



(51) International Patent Classification:

H04L 41/06 (2022.01) *G06F 11/16* (2006.01)
H04L 43/0823 (2022.01)

(21) International Application Number:

PCT/IN2024/051190

(22) International Filing Date:

12 July 2024 (12.07.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

202321048623 19 July 2023 (19.07.2023) IN

(71) Applicant: **JIO PLATFORMS LIMITED** [IN/IN]; Of-
fice-101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Am-
bawadi, Ahmedabad, Gujarat 380006 (IN).

(72) Inventors: **BHATNAGAR, Aayush**; Tower-7, 15B, Bev-
erly Park, Sector-14 Koperkhairane, Navi Mumbai, Ma-
harashtra 400701 (IN). **SHETTY, Mukta**; Flat No 302,
Mukund Park, Sheetal Nagar, Mira Road (East), Thane, Ma-
harashtra 401107 (IN). **JHA, Alok B**; B1-1701, G21 Av-
enue, Sector 83, Vatika INXT, Gurugram, Haryana 122004
(IN). **KUMAR, Sanjeev**; House No. 8, V.P.O. - Kalawar,
Tehsil Jagadhri, Distt - Yamuna Nagar, Haryana 133103
(IN).

(74) Agent: **DJ, Solomon et al.**; De Penning & De Penning, 120
Velachery Main Road, Guindy, Chennai India 600032 (IN).

(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, CV, CZ, DE, DJ, DK, DM,
DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,
HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG,

(54) Title: SYSTEM AND METHOD FOR IDENTIFYING NETWORK ENTITIES CAUSING ERRORS AND TIMEOUTS

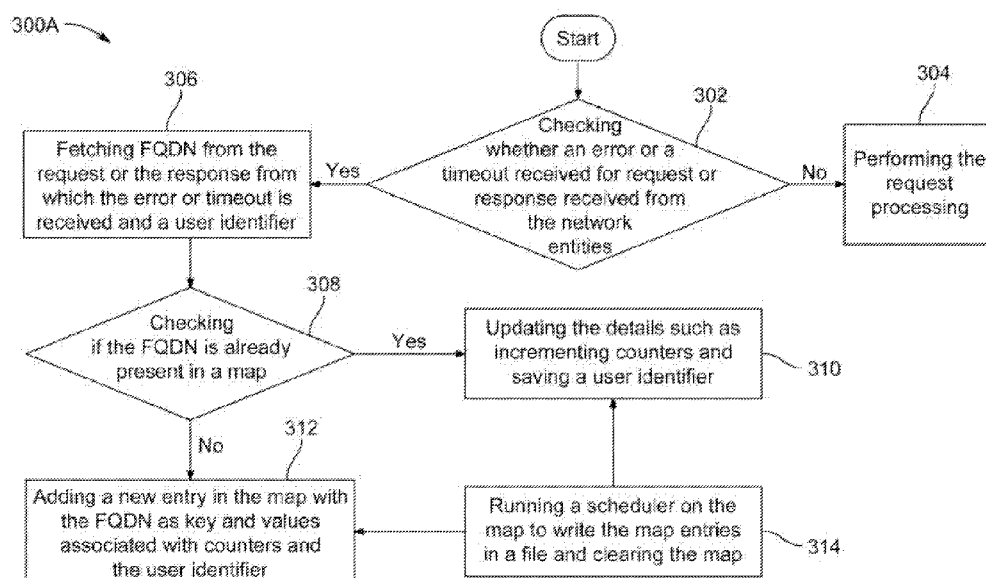


FIG. 3A

(57) Abstract: The present disclosure relates to a system and a method for identifying network entities(110) causing errors and time-outs in a network(106). The system comprises receiving a request for accessing services from a first network entity of plurality of network entities(110) and forwards the received request to a second network entity in the network for fulfilling the received request. The system receives a response from the second network entity. When the received request or the received response is indicative of an error or a timeout, the system extracts a set of error attributes including a first unique identifier associated with the network entity, timestamps associated with the received request and the received response, error data, and a second unique identifier associated with a user equipment (UE)(104). The system may identify the network entities based on the error attributes extracted therefrom. The set of error attributes is stored in a database as key-value pairs.

KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

- *with international search report (Art. 21(3))*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*
- *in black and white; the international application as filed contained color or greyscale and is available for download from PATENTSCOPE*

SYSTEM AND METHOD FOR IDENTIFYING NETWORK ENTITIES CAUSING ERRORS AND TIMEOUTS

RESERVATION OF RIGHTS

5 [0001] A portion of the disclosure of this patent document contains material, which is subject to intellectual property rights such as, but are not limited to, copyright, design, trademark, Integrated Circuit (IC) layout design, and/or trade dress protection, belonging to Jio Platforms Limited (JPL) or its affiliates (hereinafter referred as owner). The owner has no objection to the
10 facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all rights whatsoever. All rights to such intellectual property are fully reserved by the owner.

15 FIELD OF DISCLOSURE

[0002] The embodiments of the present disclosure generally relate to communication networks. In particular, the present disclosure relates to a system and a method for identifying network entities causing errors and timeouts.

20 DEFINITION

[0003] As used in the present disclosure, the following terms are generally intended to have the meaning as set forth below, except to the extent that the context in which they are used to indicate otherwise.

[0004] A “diameter routing agent (DRA)” is a functional element in a 3G
25 or 4G (such as LTE) network that provides real-time routing capabilities to ensure that messages are routed among the correct elements in a network.

[0005] A “fully qualified domain name (FQDN)” is a complete domain name for a specific computer, resource, or network entity in a network. The FQDN includes both the hostname and the domain name, providing a precise and
30 unique identifier for the resource within the larger domain name system (DNS).

[0006] A “command line interface (CLI)” offers a unified control interface to the router and can be used to get/set configuration parameters, apply updates, restart services or perform other system tasks.

BACKGROUND OF DISCLOSURE

[0007] The following description of related art is intended to provide background information pertaining to the field of the disclosure. This section may include certain aspects of the art that may be related to various features of the present disclosure. However, it should be appreciated that this section be used only to enhance the understanding of the reader with respect to the present disclosure, and not as admissions of prior art.

[0008] Policy and Charging Rules Function (PCRF) units play an important role in coordinating between multiple network entities in cellular networks. The network entities may include IP Multimedia Subsystems (IMS), data transmission nodes of Internet, application servers, Home Subscriber Server (HSS), Access and Mobility Management Function (AMF) units, Session Management Function (SMF) units, and the like. PCRFs may be connected to the network entities or gateways via Diameter Routing Agents (DRA). During a sudden increase in error or timeout scenarios from gateways that are not directly connected with PCRF, and since there may be multiple instances of the same network entity, it becomes difficult to identify which instance is sending those errors uniquely. Errors and timeouts may be caused by malfunctioning and responding the requests with errors or not accepting the request in some other cases.

[0009] Existing solutions often involve tracing the error-causing component using the Transmission Control Protocol (TCP) logs dumped after an error is detected. While error and timeout statistics may help in resolving the identity of the network entity causing errors and timeouts, existing solutions do not collect and store error data appropriately to identify and analyze the errors.

[0010] There is, therefore, a need in the art to provide a method and a system that can overcome the shortcomings of the existing prior arts.

SUMMARY

[0011] In an exemplary embodiment, a method for identifying at least one network entity from a plurality of network entities causing errors and timeouts in

a network is described. The method comprises receiving a request for accessing at least one of plurality of services from a first network entity and forwarding the received request to a second network entity. The method further comprises receiving a response from the second network entity. The method comprises
5 detecting an error or a timeout received in the received request or the received response. On detecting the error or the timeout in the received request or the received response, fetching a first identifier and a second identifier corresponding to the detected error or the detected timeout. The method comprises identifying the network entity causing the detected error or the detected timeout based on the
10 fetched first identifier.

[0012] In some embodiments, the method further comprises checking whether the fetched first identifier is already present in a map. The map is configured to store the plurality of statistics in a key-value pair. On detecting that the fetched first identifier is already present in the map, incrementing a first
15 counter and a second counter, accordingly. The method comprises saving the second identifier in the map. On detecting that the fetched first identifier is not present in the map, adding a new entry in the map with a plurality of statistics.

[0013] In some embodiments, the first identifier is a fully qualified domain name (FQDN) associated with the network entity and the second identifier is a
20 user identifier associated with the network entity.

[0014] In some embodiments, the plurality of network entities comprises a user equipment (UE), a network function, a network node, and a network element.

[0015] In some embodiments, the key of the key-value pair includes the FQDN, and the value of the key-value pair includes values corresponding to the
25 first counter, values corresponding to the second counter, a list of the second identifiers, and an error started timestamp.

[0016] In some embodiments, the first counter is configured to maintain a count for errors, and the second counter is configured to maintain a count for timeouts.

30 **[0017]** In some embodiments, processing the request when no error or the timeout is detected in the received request or the received response.

[0018] In some embodiments, the plurality of statistics is monitored through a command line interface (CLI).

[0019] In some embodiments, the list of second identifiers is configured and updated using the CLI responsive to increase in number of the second
5 identifiers required for analysis at a run time.

[0020] In some embodiments, a scheduler is run to write a plurality of entries in the map and clear the map.

[0021] In another exemplary embodiment, a system for identifying at least one network entity causing errors and timeouts in a network is described. The
10 system comprises a request processing unit configured to receive a request for accessing at least one of plurality of services from a first network entity, forward the received request to a second network entity. The request processing unit is further configured to receive a response from the second network entity. An
15 identification unit is configured to detect an error, or a timeout received in the received request or the received response. On detecting that the error or the timeout in the received request or the received response, the identification unit is configured to fetch a first identifier and a second identifier corresponding to the detected error or the detected timeout. The identification unit is configured to identify the network entity causing the detected error or the detected timeout
20 based on the fetched first identifier.

[0022] In some embodiments, the identification unit is configured to check whether the fetched first identifier is already present in a map. The map is configured to store the plurality of statistics in a key-value pair. On detecting that the fetched first identifier is already present in the map, an error log management
25 unit is configured to increment a first counter and a second counter accordingly. The error log management unit is configured to save the second identifier. On detecting that the fetched first identifier is not present in the map, the error log management unit is configured to add a new entry in the map with a plurality of statistics.

[0023] In some embodiments, the first identifier comprises a fully qualified domain name (FQDN) associated with the network entity and the second identifier comprises a user identifier associated with a user equipment (UE).

[0024] In some embodiments, the plurality of network entities comprises a
5 user equipment (UE), a network function, a network node, and a network element.

[0025] In some embodiments, the key of the key-pair value includes the FQDN, and the value of the key-value pair includes values corresponding to the first counter, values corresponding to the second counter, a list of the second identifiers, and an error-started timestamp.

10 [0026] In some embodiments, the first counter is configured to maintain a count for errors, and the second counter is configured to maintain a count for timeouts.

[0027] In some embodiments, when no error or the timeout is detected in the received request or the received response, the request processing unit is
15 configured to process the request.

[0028] In some embodiments, the plurality of statistics is monitored through a command line interface (CLI).

[0029] In some embodiments, the list of second identifiers is configured and updated using the CLI responsive to increase in number of the second
20 identifiers required for analysis at a run time.

[0030] In some embodiments, a scheduler is run to write a plurality of entries in the map and clear the map.

[0031] In some embodiments, a user equipment is communicatively coupled with a system. The coupling comprises steps of receiving, by the system,
25 a connection request and sending, by the system, an acknowledgment of the connection request to the UE. The coupling further comprises transmitting a plurality of signals in response to the connection request. The system is configured for identifying a network entity causing errors and timeouts in a network.

[0032] Other objects and advantages of the present disclosure will be more apparent from the following description, which is not intended to limit the scope of the present disclosure.

5 OBJECTS OF THE PRESENT DISCLOSURE

[0033] Some of the objects of the present disclosure, which at least one embodiment herein satisfies are as listed herein below.

[0034] An object of the present disclosure is to provide a system and a method for identifying network entities in a network causing errors and timeouts.

10 [0035] Another object of the present disclosure is to provide a system and a method that reduces time tracing unique identifiers associated with the network entity causing errors and timeouts.

[0036] Another object of the present disclosure is to provide a system and a method that identifies unique identifiers associated with the user equipments,
15 analyzing the cause of errors and timeouts.

[0037] Another object of the present disclosure is to provide a system and a method that maintains a queue of error or timeout data to prevent overflow of a list of unique identifiers.

[0038] Another object of the present disclosure is to provide a system and
20 a method for obtaining and analyzing data associated with the errors and timeouts using a command line interface (CLI).

BRIEF DESCRIPTION OF DRAWINGS

[0039] The accompanying drawings, which are incorporated herein, and
25 constitute a part of this disclosure, illustrate exemplary embodiments of the disclosed methods and systems in which like reference numerals refer to the same parts throughout the different drawings. Components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present disclosure. Some drawings may indicate the components
30 using block diagrams and may not represent the internal circuitry of each component. It will be appreciated by those skilled in the art that disclosure of such

drawings includes the disclosure of electrical components, electronic components or circuitry commonly used to implement such components.

[0040] FIG. 1 illustrates an exemplary architecture for implementing a system for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0041] FIG. 2 illustrates a block diagram of a system for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0042] FIG. 3A illustrates a sequence diagram for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0043] FIG. 3B illustrates a flowchart for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0044] FIG. 4 illustrates an exemplary computer system (400) in which or with which embodiments of the present disclosure may be implemented.

[0045] The foregoing shall be more apparent from the following more detailed description of the disclosure.

LIST OF REFERENCE NUMERALS

- 100 – Network Architecture
- 102-1, 102-2...102-N – Users
- 104-1, 104-2...104-N – User Equipments
- 108 – System
- 110-1, 110-2...110-N – Network Entities
- 112-1, 112-2 – Base Stations
- 114 – Diameter Routing Agent
- 200 – Block Diagram
- 202 – One or more processor(s)
- 204 – Memory
- 206 – Interfaces
- 208 – Processing Engine

- 210 - Database
- 212 – Request Processing Unit
- 214 – Identification Unit
- 216 – Error Log Management Unit
- 5 218 – Other Units
 - 300A – Flow Diagram
 - 302 – Step
 - 304 – Step
 - 306 – Step
 - 10 308 – Step
 - 310 – Step
 - 312 – Step
 - 314 – Step
 - 300B – Flow Diagram
 - 15 322 – Step
 - 324 – Step
 - 326 – Step
 - 328- Step
 - 330 – Step
 - 20 332 – Step
 - 400 – Computing System
 - 410 – External Storage Device
 - 420 – Bus
 - 430 – Main Memory
 - 25 440 – Read Only Memory
 - 450 – Mass Storage Device
 - 460 – Communication Port
 - 470 – Processor

DETAILED DESCRIPTION OF DISCLOSURE

[0046] In the following description, for the purposes of explanation, various specific details are set forth in order to provide a thorough understanding of embodiments of the present disclosure. It will be apparent, however, that
5 embodiments of the present disclosure may be practiced without these specific details. Several features described hereafter can each be used independently of one another or with any combination of other features. An individual feature may not address all of the problems discussed above or might address only some of the problems discussed above. Some of the problems discussed above might not be
10 fully addressed by any of the features described herein.

[0047] The ensuing description provides exemplary embodiments only, and is not intended to limit the scope, applicability, or configuration of the disclosure. Rather, the ensuing description of the exemplary embodiments will provide those skilled in the art with an enabling description for implementing an
15 exemplary embodiment. It should be understood that various changes may be made in the function and arrangement of elements without departing from the spirit and scope of the disclosure as set forth.

[0048] Specific details are given in the following description to provide a thorough understanding of the embodiments. However, it will be understood by
20 one of ordinary skill in the art that the embodiments may be practiced without these specific details. For example, circuits, systems, networks, processes, and other components may be shown as components in block diagram form in order not to obscure the embodiments in unnecessary detail. In other instances, well-known circuits, processes, algorithms, structures, and techniques may be shown
25 without unnecessary detail in order to avoid obscuring the embodiments.

[0049] Also, it is noted that individual embodiments may be described as a process which is depicted as a flowchart, a flow diagram, a data flow diagram, a structure diagram, or a block diagram. Although a flowchart may describe the operations as a sequential process, many of the operations can be performed in
30 parallel or concurrently. In addition, the order of the operations may be re-arranged. A process is terminated when its operations are completed but could

have additional steps not included in a figure. A process may correspond to a method, a function, a procedure, a subroutine, a subprogram, etc. When a process corresponds to a function, its termination can correspond to a return of the function to the calling function or the main function.

5 **[0050]** The word “exemplary” and/or “demonstrative” is used herein to mean serving as an example, instance, or illustration. For the avoidance of doubt, the subject matter disclosed herein is not limited by such examples. In addition, any aspect or design described herein as “exemplary” and/or “demonstrative” is not necessarily to be construed as preferred or advantageous over other aspects or
10 designs, nor is it meant to preclude equivalent exemplary structures and techniques known to those of ordinary skill in the art. Furthermore, to the extent that the terms “includes,” “has,” “contains,” and other similar words are used in either the detailed description or the claims, such terms are intended to be inclusive in a manner similar to the term “comprising” as an open transition word
15 without precluding any additional or other elements.

[0051] Reference throughout this specification to “one embodiment” or “an embodiment” or “an instance” or “one instance” means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the present disclosure. Thus, the
20 appearances of the phrases “in one embodiment” or “in an embodiment” in various places throughout this specification are not necessarily all referring to the same embodiment. Furthermore, the particular features, structures, or characteristics may be combined in any suitable manner in one or more embodiments.

25 **[0052]** The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the disclosure. As used herein, the singular forms “a,” “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will be further understood that the terms “comprises” and/or “comprising,” when used in
30 this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or

addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof. As used herein, the term “and/or” includes any and all combinations of one or more of the associated listed items.

[0053] The present disclosure relates to a system and a method for
5 identifying network entities causing errors and timeouts. The system receives a request for accessing services from a first network entity of a plurality of network entities and forwarding the received request to a second network entity in the network for fulfilling the received request. The system receives a response from the second network entity. When the received request or the received response is
10 indicative of an error or a timeout, the system extracts a set of error attributes including a first unique identifier associated with the network entity, timestamps associated with the received request and the received response, error data, and a second unique identifier associated with a user equipment (UE). The system may identify the network entities based on the error attributes extracted therefrom. The
15 set of error attributes is stored in a database as key-value pairs. The system receives the error attributes in a queue and periodically clears logs thereof.

[0054] The various embodiments throughout the disclosure will be explained in more detail with reference to FIGS. 1-4.

[0055] FIG. 1 illustrates an exemplary network architecture (100) for
20 implementing a system (108) for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0056] Referring to FIG. 1, the network architecture (100) may include one or more computing devices or user equipments (104-1, 104-2...104-N) associated with one or more users (102-1, 102-2...102-N) in an environment. A
25 person of ordinary skill in the art will understand that one or more users (102-1, 102-2...102-N) may be individually referred to as the user (102) and collectively referred to as the users (102). Similarly, a person of ordinary skill in the art will understand that one or more user equipments (104-1, 104-2...104-N) may be individually referred to as the user equipment (104) and collectively referred to as
30 the user equipment (104). A person of ordinary skill in the art will appreciate that the terms “computing device(s)” and “user equipment” may be used

interchangeably throughout the disclosure. Although two user equipments (104) are depicted in FIG. 1, however any number of the user equipments (104) may be included without departing from the scope of the ongoing description.

[0057] In an embodiment, the user equipment (104) may include smart devices operating in a smart environment, for example, an Internet of Things (IoT) system. In such an embodiment, the user equipment (104) may include, but is not limited to, smart phones, smart watches, smart sensors (e.g., mechanical, thermal, electrical, magnetic, etc.), networked appliances, networked peripheral devices, networked lighting system, communication devices, networked vehicle accessories, networked vehicular devices, smart accessories, tablets, smart television (TV), computers, smart security system, smart home system, other devices for monitoring or interacting with or for the users (102) and/or entities, or any combination thereof. A person of ordinary skill in the art will appreciate that the user equipment (104) may include, but is not limited to, intelligent, multi-sensing, network-connected devices, that can integrate seamlessly with each other and/or with a central server or a cloud-computing system or any other device that is network-connected.

[0058] In an embodiment, the user equipment (104) may include, but is not limited to, a handheld wireless communication device (e.g., a mobile phone, a smartphone, a phablet device, and so on), a wearable computer device (e.g., a head-mounted display computer device, a head-mounted camera device, a wristwatch computer device, and so on), a Global Positioning System (GPS) device, a laptop computer, a tablet computer, or another type of portable computer, a media playing device, a portable gaming system, and/or any other type of computer device with wireless communication capabilities, and the like. In an embodiment, the user equipment (104) may include, but is not limited to, any electrical, electronic, electro-mechanical, or an equipment, or a combination of one or more of the above devices such as virtual reality (VR) devices, augmented reality (AR) devices, laptop, a general-purpose computer, desktop, personal digital assistant, tablet computer, mainframe computer, or any other computing device, wherein the user equipment (104) may include one or more in-built or

externally coupled accessories including, but not limited to, a visual aid device such as a camera, an audio aid, a microphone, a keyboard, and input devices for receiving input from the user (102) or the entity such as touch pad, touch enabled screen, electronic pen, and the like. A person of ordinary skill in the art will appreciate that the user equipment (104) may not be restricted to the mentioned devices and various other devices may be used.

[0059] Referring to FIG. 1, the user equipment (104) may communicate with the system (108) via a network (106). In an embodiment, the network (106) may include at least one of a Fifth Generation (4G) network, Sixth Generation (6G) network, or the like. The network (106) may enable the user equipment (104) to communicate with other devices in the network architecture (100) and/or with the system (108). The network (106) may include a wireless card or some other transceiver connection to facilitate this communication. In another embodiment, the network (106) may be implemented as, or include any of a variety of different communication technologies such as a wide area network (WAN), a local area network (LAN), a wireless network, a mobile network, a Virtual Private Network (VPN), the Internet, the Public Switched Telephone Network (PSTN), or the like. In an embodiment, the network (106) may include one or more base stations (112) for facilitating communication between the one or more UEs (104). The network (106) may be formed by a set of base stations (112) communicatively coupled to enable telecommunication exchanges between one or more UEs (104).

[0060] The base station (112) may be a network infrastructure that provides wireless access to one or more terminals associated therewith. The base station may have coverage defined to be a predetermined geographic area based on the distance over which a signal may be transmitted. The base station (112) may include, but not be limited to, wireless access point, evolved NodeB (eNodeB), 5G node or next generation NodeB (gNB), wireless point, transmission/reception point (TRP), and the like. In an embodiment, the base station (112) may include one or more operational units that enable telecommunication between two or more UEs (104). In an embodiment, the one or

more operational units may include, but not be limited to, transceivers, baseband unit (BBU), (remote radio unit - RRU), antennae, mobile switching centres, radio network control units, one or more processors associated thereto.

[0061] In an embodiment, the network (106) may include one or more network entities (110) depicted by network entity 1 (110-1) and network entity 2 (110-2) in FIG. 1. In an embodiment, the network entities (110) may include, but not limited to, a serving gateway (S-GW), a packet data network (PDN) gateway (P-GW), a mobility management entity (MME), and the like. In embodiments where the network (106) is a 5G network, the network (106) may further include a user plane, and a control plane having network entities (110) such as an access and mobility management function (AMF) unit, a session management function (SMF) unit, and a network exposure function (NEF) unit, but not limited thereto. In an embodiment, the network entities (110) forming the network (106) may be implemented as a hardware component, software component, or any combination thereof. In an embodiment, the network entities (110) may have a unique identifier associated therewith. In an example, the unique identifier may be a fully qualified domain name (FQDN), a uniform resource identifier (URI), a source internet protocol (IP), and the like. In an aspect, the network entities (110) comprise a user equipment (UE) (104), a network function, a network node, and a network element.

[0062] In an embodiment, the system (108) may be a network entity (110) indicative of policy and charging rules function (PCRF). In an embodiment, the PCRF may be configured to determine policy and charging rules associated with the UEs (104) attempting to connect to the network (106). In an embodiment, the PCRF may be configured to interact with the network entities (110) via a diameter routing agent (DRA) (114). The DRA (114) may communicate with the network entities (110) by transmitting application programming interface (API) call requests to the network entities (110) and receiving API responses therefrom.

[0063] In accordance with embodiments of the present disclosure, the system (108) may be designed and configured for identifying network entities (110) causing errors and timeouts. In an embodiment, the system (108) may

distinguish multiple instances of network entities (110) from those that are causing errors or timeouts.

[0064] In an embodiment, the user equipment (104) is communicatively coupled with the system (108). The system (108) may receive a connection request from the UE (104). The system (108) may send an acknowledgment of the connection request to the UE (104). The UE (104) may transmit a plurality of signals in response to the connection request. The system (108) may configure for automatically identifying a network entity causing errors and timeouts in the network (106).

10 [0065] FIG. 2 illustrates a block diagram (200) of the system (108), in accordance with embodiments of the present disclosure.

[0066] In an aspect, the system (108) may include one or more processor(s) (202). The one or more processor(s) (202) may be implemented as one or more microprocessors, microcomputers, microcontrollers, edge or fog microcontrollers, digital signal processors, central processing units, logic circuitries, and/or any devices that process data based on operational instructions. Among other capabilities, the one or more processor(s) (202) may be configured to fetch and execute computer-readable instructions stored in a memory (204) of the system (108). The memory (204) may be configured to store one or more computer-readable instructions or routines in a non-transitory computer readable storage medium, which may be fetched and executed to create or share data packets over a network service. The memory (204) may comprise any non-transitory storage device including, for example, volatile memory such as Random Access Memory (RAM), or non-volatile memory such as Erasable Programmable Read-Only Memory (EPROM), flash memory, and the like.

[0067] Referring to FIG. 2, the system (108) may include an interface(s) (206). The interface(s) (206) may include a variety of interfaces, for example, interfaces for data input and output devices, referred to as I/O devices, storage devices, and the like. The interface(s) (206) may facilitate communication to/from the system (108). The interface(s) (206) may also provide a communication pathway for one or more components of the system (108). Examples of such

components include, but are not limited to, processing unit/engine(s) (208) and a database (210).

[0068] In an embodiment, the processing unit/engine(s) (208) may be implemented as a combination of hardware and programming (for example, programmable instructions) to implement one or more functionalities of the processing engine(s) (208). In examples described herein, such combinations of hardware and programming may be implemented in several different ways. For example, the programming for the processing engine(s) (208) may be processor-executable instructions stored on a non-transitory machine-readable storage medium and the hardware for the processing engine(s) (208) may include a processing resource (for example, one or more processors), to execute such instructions. In the present examples, the machine-readable storage medium may store instructions that, when executed by the processing resource, implement the processing engine(s) (208). In such examples, the system (108) may include the machine-readable storage medium storing the instructions and the processing resource to execute the instructions, or the machine-readable storage medium may be separate but accessible to the system (108) and the processing resource. In other examples, the processing engine(s) (208) may be implemented by electronic circuitry.

[0069] In an embodiment, the database (210) includes data that may be either stored or generated as a result of functionalities implemented by any of the components of the processor (202) or the processing engines (208). In an embodiment, the database (210) may be separate from the system (108). In an embodiment, the database (210) may be indicative of including, but not limited to, a relational database, a distributed database, a cloud-based database, or the like.

[0070] In an exemplary embodiment, the processing engine (208) may include one or more units selected from any of a request processing unit (212), an identification unit (214), an error log management unit (216), and other units (218) having functions that may include, but are not limited to, testing, storage, and peripheral functions, such as wireless communication unit for remote operation, audio unit for alerts and the like.

[0071] In an embodiment, the request processing unit (212) receives a request for accessing services from the first network entity (e.g., network entity (110-1)). In an example, the request for service may be indicative of a first UE (104-1) requesting to make a voice call to a second 5G UE (104-2). In an embodiment, the request processing unit (212) may redirect the received request to the corresponding second network entity (e.g., network entity (110-2)) in the network (106) for fulfilling the received request.

[0072] In an embodiment, the identification unit (214) may be configured to receive a response from the second network entity. When the response is indicative of an error or a timeout, the identification unit (214) may extract a set of error attributes including a first unique identifier associated with the network entity (110), timestamps associated with the received request and the received response, error data, and a second unique identifier associated with the UE (104). In an embodiment, errors and timeouts may be caused by malfunctioning of network entities (110) or due the network entities (110) being irresponsive for a predetermined duration to the requests. The identification unit (214) may uniquely identify the network entities (110) based on the error attributes extracted from the received request and the received response.

[0073] In an embodiment, the identification unit (214) may also be configured to extract the second unique identifier associated with the UE (104). In such embodiments, the identification unit (214) may identify whether the cause of the error or timeout relates to the UE (104) or any specifications or configurations thereof.

[0074] In an embodiment, the error log management unit (216) may store set of error attributes in the database (210) as key-value pairs. In an embodiment, if the database (210) includes an entry having the first unique identifier of the network entity (110), the error log management unit (216) may update the value associated with the entry with the set of error attributes extracted from the request or the response. In other embodiments where the key is being stored for the first time in the database (210), a new entry may be generated, and the set of error attributes may be assigned as its corresponding value. In an embodiment, the error

attributes may be stored with an error counter value to maintain a count of the number of times a network entity (110) causes errors or timeouts.

[0075] In an embodiment, the error log management unit (216) may receive the error attributes in a queue and periodically clear logs thereof. In an example, the logs may be cleared daily. In an embodiment, the error log management unit (216) may include a rollover mechanism such that a predetermined number of identifiers are captured and made accessible from memory, thereby freeing space in memory and storage. In an embodiment, a list of a predetermined number of errors or timeouts may be maintained in a queue along with timestamps associated therewith. In an embodiment, the database (210) having the error and timeout data stored therein may be made accessible via a command line interface (CLI).

[0076] Exemplary Scenario

[0077] In an example, the error attributes extracted by the system (108) may be stored as key-value pairs, where the key is a FQDN/source internet protocol (IP) of the network function and value includes data on error or timeout in a plain old java object (POJO), counters for the error and the timeouts, error start timestamp, and second unique identifiers associated with the UE (104). In such examples, two separate counters may be maintained for error and timeout each.

[0078] In an example, for a first unique identifier attribute such as “Map <String FQDN, ProblemDetails pojo>” may be stored as the key with a corresponding value of “ka1draIj2501.dra.mnbb861.mcjk405.3gppnetwork.org : PCRFGxErrorTimeoutStats [cntr_gx_error_received=3, cntr_gx_timeout_received=6, SubscriberIds=[3146223421, 3146223421, 3146223421]]”, where “PCRFGxErrorTimeoutStats “ corresponds to error or timeout attribute, cntr_gx_timeout_received to the counter value, and SubscriberIds corresponds to the second identifier attribute.

[0079] FIG. 3A illustrates a flowchart (300A) for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0080] The method includes receiving a request for accessing services from one of plurality of network entities (e.g., first network entity (110-1)). In an aspect, the request for accessing services may send by the UE (104) to the first network entity (110-1) via the network (106) of FIG. 1. The first network entity may send the received request to a corresponding network entity (e.g., second network entity (110-2)) in the network (106) for fulfilling the request. The method includes receiving a response from the second network entity (110-2). The method includes extracting a set of error attributes including a first unique identifier associated with the network entity, timestamps associated with the request and the response, error data, and a second unique identifier associated with the UE (104), when the response is indicative of an error or a timeout. The method includes storing the set of error analysis attributes in a database, such as the database (210), as key-value pairs. The method includes receiving the error analysis attributes in a queue and periodically clearing logs thereof.

15 [0081] At step 302, checking an error or timeout received in the received request or the response.

[0082] At step 304, on detecting that no error or timeout received in the request, performing request progressing (e.g., call flow processing).

20 [0083] At step 306, on detecting that error or timeout is received in the received request or the response, fetching identifier (e.g., peer FQDN) from which error or timeout is received. Also, the user identifier is fetched. Based on the fetched FQDN, determining the network entity causing errors and timeouts.

[0084] Further, at step 308, checking whether the FQDN is already present in a map. In example, the map is maintained by the system 108 comprising a key-value pair.

[0085] At step 310, on detecting the FQDN is already present in the map, updating details. Updating of details include incrementing the relevant counters and saving the user identifier. The counters may include counter for errors and the timeouts.

30 [0086] At step 312, on detecting that the fetched first identifier is not present in the map, the error log management unit (216) adds a new entry in the

map with the FQDN as key and value as details POJO with counters and the user identifier.

[0087] At step 314, a scheduler is run on the map to write map entries in a file and clearing the map. The map entries may include a key-value pair. The key part is FQDN/Source IP of the network function. The value part may include a detailed POJO containing separate counters for error and time, error started timestamp and list of user identifiers. In an example implementation, a map or a dictionary may be used to store the statistics with keys being the FQDN/Source IP and values being detailed objects (POJOs) with various counters and data. A network function may be implemented to write map entries in a file. Another function may be implemented to clear the map entries. The scheduler is configured to run these functions at a defined intervals to write and clear the map entries.

[0088] FIG. 3B illustrates a flowchart (300B) for identifying network entities causing errors and timeouts, in accordance with embodiments of the present disclosure.

[0089] At step 322, receiving a request for accessing at least one of plurality of services from a first network entity.

[0090] At step 324, forwarding the received request to a second network entity.

[0091] At step 326, receiving a response from the second network entity.

[0092] At step 328, detecting an error or a timeout received in the received request or the received response.

[0093] At step 330, on detecting that the error or the timeout is received in the received request or the received response, fetching a first identifier and a second identifier corresponding to the detected error or the detected timeout. When no error or the timeout is detected in the received request or the received response, processing the request.

[0094] At step 332, identifying the network entity causing the detected error or the detected timeout based on the fetched first identifier. The first identifier is a fully qualified domain name (FQDN) associated with the network

entity and the second identifier is a user identifier associated with a user equipment (UE).

[0095] In an embodiment, detecting whether the fetched first identifier is already present in a map. The map is configured to store a plurality of statistics in a key-value pair. The statistics may refer to error and timeout statistics. The key of the key-value pair includes a fully qualified domain name (FQDN), and the value of the key-value pair includes values corresponding to the first counter, values corresponding to the second counter, a list of the second identifiers, and an error started timestamp. The first counter is configured to maintain a count for errors and the second counter is configured to maintain a count for timeouts. In examples, data structure like a dictionary (in Python), hash map (in Java), or similar structures available in other programming languages may be used to store the plurality of statistics in a key-value pair. On detecting that the fetched first identifier is already present in the map, a first counter and a second counter are incremented, accordingly. The second identifier is saved in the map. In an aspect, a scheduler is run to write a plurality of entries in the map and clear the map.

[0096] On detecting that the fetched first identifier is not present in the map; a new entry is added in the map with the plurality of statistics.

[0097] FIG. 4 illustrates an exemplary computer system (400) in which or with which embodiments of the present disclosure may be implemented. As shown in FIG. 4, the computer system (400) may include an external storage device (410), a bus (420), a main memory (430), a read only memory (440), a mass storage device (450), a communication port (460), and a processor (470). A person skilled in the art will appreciate that the computer system (400) may include more than one processor (470) and communication ports (460). Processor (470) may include various modules associated with embodiments of the present disclosure.

[0098] In an embodiment, the communication port (460) may be any of an RS-232 port for use with a modem-based dialup connection, a 10/100 Ethernet port, a Gigabit or 10 Gigabit port using copper or fiber, a serial port, a parallel port, or other existing or future ports. The communication port (460) may be

chosen depending on a network, such a Local Area Network (LAN), Wide Area Network (WAN), or any network to which the computer system (400) connects.

[0099] In an embodiment, the memory (430) may be Random Access Memory (RAM), or any other dynamic storage device commonly known in the art. Read-only memory (440) may be any static storage device(s) e.g., but not limited to, a Programmable Read Only Memory (PROM) chips for storing static information e.g., start-up or Basic Input/Output System (BIOS) instructions for the processor (470).

[00100] In an embodiment, the mass storage (450) may be any current or future mass storage solution, which may be used to store information and/or instructions. Exemplary mass storage solutions include, but are not limited to, Parallel Advanced Technology Attachment (PATA) or Serial Advanced Technology Attachment (SATA) hard disk drives or solid-state drives (internal or external, e.g., having Universal Serial Bus (USB) and/or Firewire interfaces), one or more optical discs, Redundant Array of Independent Disks (RAID) storage, e.g., an array of disks (e.g., SATA arrays).

[00101] In an embodiment, the bus (420) communicatively couples the processor(s) (470) with the other memory, storage and communication blocks. The bus (420) may be, e.g., a Peripheral Component Interconnect (PCI)/PCI Extended (PCI-X) bus, Small Computer System Interface (SCSI), Universal Serial Bus (USB) or the like, for connecting expansion cards, drives and other subsystems as well as other buses, such a front side bus (FSB), which connects the processor (470) to the computer system (400).

[00102] Optionally, operator and administrative interfaces, e.g., a display, keyboard, joystick, and a cursor control device, may also be coupled to the bus (420) to support direct operator interaction with the computer system (400). Other operator and administrative interfaces may be provided through network connections connected through the communication port (460). Components described above are meant only to exemplify various possibilities. In no way should the aforementioned exemplary computer system (400) limit the scope of the present disclosure.

[00103] While considerable emphasis has been placed herein on the preferred embodiments, it will be appreciated that many embodiments can be made and that many changes can be made in the preferred embodiments without departing from the principles of the disclosure. These and other changes in the preferred embodiments of the disclosure will be apparent to those skilled in the art from the disclosure herein, whereby it is to be distinctly understood that the foregoing descriptive matter to be implemented merely as illustrative of the disclosure and not as limitation.

[00104] The present disclosure provides technical advancement related to managing errors and timeouts in a telecommunication network. This advancement addresses the limitations of existing techniques of error handling, which would require a time-consuming process of identifying the error by using a TCP logs dump in production environments and then filter out the error messages from the logs further investigate a source of error. The disclosure describes an organized manner of detecting errors or timeouts, fetching identifiers of the error or timeout, and identifying a network entity causing the error or timeout and rectifying it. By implementing the disclosure, a system is implemented to easily track, store, and process the errors, and identify the entities causing it to rectify the issues of the entities. This approach optimizes system performance, reduces operational risks, and ensures consistent service availability, thereby improving overall network efficiency and user experience.

ADVANTAGES OF THE PRESENT DISCLOSURE

[00105] The present disclosure provides a system and a method for identifying network entities in a network causing errors and timeouts.

[00106] The present disclosure provides a system and a method that reduces time tracing unique identifiers associated with the network entity causing errors and timeouts.

[00107] The present disclosure provides a system and a method that identifies unique identifiers associated with user equipment analysing cause of errors and timeouts.

[00108] The present disclosure provides a system and a method that maintains a queue of error or timeout data to prevent overflow of list of unique identifiers.

[00109] The present disclosure provides a system and a method for
5 obtaining analyzing data associated with the errors and timeouts using a command line interface (CLI).

CLAIMS

1. A method (300B) for identifying a network entity causing errors and timeouts in a network (106), the method comprising:
 - 5 receiving (322) a request for accessing at least one of a plurality of services from a first network entity;
 - forwarding (324) the received request to a second network entity;
 - receiving (326) a response from the second network entity;
 - detecting (328) an error or a timeout received in the received request or the received response;
 - 10 on detecting the error or the timeout in the received request or the received response, fetching (330) a first identifier and a second identifier corresponding to the detected error or the detected timeout; and
 - identifying (332) the network entity causing the detected error or the detected timeout based on the fetched first identifier.
 - 15
2. The method (300B) as claimed in claim 1 further comprising:
 - checking whether the fetched first identifier is already present in a map, wherein the map is configured to store a plurality of statistics in a key-value pair;
 - 20 on detecting that the fetched first identifier is already present in the map, incrementing a first counter and a second counter, accordingly;
 - saving the second identifier in the map; and
 - on detecting that the fetched first identifier is not present in the map, adding a new entry in the map with the plurality of statistics.
 - 25
3. The method (300B) as claimed in claim 1, wherein the first identifier is a fully qualified domain name (FQDN) associated with the network entity and the second identifier is a user identifier associated with the network entity.
- 30

4. The method (300B) as claimed in claim 1, wherein the plurality of network entities comprises a user equipment (UE), a network function, a network node, and a network element.
- 5 5. The method (300B) as claimed in claim 2, wherein the key of the key-value pair includes the FQDN, and the value of the key-value pair includes values corresponding to the first counter, values corresponding to the second counter, a list of the second identifiers, and an error started timestamp.
- 10 6. The method (300B) as claimed in claim 2, wherein the first counter is configured to maintain a count for errors, and the second counter is configured to maintain a count for timeouts.
- 15 7. The method (300B) as claimed in claim 1, further comprising processing the request when no error or the timeout is detected in the received request or the received response.
- 20 8. The method (300B) as claimed in claim 2, wherein the plurality of statistics is monitored through a command line interface (CLI).
- 25 9. The method (300B) as claimed in claim 5, wherein the list of second identifiers is configured and updated using the CLI responsive to increase in number of the second identifiers required for analysis at a run time.
10. The method (300B) as claimed in claim 1, wherein a scheduler is run to write a plurality of entries in the map and clear the map.
11. A system (108) for identifying a network entity causing errors and timeouts in a network (106), the system comprising:
a request processing unit (212) configured to:
- 30

receive a request for accessing at least one of a plurality of services from a first network entity;

forward the received request to a second network entity and is further configured to:

5 receive a response from the second network entity; and
an identification unit (214) configured to:

detect an error or a timeout received in the received request or the received response;

10 fetch, on detecting the error or the timeout in the received request or the received response, a first identifier and a second identifier corresponding to the detected error or the detected timeout; and

identify the network entity causing the detected error or the detected timeout based on the fetched first identifier.

15

12. The system (108) as claimed in claim 11, wherein:

the identification unit (214) is configured to:

20 check whether the fetched first identifier is already present in a map, wherein the map is configured to store a plurality of statistics in a key-value pair;

on detecting that the fetched first identifier is already present in the map, an error log management unit (216) configured to increment a first counter and a second counter accordingly;

25 the error log management unit (216) configured to save the second identifier; and

on detecting that the fetched first identifier is not present in the map, the error log management unit (216) is configured to add a new entry in the map with the plurality of statistics.

30 13. The system (108) as claimed in claim 11, wherein the first identifier comprises a fully qualified domain name (FQDN) associated with the

network entity and the second identifier is a user identifier associated with the network entity.

5 14. The system (108) as claimed in claim 11, wherein the plurality of network entities comprises a user equipment (UE), a network function, a network node, and a network element.

10 15. The system (108) as claimed in claim 12, wherein the key of the key-value pair includes the FQDN, and the value of the key-value pair includes values corresponding to the first counter, values corresponding to the second counter, a list of the second identifiers, and an error started timestamp.

15 16. The system (108) as claimed in claim 12, wherein the first counter is configured to maintain a count for errors, and the second counter is configured to maintain a count for timeouts.

20 17. The system (108) as claimed in claim 11, wherein when no error or the timeout is detected in the received request or the received response, the request processing unit (212) is configured to process the request.

18. The system (108) as claimed in claim 12, wherein the plurality of statistics is monitored through a command line interface (CLI).

25 19. The system (108) as claimed in claim 15, wherein:
the list of second identifiers is configured and updated using the CLI responsive to increase in number of the second identifiers required for analysis at a run time.

30 20. The system (108) as claimed in claim 11, wherein a scheduler is run to write a plurality of entries in the map and clear the map.

21. A user equipment (104) communicatively coupled with a system (108), the coupling comprises steps of:

receiving, by the system (108), a connection request;

5 sending, by the system (108), an acknowledgment of the connection request to the UE (104); and

transmitting a plurality of signals in response to the connection request, wherein the system (108) is configured for identifying at least one network entity from a plurality of network entities (110) causing errors and timeouts in a network (106), as claimed in claim 11.

10

22. A computer program product comprising a non-transitory computer-readable medium comprising instructions that, when executed by one or more processors, cause the one or more processors to perform a method (300B) for identifying a network entity causing errors and timeouts in a network (106), the method comprising:

15

receiving (322) a request for accessing at least one of a plurality of services from a first network entity;

forwarding (324) the received request to a second network entity;

receiving (326) a response from the second network entity;

20

detecting (328) an error or a timeout received in the received request or the received response;

on detecting the error or the timeout in the received request or the received response, fetching (330) a first identifier and a second identifier corresponding to the detected error or the detected timeout; and

25

identifying (332) the network entity causing the detected error or the detected timeout based on the fetched first identifier.

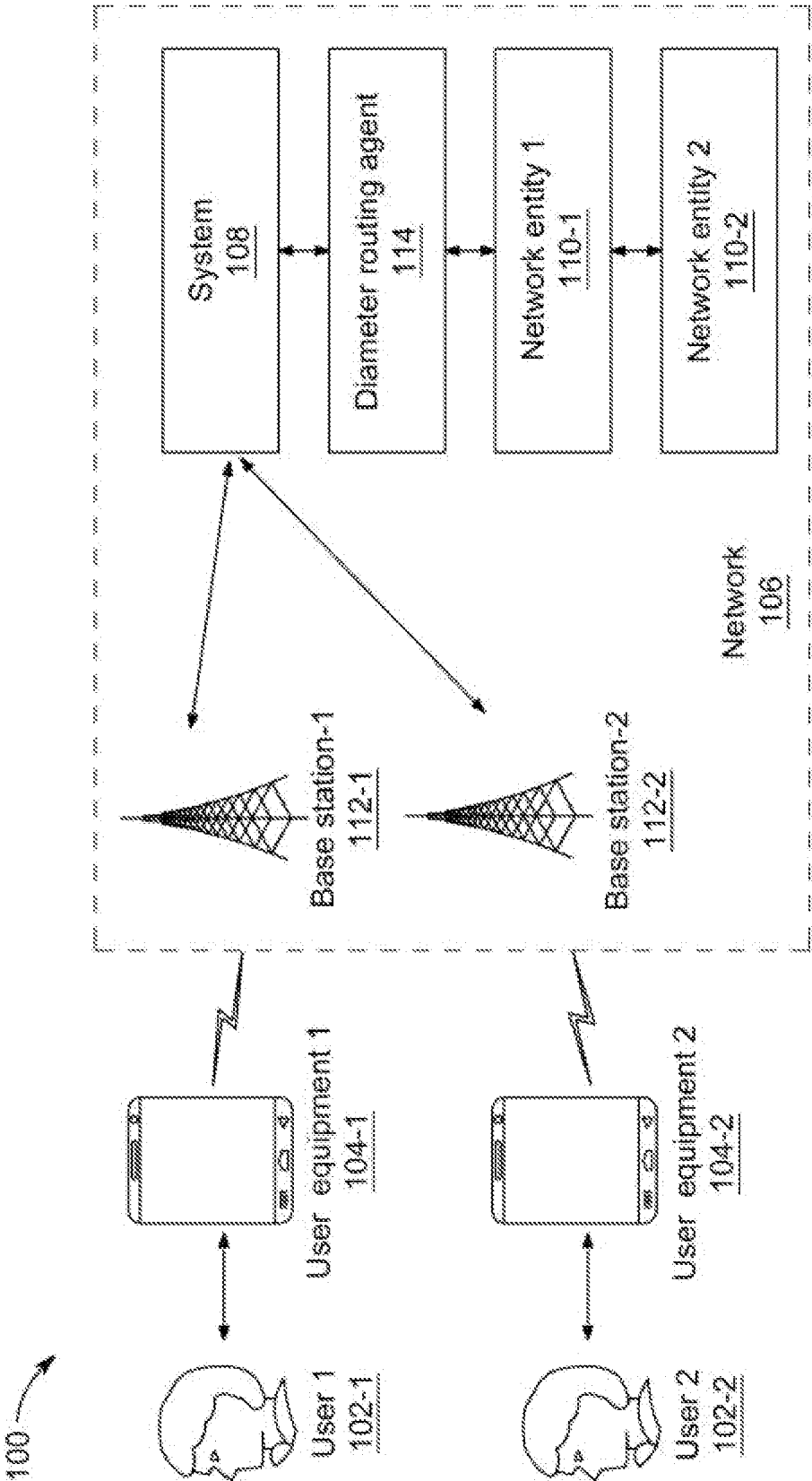


FIG. 1

2/5

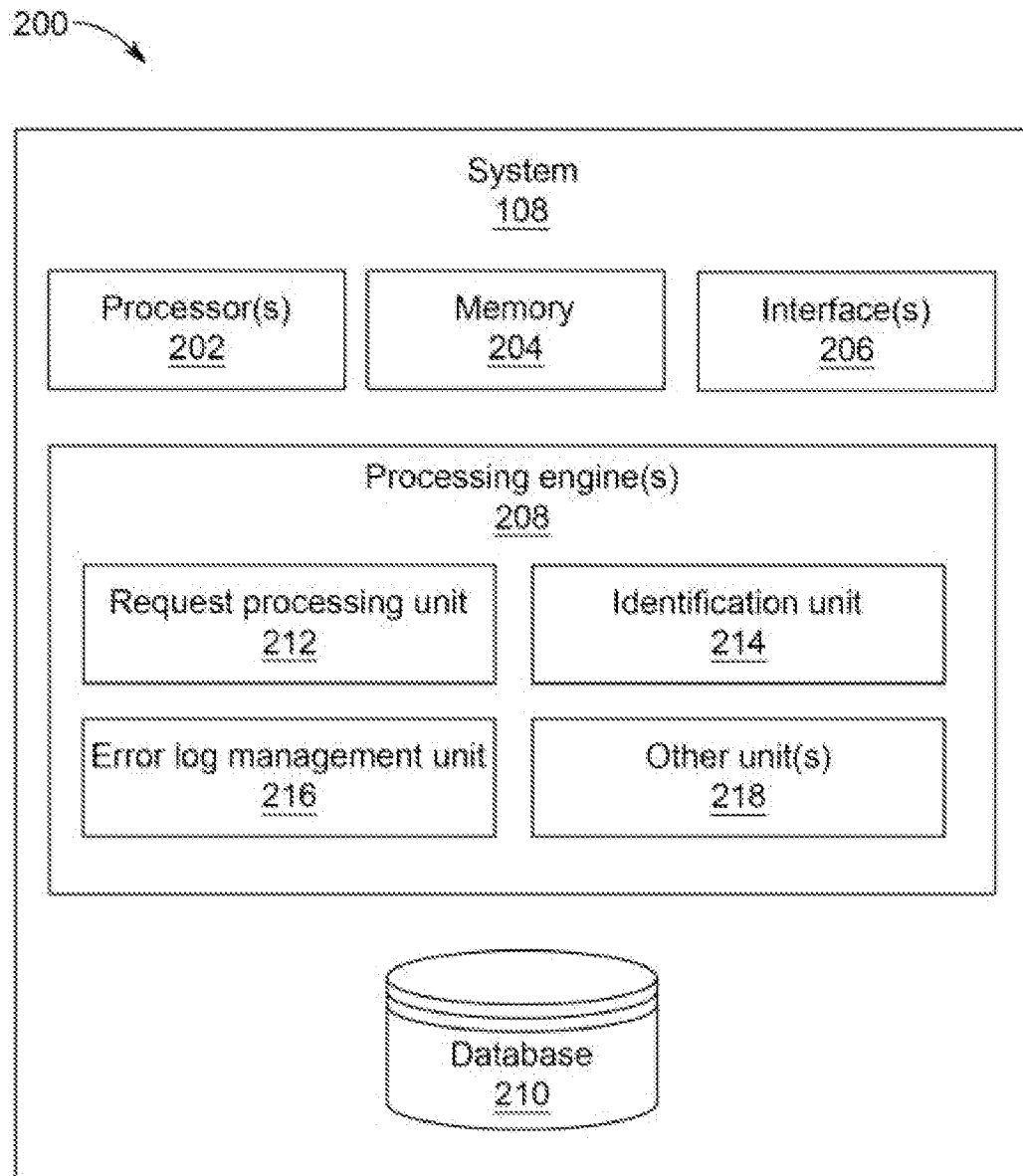


FIG. 2

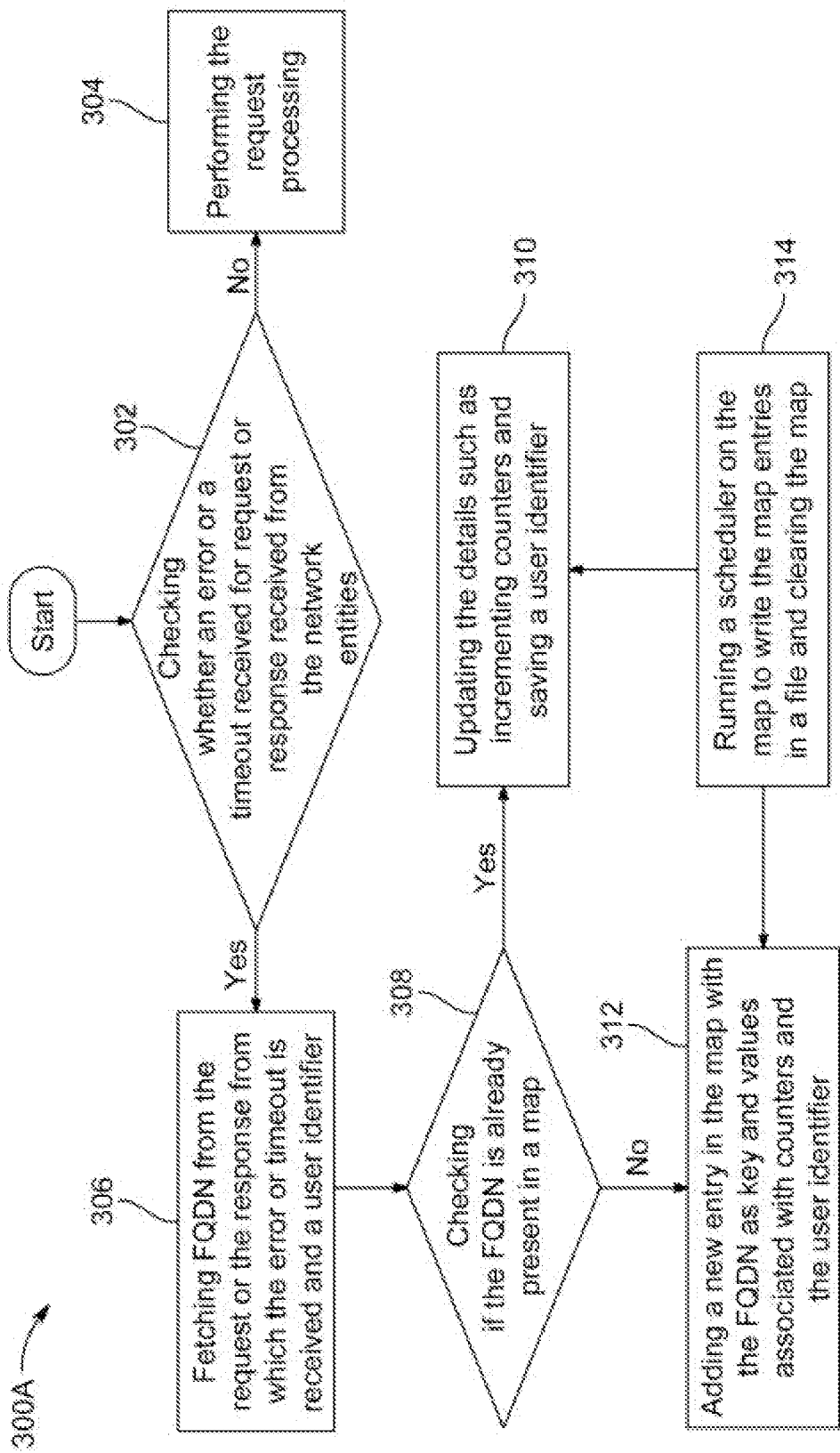


FIG. 3A

4/5

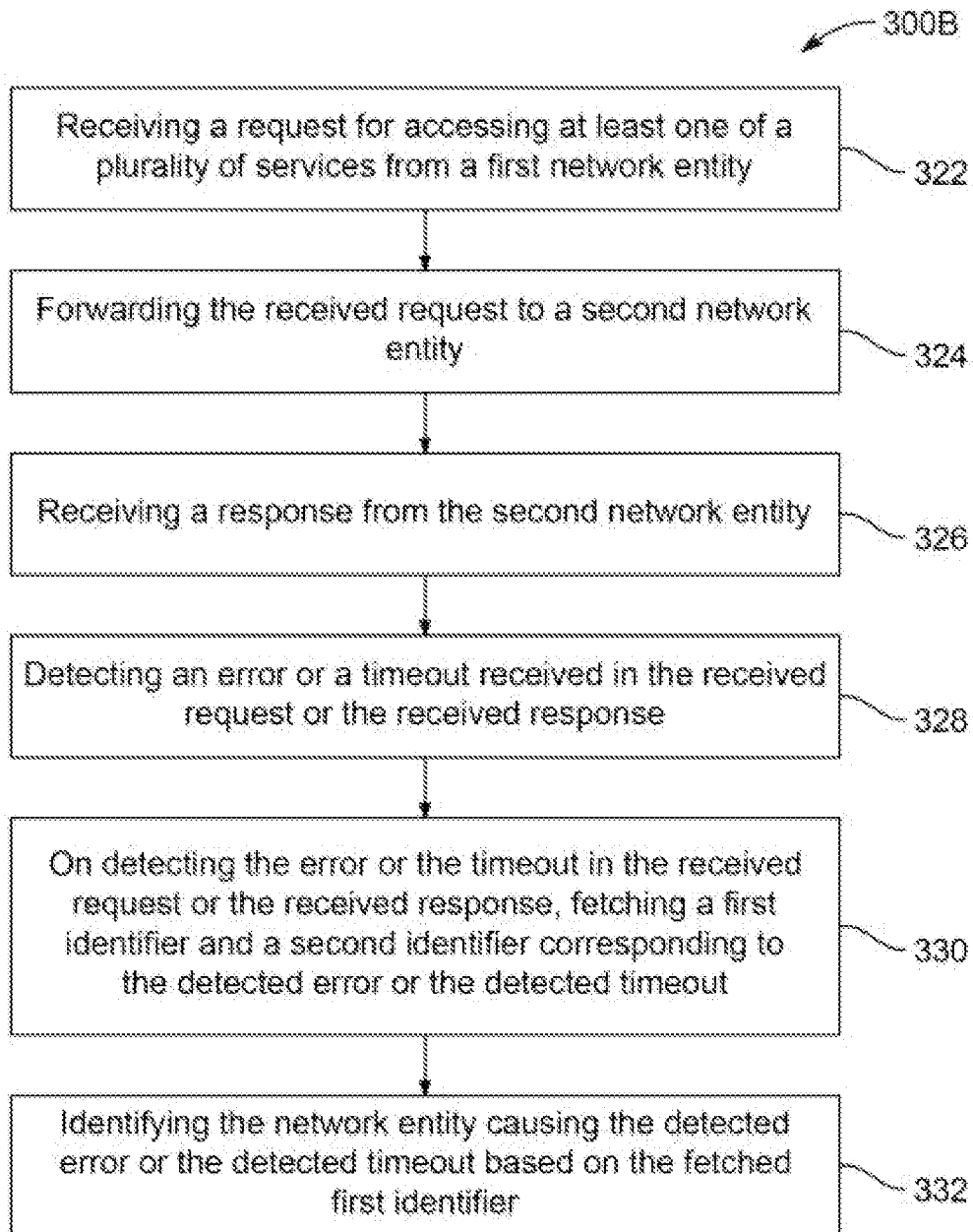


FIG. 3B

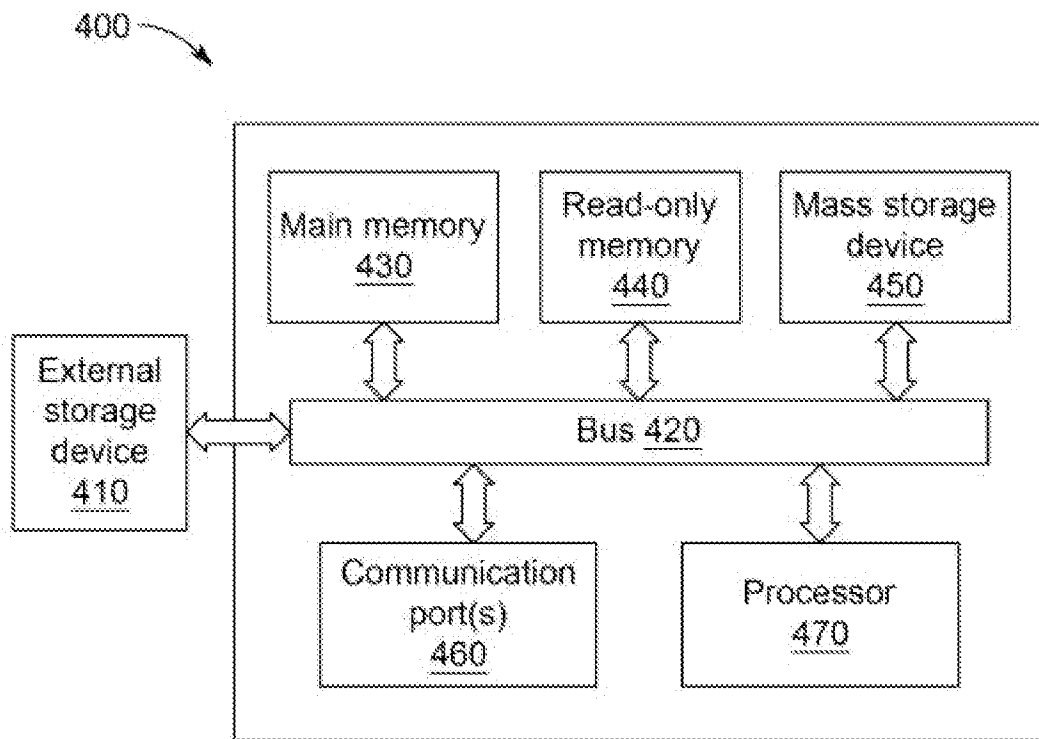


FIG. 4

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IN2024/051190

A. CLASSIFICATION OF SUBJECT MATTER
H04L41/06, H04L43/0823, G06F11/16 Version=2024.01

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, G06F

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

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

PatSeer, IPO Internal Database

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	CN110022227A (Refer to the English Translation on Esapcenet) (ALIBABA GROUP HOLDING LTD) 16 Jul 2019 (16-07-2019) abstract; claims 1, 4	1-22
Y	US8572427B1 (GOOGLE INC [US]) 29 Oct 2013 (29-10-2013) claims 1, 3; columns 1-3	1-22

☐ 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

"D" document cited by the applicant in the international application

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search

08-11-2024

Date of mailing of the international search report

08-11-2024

Name and mailing address of the ISA/

Indian Patent Office
Plot No.32, Sector 14, Dwarka, New Delhi-110075
Facsimile No.

Authorized officer

Ravi Kumar

Telephone No. +91-1125300200