.

13. The at least one non-transitory computer-readable medium of any of claims 9 to 11, wherein each of the one or more Segment Routing nodes selected for iOAM data collection sends a copy of the segment routing header containing iOAM data to a controller using a collector mechanism.

14. The at least one non-transitory computer-readable medium of claim 13, wherein the collector mechanism corresponds to a Netflow or IPFIX collector mechanism.

15. The at least one non-transitory computer-readable medium of any of claims 9 to 14, further comprising: varying, using a controller entity, the one or more segment routing nodes selected for iOAM data collection to thereby optimize iOAM data collection without increasing iOAM traffic.

16. The at least one non-transitory computer-readable medium of claim 15, wherein the controller entity is an Segment Routing Policy Headend router.

17. A system comprising:

one or more processors; and

at least one computer-readable storage medium having stored therein instructions which, when executed by the one or more processors, cause the system to:

insert an in-situ Operation, Administration, and Maintenance (iOAM) probe in a function field of one or more segment identifiers selected from plurality of segment identifiers in a segment identifier list of a segment routing header in a first packet, wherein the one or more segment identifiers correspond to one or more segment routing nodes selected for iOAM data collection, wherein the segment identifier list represents a segment routing policy path;

send the first packet to the one or more segment routing nodes based on the segment routing header;

receive a second packet containing the iOAM data from the one or more segment routing nodes selected for iOAM data collection; and

process the iOAM data in the second packet from the one or more segment routing nodes selected for iOAM data collection.

18. The system of claim 17, wherein the in-situ OAM probe is inserted in an argument field in the function field of the one or more segment identifiers.

19. The system of claim 17 or 18, wherein the iOAM data provided by the one or more Segment Routing nodes is inserted into one or more Type, Length, Value (TLV) fields of a second segment routing header associated with the second packet.

20. The system of any of claims 17 to 19, wherein the at least one computer-readable storage medium stores additional instructions which, when executed by the one or more processors, cause the system to:

decapsulate the segment routing header to yield decapsulated iOAM data; and
send the decapsulated iOAM data to a collector entity.

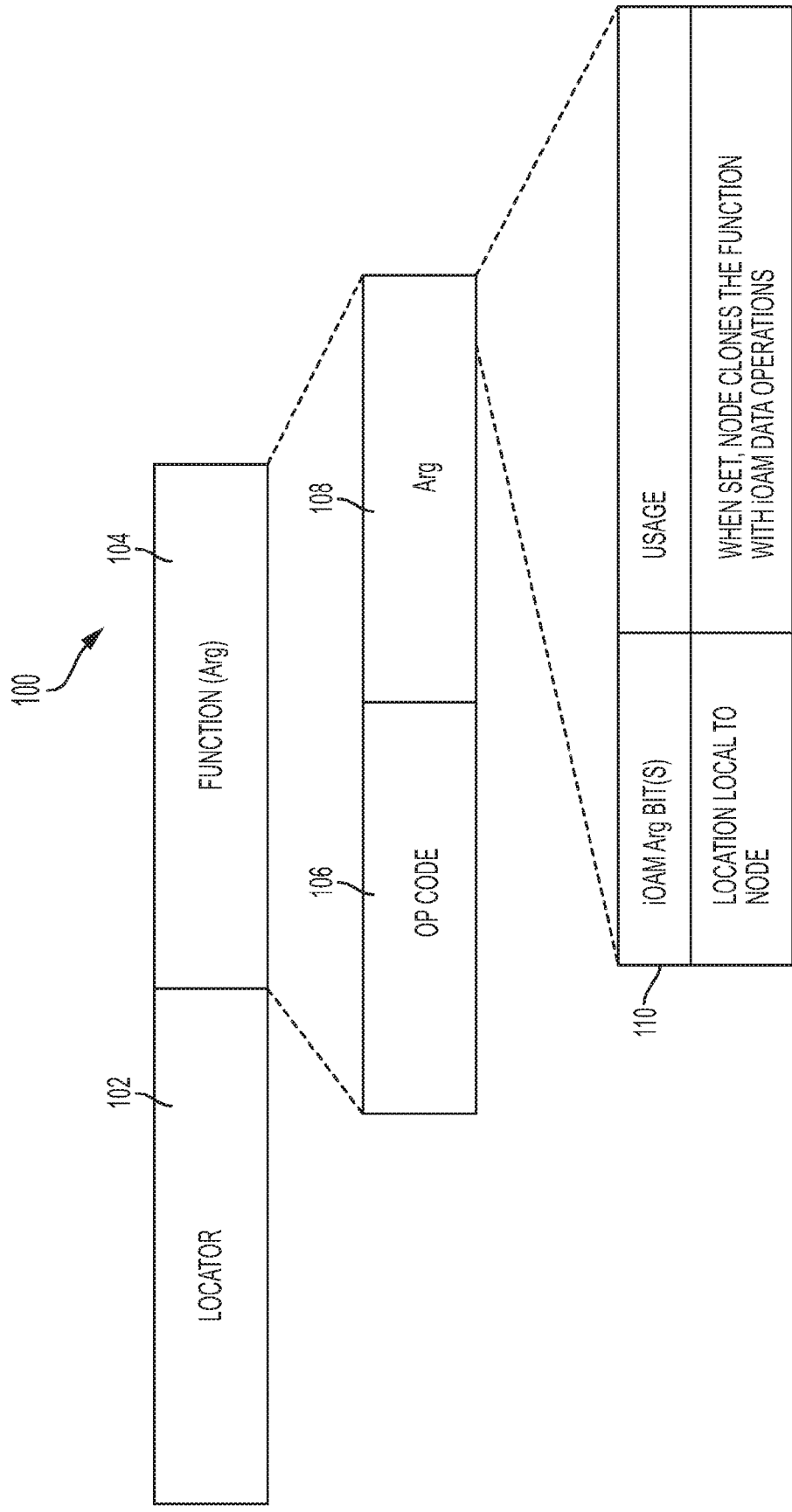


FIG. 1

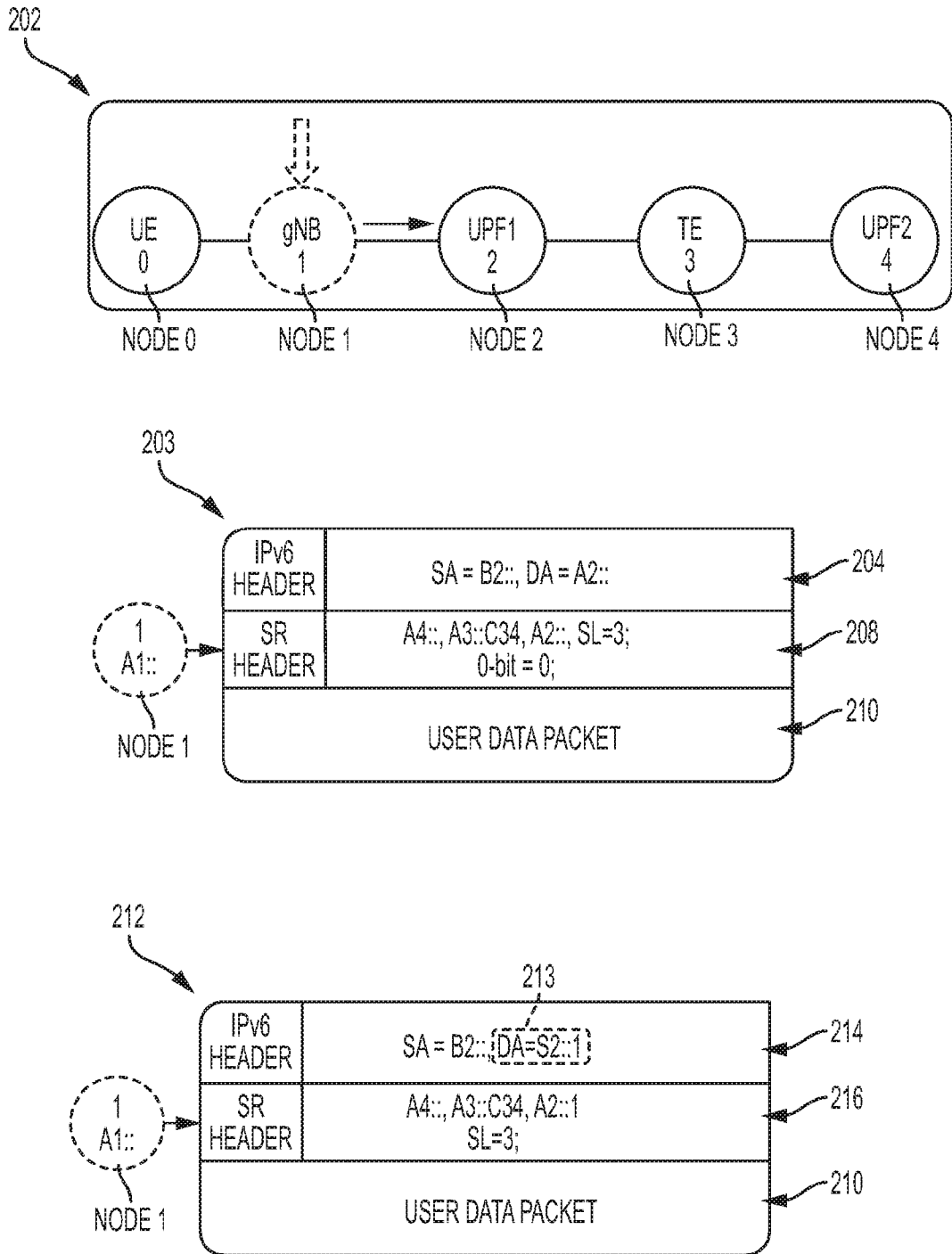


FIG. 2

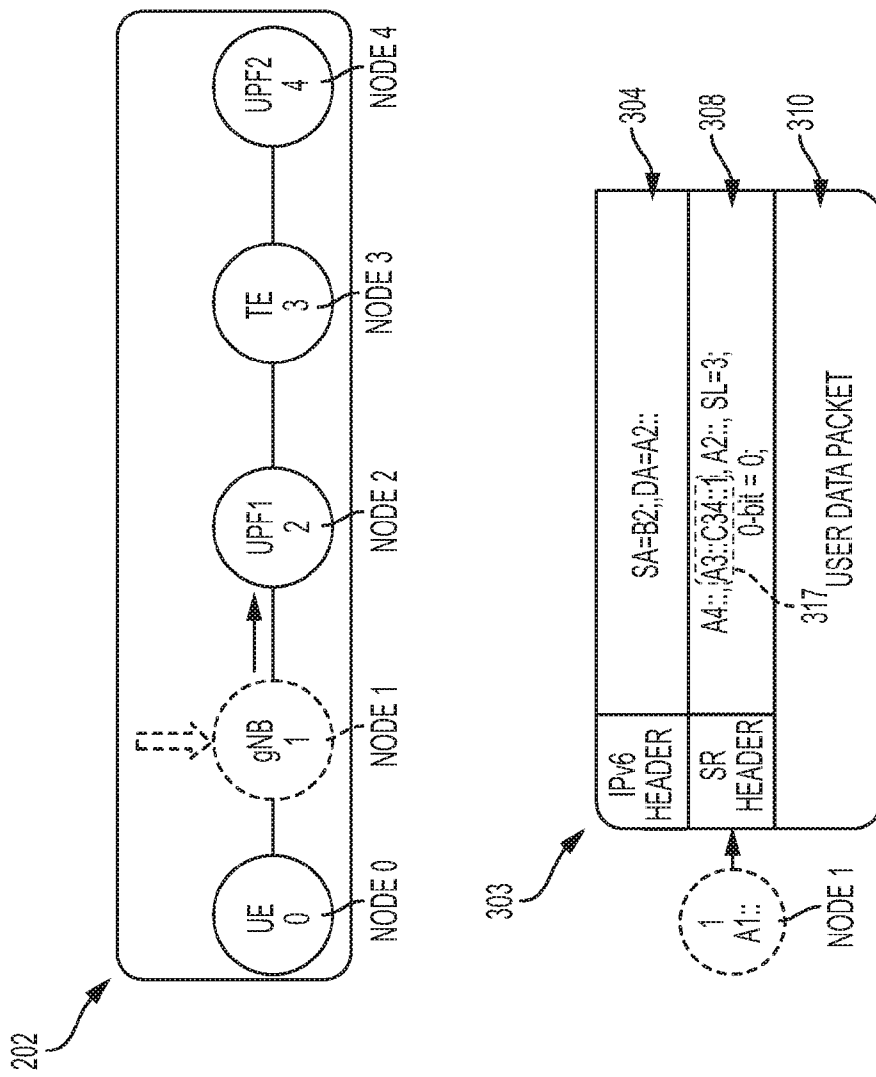


FIG. 3

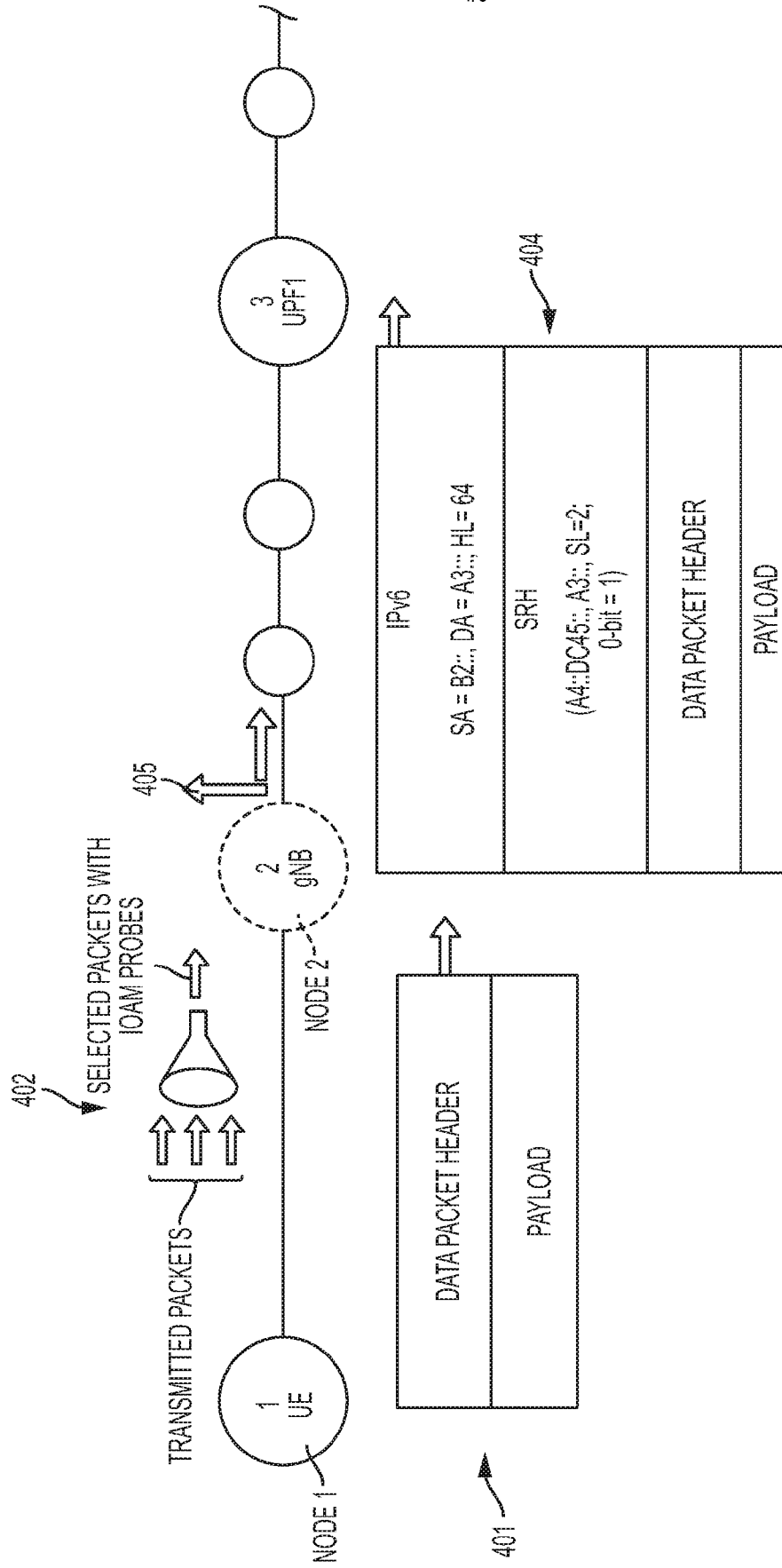


FIG. 4

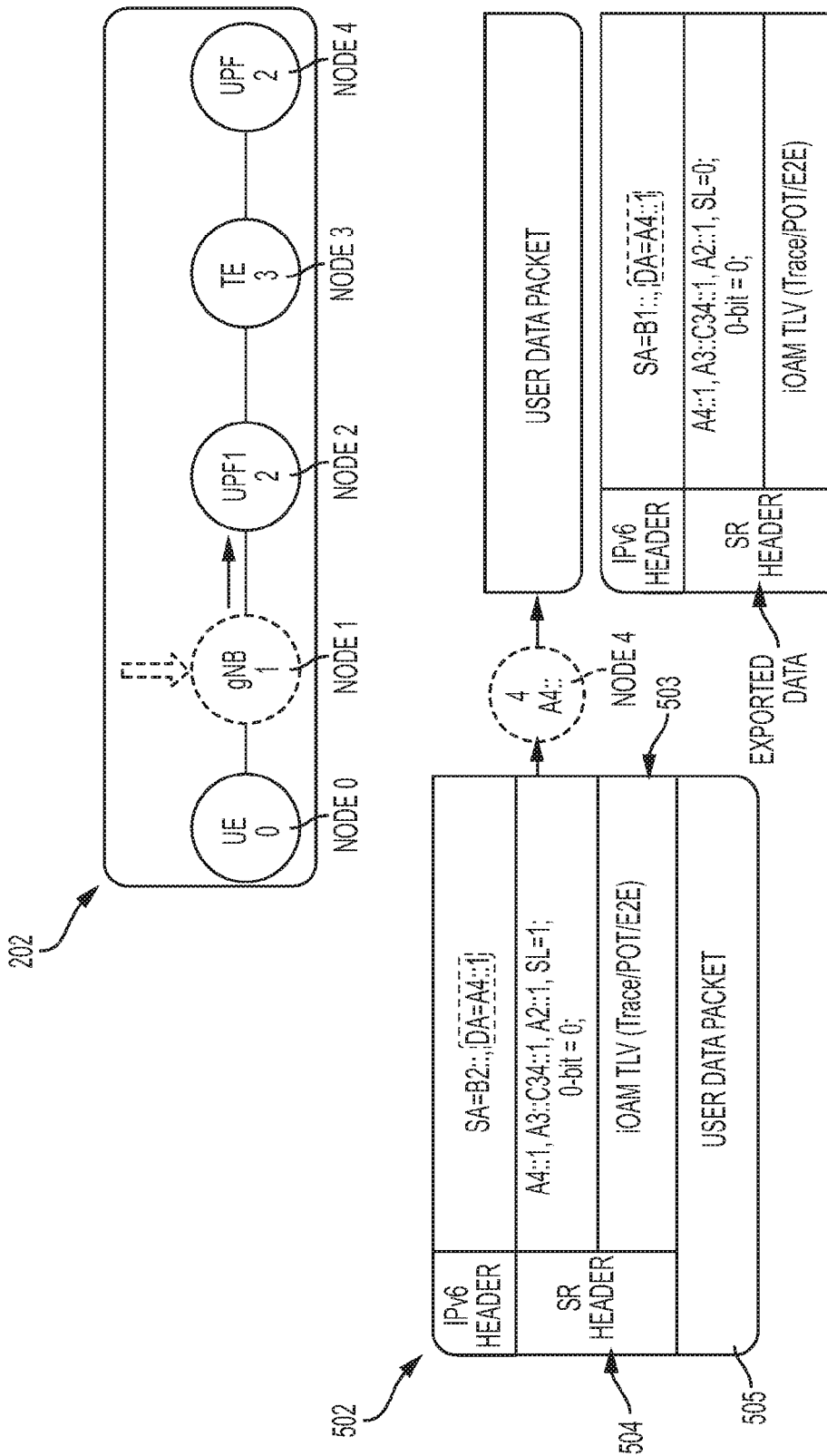


FIG. 5

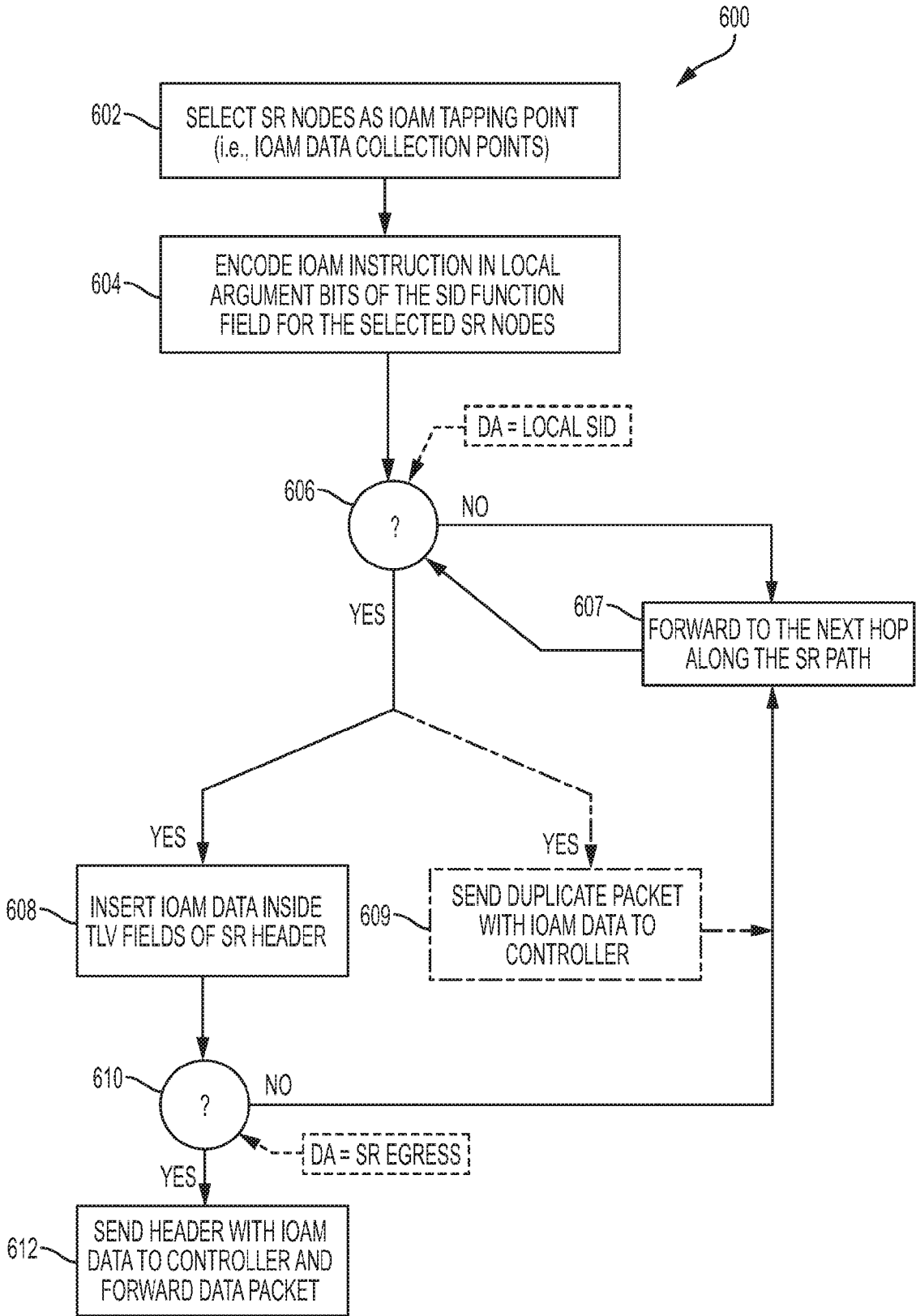


FIG. 6

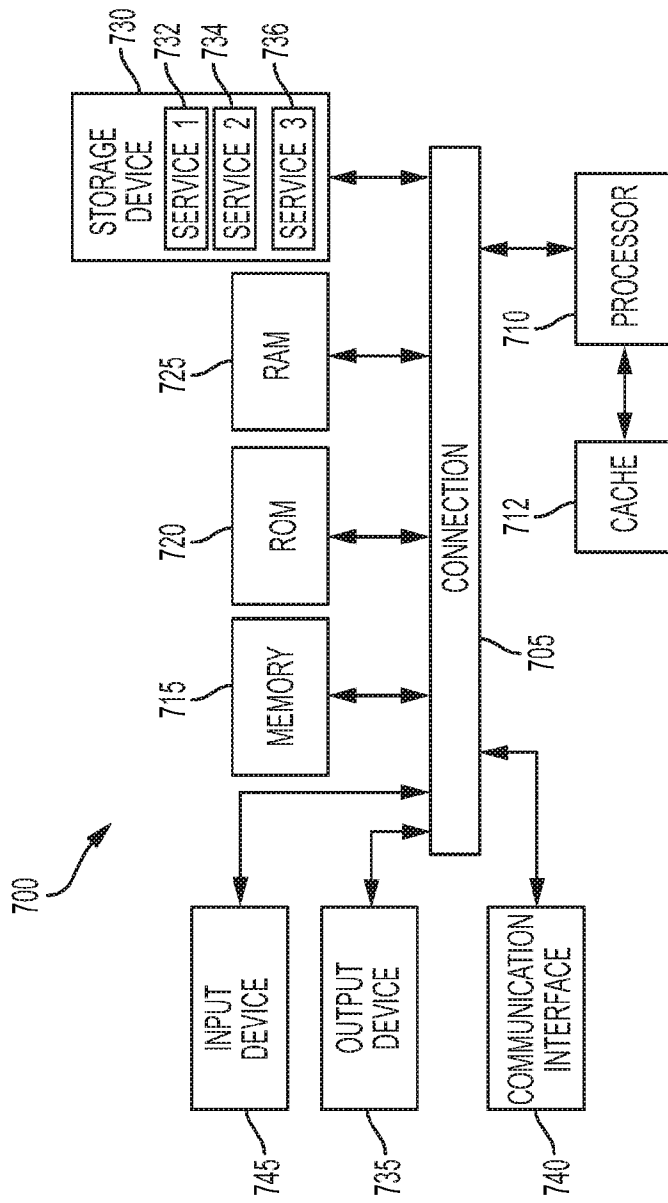


FIG. 7

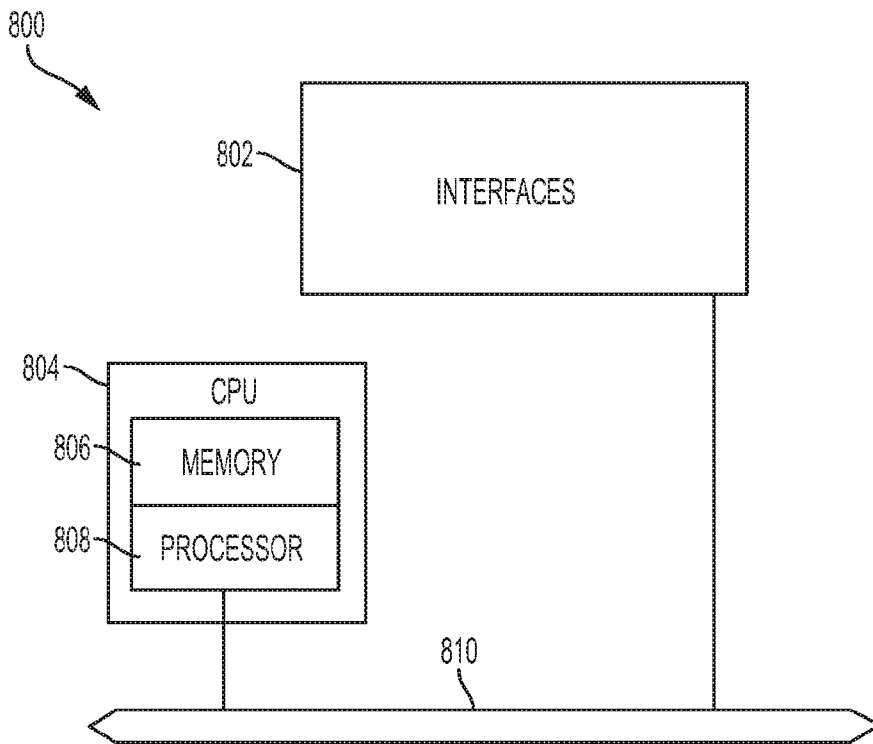


FIG. 8

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2019/052168

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L12/721 H04L12/26
ADD.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED
Minimum documentation searched (classification system followed by classification symbols)
H04L
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	BROCKNERS S BHANDARI V GOVINDAN C PIGNATARO CISCO H GREDLER RTBRICK INC J LEDDY COMCAST S YOUELL JMPC T MIZRAHI MARVELL D MOZES ME: "Encapsulations for In-situ OAM Data; draft-brockners-inband-oam-transport-05.tx t", ENCAPSULATIONS FOR IN-SITU OAM DATA; DRAFT-BROCKNERS-INBAND-OAM-TRANSPORT-05.TX T; INTERNET-DRAFT: IPPM, INTERNET ENGINEERING TASK FORCE, IETF; STANDARDWORKINGDRAFT, INTERNET SOCIETY (ISOC) 4, RUE DES FALAISES CH- 1205 GENEVA, SWITZERLAND, no. 5, 3 July 2017 (2017-07-03), pages 1-30, XP015120710, [retrieved on 2017-07-03] paragraphs [6.1" In] - [situOAMTracinginNSH"] -/--	1-4, 9-12, 17-20

Further documents are listed in the continuation of Box C.

See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance	"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"E" earlier application or patent but published on or after the international filing date	"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search 21 November 2019	Date of mailing of the international search report 04/12/2019
--	---

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

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2019/052168

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	<p>paragraphs [7"ln] - [situOAMMetadataTransportinSR"] ----- BROCKNERS S BHANDARI C PIGNATARO CISCO H GREDLER RTBRICK INC J LEDDY COMCAST S YOUPELL JPMC T MIZRAHI MARVELL D MOZES P LAPUKHOV FA: "Data Fields for In-situ OAM; draft-ietf-ippm-ioam-data-03.txt", DATA FIELDS FOR IN-SITU OAM; DRAFT-IETF-IPPM-IOAM-DATA-03.TXT; INTERNET-DRAFT: IPPM, INTERNET ENGINEERING TASK FORCE, IETF; STANDARDWORKINGDRAFT, INTERNET SOCIETY (ISOC) 4, RUE DES FALAISES CH- 1205 GENEVA, SWITZERLAND, no. 3, 27 June 2018 (2018-06-27), pages 1-35, XP015127115, [retrieved on 2018-06-27] paragraph [4.1"IOAMTracingOptions"] paragraph [6"IOAMDataExport"] -----</p>	1,5-9, 13-17
A	<p>US 2018/176134 A1 (PIGNATARO CARLOS M [US] ET AL) 21 June 2018 (2018-06-21) paragraph [0033] - paragraph [0036]; claims 1-11 -----</p>	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2019/052168

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2018176134	A1	NONE	