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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
19 December 2024 (19.12.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/254726 A1

(51) International Patent Classification:
H04L 43/0829 (2022.01) *H04L 43/10* (2022.01)

(21) International Application Number:
PCT/CN2023/099605

(22) International Filing Date:
12 June 2023 (12.06.2023)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **MICROSOFT TECHNOLOGY LICENSING, LLC** [US/US]; One Microsoft Way, Redmond, Washington 98052 (US).

(72) Inventors: **PEPPER, Donald Eric**; One Microsoft Way, Redmond, Washington 98052 (US). **ROSOIU, Irina Andreea**; One Microsoft Way, Redmond, Washington 98052 (US). **KOTIPALLI, Srinivasachakrapani**; One Microsoft Way, Redmond, Washington 98052 (US). **RUTKOWSKI, Bradley David**; One Microsoft Way, Redmond, Washington 98052 (US). **JIN, Dengke**; One Microsoft Way, Redmond, Washington 98052 (US). **YIN, Zhaoyang**; One Microsoft Way, Redmond, Washington 98052 (US). **ZITNICK, Paula Rose**; One Microsoft Way, Redmond, Washington 98052 (US). **CUI, Pengfei**; One Microsoft Way, Redmond, Washington 98052 (US).

(74) Agent: **KING & WOOD MALLESONS**; 20th Floor, East Tower, World Financial Centre, No. 1 Dongsanhuan Zhonglu, Chaoyang District, Beijing 100020 (CN).

(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, 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,

(54) Title: DETERMINING LOW-GRADE PACKET LOSS BETWEEN COMPUTING SYSTEMS UTILIZING USER DATAGRAM PROTOCOL (UDP) ECHO MESSAGES

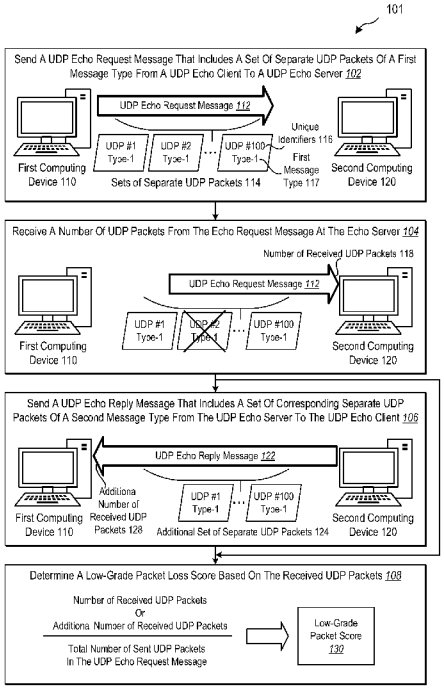

FIG. 1

(57) Abstract: This disclosure describes a connectivity measurement system that accurately and flexibly measures network connectivity between computing devices and/or systems that regularly communicate. For instance, the connectivity measurement system utilizes echo messages that include sets of separate but related UDP packets to determine the amount of low-grade packet loss between computing devices or systems. Additionally, the connectivity measurement system utilizes various actions and components to detect and determine occurrences of low-grade packet loss.

WO 2024/254726 A1

[Continued on next page]

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))*

DETERMINING LOW-GRADE PACKET LOSS BETWEEN COMPUTING SYSTEMS UTILIZING USER DATAGRAM PROTOCOL (UDP) ECHO MESSAGES

BACKGROUND

[0001] Network connectivity is a vital component of high-availability services. In distributed computing environments, where services span different servers or datacenters, network connectivity ensures seamless interaction and cohesive operation. Many computing systems utilize Internet Control Message Protocol (ICMP) ping messages to measure and test network connectivity. However, ICMP messages suffer from several limitations, such as being blocked by some systems, unsupported in certain scenarios, deprioritized compared to other network traffic types, and other issues, which make ICMP messages inaccurate and unreliable.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description provides specific and detailed implementations accompanied by drawings. Additionally, each of the figures listed below corresponds to one or more implementations discussed in this disclosure.

[0003] FIG. 1 illustrates an overview example of implementing a connectivity measurement system that determines low-grade packet loss between computing systems and/or devices by utilizing sets of user datagram protocol (UDP) packets.

[0004] FIG. 2A illustrates a system environment where a connectivity measurement system is implemented.

[0005] FIG. 2B illustrates a block diagram of an example UDP echo client device and a UDP echo server device.

[0006] FIG. 3 illustrates a block diagram of an example UDP packet sent as part of a UDP echo message.

[0007] FIG. 4 illustrates a sequence flow diagram example of determining low-grade packet loss between computing devices when UDP echo request message packets are successfully received.

[0008] FIG. 5 illustrates another sequence flow diagram example of determining low-grade packet loss between computing devices when one or more of the UDP echo request message packets are not successfully received.

[0009] FIG. 6 illustrates an example series of acts in a computer-implemented method for determining low-grade packet loss between computing systems and/or devices utilizing UDP messages.

[0010] FIG. 7 illustrates components included within an example computer system.

DETAILED DESCRIPTION

[0011] The present disclosure describes a connectivity measurement system that accurately and flexibly measures network connectivity between computing systems that regularly communicate. For instance, the connectivity measurement system utilizes echo messages that include sets of separate but related UDP packets to determine the amount of low-grade packet loss between computing devices or systems. Additionally, the connectivity measurement system utilizes various actions and components to detect and determine occurrences of low-grade packet loss.

[0012] To illustrate, consider the connectivity measurement system measuring network connectivity between two computing devices that regularly communicate to facilitate one or more network services. For example, the connectivity measurement system on the first device generates and sends an echo request message that includes a set of separate UDP packets to the

second computing device. In response, the second device receives a number (e.g., a number count) of the UDP packets of the echo request message within a message-receiving threshold. Based on the number of received UDP packets, the connectivity measurement system determines a low-grade packet loss score between the first device and the second device.

[0013] In some cases, the connectivity measurement system on the second device generates and sends an echo reply message that includes an additional set of separate UDP packets to the first device. In response, the first device receives some or all of the UDP packets of the UDP echo reply message. In these cases, the connectivity measurement system on the first device may utilize the number of additional UDP packets received to determine a low-grade packet loss score based on the number of UDP packets of the UDP echo request message (as the additional number of packets may indicate that one or more of the packets in the echo request message were lost) and/or the number of UDP packets of the UDP echo reply message.

[0014] As described in this document, the connectivity measurement system delivers several significant technical benefits in terms of computing accuracy, flexibility, and efficiency compared to existing systems. Moreover, the connectivity measurement system provides several practical applications that address problems relating to measuring network connectivity between computing devices, particularly in determining low-grade packet loss.

[0015] As mentioned above, many existing systems utilize Internet Control Message Protocol (ICMP) messages, such as ICMP echo request messages and ICMP echo reply messages, to measure and analyze network connectivity. However, using ICMP messages leads to inaccurate and unreliable measurements. In contrast, the connectivity measurement system utilizes various actions, features, and components to address the problems associated with ICMP messages, as discussed below.

[0016] By utilizing user datagram protocol (UDP) messages, the connectivity measurement system provides greater accuracy and flexibility compared to existing systems that utilize ICMP messages. For example, many environments block ICMP traffic by using firewalls to filter or restrict it out. If a receiving computing device cannot receive ICMP traffic, it cannot accurately measure network connectivity. Instead, the connectivity measurement system utilizes UDP packets, which are rarely blocked.

[0017] As another example, by utilizing UDP messages, the connectivity measurement system provides greater accuracy by simulating real-world scenarios. For instance, network connectivity measurements are used to determine whether data between separate services or devices is being successfully transmitted. Since many services do not use ICMP traffic, measuring network connectivity using ICMP packets produces inaccurate results. Instead, the connectivity measurement system uses UDP messages, which represent the same type of network traffic being sent between computing devices.

[0018] Furthermore, many systems and devices prioritize other types of network traffic over ICMP, resulting in longer response times for ICMP messages. This further diminishes the accuracy of connectivity measurements since these systems are determining evaluation metrics for network traffic types that are not deprioritized using delayed ICMP messages. Once again, by employing UDP messages, the connectivity measurement system emulates the network traffic exchanged between computing devices. Additionally, by utilizing a larger number of UDP packets for sending UDP echo messages (to match or exceed real-world data), the system achieves higher accuracy in measuring low-grade packet loss.

[0019] Moreover, the connectivity measurement system frequently generates UDP echo messages that closely resemble real-world data. For instance, the connectivity measurement

system creates UDP echo messages that include large buffer sizes of padding data that are separated across multiple separate UDP packets. By utilizing large buffer sizes of padding data, the connectivity measurement system better replicates the characteristics of actual data transmitted between the computing devices, resulting in highly precise connectivity measurements.

[0020] As mentioned earlier, the use of UDP messages enables the connectivity measurement system to more accurately represent real-world data transmission between the computing devices, particularly when measuring low-grade packet loss. Unlike TCP, which is a protocol within the TCP/IP suite and involves packet retransmission, UDP data transmission does not involve the retransmission of dropped packets. Hence, when two computing devices use UDP data, occurrences of low-grade packet loss are not concealed through packet retransmission. Thus, the utilization of UDP messages prevents the masking of low-grade packet loss incidents.

[0021] As mentioned, the connectivity measurement system generates UDP echo messages that include sets of separate yet related UDP packets. Some existing systems send echo messages with large amounts of data, which may exceed the Maximum Transmission Unit (MTU) size. In such cases, the operating system (OS) fragments the echo message into multiple packets. However, fragmented packets are often dropped or ignored by many receiving computing devices. Moreover, real-world data transmission between the computing devices is not sent as fragmented packets. To address this, the connectivity measurement system ensures that UDP echo messages do not exceed the MTU size by separating the data into non-fragmented UDP packets. It then sends the set of UDP packets as separate, non-fragmented packets to the receiving computing device. This approach enables the connectivity measurement system to accurately measure network connectivity and low-grade packet loss.

[0022] By utilizing UDP echo messages with different message types, the connectivity measurement system enhances security by mitigating malicious attacks, such as traffic flooding attacks. For example, some existing systems send a first echo request message with packet data to a second device, and the second device responds by copying the echo message and sending it back. However, if the first echo message maliciously or inadvertently includes the second device as both the sender and the recipient, the second device will mistakenly send the second echo message back to itself. Consequently, upon receiving the message, it will duplicate it and send it back to itself again, resulting in an infinite loop that can cause excessive resource consumption, instability, and unresponsiveness.

[0023] Instead, the connectivity measurement system assigns distinct message types to UDP echo request messages. The first device (e.g., a UDP echo client) exclusively sends UDP echo request messages of the first type and receives UDP echo reply messages of the second type. Similarly, the second device (e.g., a UDP echo server) solely sends UDP echo reply messages of the second type and receives UDP echo request messages of the first type. This prevents the second device from erroneously processing UDP echo reply messages intended for itself.

[0024] This disclosure uses several terms to describe the features and benefits of one or more implementations. As an example, a "UDP echo message" refers to a message transmitted using the user datagram protocol (UDP) between two computing devices. A UDP echo message includes one or more UDP data packets (or simply "UDP packets"). For example, a UDP echo message includes a set of separate, independent UDP packets. Additionally, within this disclosure, an echo message that includes one or more UDP packets is synonymous with a UDP echo message.

[0025] Additionally, a UDP echo message can include UDP echo request messages and UDP echo reply messages. As an example, a "UDP echo request message" refers to a network communication packet sent via UDP to request a response from a target device. The echo request message prompts the target device to send back an echo reply, thereby confirming reachability and responsiveness. Similarly, a "UDP echo reply message" refers to a network communication packet sent in response to a UDP echo request message acknowledging the receipt of the echo request.

[0026] As noted above, a UDP echo request message and/or a UDP echo reply message can include a set of separate UDP packets. As an example, the term "set of separate UDP packets" refers to related yet separate, non-fragmented UDP packets in a set. For instance, the connectivity measurement system transmits a UDP echo request message by generating and sending out a set of separate UDP packets at the same time from a sender device to a target device. In some cases, each packet within a set of separate UDP packets includes a unique identifier such as a globally unique identifier or a set index number. In various implementations, a set of separate UDP packets includes a predetermined number of packets. Notably, in many instances, it is the connectivity measurement system rather than an operating system (OS) that generates sets of separate UDP packets.

[0027] When sending over UDP echo messages that include a set of separate UDP packets, some or all of the UDP packets may be successfully delivered. As an example, the terms "number of received UDP packets" or "number of UDP packets received" refer to a number count or count of UDP packets in a set of separate UDP packets that are received without errors by the recipient device. In some instances, a UDP packet is successfully received if it arrives without any error within a predefined time window, such as a message-receiving threshold.

[0028] As an example, the term "low-grade packet loss" refers to the occasional or intermittent loss of data packets during transmission between computing devices, typically occurring at a relatively low rate. This type of packet loss may not have a significant impact on overall performance but can cause disruptions or degradation in network communication. As an example, the term "low-grade packet loss score" refers to a metric used to measure the occurrence of low-grade packet loss between computing devices. In some instances, a low-grade packet loss score indicates the number, amount, count, rate, and/or percentage of dropped packets between computing devices. As described further below, a low-grade packet loss score is often determined based on the number of received UDP packets associated with a UDP echo message.

[0029] As mentioned, the connectivity measurement system facilitates the transmission of UDP echo messages between computing devices. As an example, within this disclosure, the term "computing devices" refers to separate and distinct computing devices. Often, the computing devices are referred to as the first device and the second device, the sender device and the target device, or the UDP echo client and the UDP echo server, respectively. Examples of computing devices include different server devices, mobile devices, devices in different computing systems, and/or devices in different datacenters. Additionally, within this disclosure, sending communications between computing devices encompasses the exchange of communications between a service on the first device and a service on the second device.

[0030] Further details regarding an example implementation of the connectivity measurement system are discussed in connection with the following figures. For example, FIG. 1 provides an overview example of implementing the connectivity measurement system to determine low-grade packet loss between computing systems and/or devices by utilizing sets of

user datagram protocol (UDP) packets according to some implementations. As shown, FIG. 1 includes a series of acts 101 performed directly or indirectly by the connectivity measurement system. FIG. 1 also shows various elements, features, and components associated with the connectivity measurement system, including a first computing device 110 and a second computing device 120.

[0031] As previously mentioned, the connectivity measurement system determines network connectivity metrics between computing devices. In particular, the connectivity measurement system detects and measures low-grade packet loss between two devices. Hence, when a connectivity measurement is needed, the connectivity measurement system may perform the series of acts 101 to determine a low-grade packet loss score between the two computing devices.

[0032] To provide an illustration, the series of acts 101 in FIG. 1 includes the act 102 of sending a UDP echo request message that includes a set of separate UDP packets of a first message type from a UDP echo client to a UDP echo server. For example, when a trigger prompts a network connectivity measurement, an instance of the connectivity measurement system on the first computing device 110, which includes a UDP echo client, initiates the measurement by generating a UDP echo request message 112 to be sent to the second computing device 120 hosting a UDP echo server.

[0033] As shown in connection with the act 102, the UDP echo request message 112 includes a set of separate UDP packets 114. Moreover, each UDP packet in the set of separate UDP packets 114 is assigned a unique identifier 116 and a first message type 117 (e.g., a UDP echo request message type). Additionally, FIG. 1 shows that the set of separate UDP packets 114 includes one- hundred non-fragmented UDP packets in the set of separate UDP packets 114. In various implementations, these UDP packets within the set of separate UDP packets 114 include

header information and padding data. An example of a UDP packet is provided below in connection with FIG. 3.

[0034] As shown, the series of acts 101 includes the act 104 of receiving a number of UDP packets from the echo request message at the echo server. For example, the UDP echo server on the second computing device 120 receives a number of received UDP packets 118 of the UDP echo request message 112. In some cases, the UDP echo server successfully receives all of the UDP packets. However, in other scenarios, certain UDP packets may be lost (e.g., dropped), and the number of received UDP packets 118 indicates that one or more UDP packets were not received by the UDP echo server. In some implementations, UDP packets not received within a specified time window (e.g., a message-receiving threshold) are considered lost packets. Further details regarding receiving some or all of the UDP packets of a UDP echo request message are provided below in connection with FIG. 4 and FIG. 5.

[0035] Once the second computing device 120 receives the UDP packets from the first computing device 110, it can send a reply message. For example, the act 106 in the series of acts 101 includes sending a UDP echo reply message that includes a set of corresponding separate UDP packets of a second message type from the UDP echo server to the UDP client.

[0036] In some implementations, an instance of the connectivity measurement system on the second computing device 120 generates a UDP echo reply message 122 that includes an additional set of separate UDP packets 124 corresponding to the UDP packets received at the UDP echo server for the UDP echo request message 112. For instance, if 100 UDP packets were received, the connectivity measurement system generates 100 UDP packets with corresponding identifiers and/or padding data. Similarly, if 98 packets were received, the connectivity measurement system generates 98 corresponding packets. Additional details regarding the

generation of UDP echo reply messages with sets of separate UDP packets are provided below in connection with FIG. 4 and FIG. 5.

[0037] Further, as shown in the act 106, the first computing device 110 receives an additional number of received UDP packets 128. The additional number (e.g., a number count) of received UDP packets 128 may include some or all of the additional set of separate UDP packets 124 depending on whether any of the UDP packets in the UDP echo reply message 122 were lost during the return trip.

[0038] As shown, the series of acts 101 includes the act 108 of determining a low-grade packet loss score based on the received UDP packets. For example, the connectivity measurement system compares the number of received UDP packets 118 and/or the additional number of received UDP packets 128 to the total number of sent UDP packets in the UDP echo request message 112 (or the UDP echo reply message 122) to determine a low-grade packet loss score 130. Subsequently, the connectivity measurement system may report the low-grade packet loss score and/or implement one or more changes based on the low-grade packet loss score 130. Additional details regarding the generation and utilization of a low-grade packet loss score are provided below in connection with FIG. 4 and FIG. 5.

[0039] In some instances, upon the second computing device 120 receiving UDP packets from the first computing device 110 (e.g., the act 104), the second computing device 120 determines a low-grade packet loss score based on the received UDP packets (e.g., the act 108). In other words, the series of acts 101 proceeds directly from the act 104 to the act 108, skipping the act 106. In these instances, the second computing device 120 may provide a UDP echo reply message 122 to the first computing device 110 or a third computing device indicating the low-grade packet loss score 130.

[0040] With a general overview of the connectivity measurement system in place, additional details are provided regarding the components, features, and elements of the connectivity measurement system. To illustrate, FIGS. 2A–2B show an example system environment where a connectivity measurement system is implemented. In particular, FIG. 2A illustrates an example system environment that includes the connectivity measurement system on a UDP echo client device and a UDP echo server device, and FIG. 2B illustrates a block diagram providing additional details of the UDP echo client device and the UDP echo server device. While FIGS. 2A–2B show example arrangements and configurations of the connectivity measurement system and associated components, other arrangements and configurations are possible.

[0041] As shown, FIG. 2A includes a computing environment 200 of a computing system having a first datacenter 202 and a second datacenter 212, which are connected by a network 240. In various implementations, the computing environment 200 represents a distributed computing environment, such as datacenters located in different geographical regions. Further details regarding these and other computing devices are provided below in connection with FIG. 7. In addition, FIG. 7 also provides additional details regarding networks, such as the network 240 shown.

[0042] As shown, the first datacenter 202 includes a first server device 204 (i.e., a first computing device) having a first service 205 and a security manager 208. Additionally, the first datacenter 202 includes a UDP echo client 210 having the connectivity measurement system 206. The first service 205, the security manager 208, and/or the UDP echo client 210 may be on the same or different computing devices within the first datacenter 202. FIG. 2A also shows the second datacenter 212, which includes a second server device 214 having a second service 215

and a security manager 218. The second datacenter 212 also includes a UDP echo server 220 having the connectivity measurement system 206.

[0043] As shown, the first datacenter 202 and the second datacenter 212 each include the connectivity measurement system 206. For example, the first datacenter 202 includes a first instance of the connectivity measurement system 206 within the UDP echo client 210 that includes some or all of the functions of the connectivity measurement system 206. Similarly, the second datacenter 212 includes a second instance of the connectivity measurement system 206 within the UDP echo server 220 that includes some or all of the functions of the connectivity measurement system 206. In some implementations, the connectivity measurement system 206 (or another instance of the connectivity measurement system 206) is located elsewhere in the first datacenter 202 and/or the second datacenter 212. Further, in various implementations, the connectivity measurement system 206 is located at only one of the datacenters or computing devices (e.g., it is at the first dataenter 202 and provides instructions to the second datacenter 212 to perform operations there on behalf of the connectivity measurement system 206).

[0044] In the computing environment 200, the first datacenter 202 can send data to the second datacenter 212. For example, the second service 215 is a crucial high-availability service that receives data from the first service 205. Accordingly, the connectivity measurement system 206 performs regular network connectivity measurements and evaluations to ensure the successful transfer of data. Additionally, in many implementations, the first datacenter 202 and the second datacenter 212 utilize their respective security managers to establish a secure connection and/or secure communication channel for data sent from the first datacenter 202 to the second datacenter 212 (and vice versa), including UDP echo messages. For example, the

security managers facilitate firewall and/or port protection preventing unauthorized communications between the first datacenter 202 and the second datacenter 212.

[0045] In various implementations, to initiate a network connection measurement, the connectivity measurement system 206 associated with the UDP echo client 210 generates and sends a UDP echo request message with a set of separate user datagram protocol (UDP) packets to the UDP echo server 220. The connectivity measurement system 206 associated with the UDP echo server 220 determines a low-grade packet loss score from the received UDP packets and/or sends a UDP echo reply message having a set of separate corresponding UDP packets to the UDP echo client 210, and the connectivity measurement system 206 associated with the UDP echo client 210 determines the low-grade packet loss score. Additional details regarding this process are provided below in connection with FIG. 4 and FIG. 5.

[0046] As mentioned earlier, FIG. 2B illustrates a block diagram that provides additional details of example UDP echo client and UDP echo server devices. As shown, FIG. 2B includes the UDP echo client 210 and the second datacenter 212, each having an instance of the connectivity measurement system 206. In some implementations, the connectivity measurement system 206 is only located on the UDP echo client 210 or the UDP echo server 220, as mentioned earlier. For ease of explanation, the two instances of the connectivity measurement system 206 include some different components. However, implementations of the connectivity measurement system 206 can include some or all of the components described in connection with FIG. 2B.

[0047] As shown, the connectivity measurement system 206 associated with the UDP echo client 210 includes an echo request message manager 222 that manages UDP echo request messages (i.e., UDP echo messages 232) having sets of UDP packets (e.g., UDP packets 234).

As shown, the echo request message manager 222 includes a first message type generator 224 that generates UDP echo request messages having UDP packets assigned to a first message type (e.g., a UDP echo request message type) indicating that the UDP packets belong to a UDP echo request message.

[0048] The connectivity measurement system 206 associated with the UDP echo client 210 also includes a second message type processor 226 (e.g., a UDP echo reply message type) that processes UDP echo reply messages having UDP packets assigned to the second message type, which are received by the UDP echo client 210. In this way, the connectivity measurement system 206 protects the UDP echo client 210 from improperly processing UDP echo request messages.

[0049] Additionally, the connectivity measurement system 206 associated with the UDP echo client 210 includes a low-grade packet loss manager 228 that determines low-grade packet loss and/or a low-grade packet loss score between the UDP echo client 210 and the UDP echo server 220. Further, the connectivity measurement system 206 includes a storage manager client device 230 having the UDP echo messages 232 and the UDP packets 234. In various implementations, the storage manager client device 230 includes additional and/or different data utilized by the connectivity measurement system 206.

[0050] Regarding the connectivity measurement system 206 associated with the UDP echo server 220, FIG. 2B shows that this system includes an echo reply message manager 242, which manages UDP echo reply messages (i.e., UDP echo messages 232) having sets of UDP packets (e.g., UDP packets 234) that correspond to UDP packets in a UDP echo request message. Additionally, the echo reply message manager 242 includes a second message type generator 244 that generates UDP echo reply messages having UDP packets assigned to the second message

type (e.g., a UDP echo reply message type) indicating that the UDP packets belong to a UDP echo reply message.

[0051] The connectivity measurement system 206 associated with the UDP echo server 220 also includes a first message type processor 246 that processes UDP echo request messages having UDP packets assigned to the first message type received by the UDP echo server 220. Again, this measure protects the UDP echo server 220 from improperly processing UDP echo reply messages and prevents the generation and processing of an infinite loop of packets.

[0052] Additionally, the connectivity measurement system 206 associated with the UDP echo server 220 includes the low-grade packet loss manager 228 and the storage manager client device 230 previously described. In some implementations, the storage manager client device 230 on the UDP echo server 220 includes different instances of UDP echo messages 232, UDP packets 234, and/or other data than the data stored on the UDP echo client 210.

[0053] As mentioned above, FIG. 3 provides additional details regarding a UDP packet 300. In particular, FIG. 3 illustrates a block diagram of an example user datagram protocol (UDP) packet sent as part of a UDP echo message according to some implementations. In various implementations, the UDP packet 300 represents one packet in a set of UDP packets making up a UDP echo message (e.g., a UDP echo request message or a UDP echo reply message). Further, in various implementations, the packets within the set of UDP packets are sent between UDP echo devices at the same time, which allows for a better test of low-grade packet loss.

[0054] The UDP packet 300 is a vehicle to transmit data between the two computing devices. The UDP packet 300 can vary in size based on other network protocols being used. For example, internet protocol (IP) packets are typically 65,535 bytes at a maximum. However, most networks are limited by a maximum transmission unit (MTU) packet size, which is often around 1,500

bytes. Packets with a buffer size larger than the MTU size are either fragmented (e.g., IPv4) or dropped (e.g., IPv6). Accordingly, in many implementations, the UDP packet 300 is part of a packet that is equal to (or less than) the MTU used by the computing devices.

[0055] In various implementations, the connectivity measurement system 206 generates UDP packets as part of a UDP echo request message or a UDP echo reply message. In particular, in many instances, the connectivity measurement system 206 utilizes a UDP echo client to send UDP packets as part of a UDP echo request message and/or a UDP echo server to send UDP packets as part of a UDP echo reply message.

[0056] Returning to FIG. 3, the UDP packet 300 includes a UDP header portion 302 and a UDP data portion 304. The UDP header portion 302 can include bytes of data for directing the UDP packet 300 to the destination. For instance, the UDP header portion 302 includes source and destination information (including port information). In some instances, the UDP header is 8 bytes in length. Additionally, in various implementations, the UDP packet 300 is part of a larger IP packet that includes its own header (e.g., a 20-byte IP header).

[0057] The UDP data portion 304 can include various pieces of data. For example, FIG. 3 shows the UDP data portion 304 includes data corresponding to a message type 306, a packet identifier 308, a packet count total 310, and padding data 312. In many instances, the message type 306, the packet identifier 308, and the packet count total 310 are each 1 byte in size while the padding data 312 can fill up the remaining space in the UDP packet 300.

[0058] In various implementations, the message type 306 indicates the type of message that the UDP packet 300 is. For example, the message type 306 indicates that the UDP packet 300 is a UDP echo request message (e.g., a UDP echo request message type) or a UDP echo reply message (e.g., a UDP echo reply message type). In various instances, the message type 306 is a

binary value (e.g., 0 or 1), different numerical values (e.g., 0, 2), or a letter (Q, P) to indicate the two types of message types. In some implementations, the act 106 includes additional message types.

[0059] As mentioned above, the connectivity measurement system 206 utilizes the message type 306 to safeguard against malicious actors or other transmission errors where a UDP echo client and the UDP echo server may become stuck in an infinite loop. In particular, the message type 306 prevents the scenario where the UDP echo client or the UDP echo server sends UDP packets to itself. For instance, the UDP echo server ensures that incoming UDP packets are part of a UDP echo request message before processing them.

[0060] In some implementations, the packet identifier 308 uniquely identifies a UDP packet from other packets within a set of separate UDP packets. In some instances, the packet identifier 308 is a globally identifiable (GUID) that identifies a packet across multiple sets of UDP packets. In some instances, the packet identifier 308 identifies a packet within a set of UDP packets, such as an index number of 1–100 in a set of 100 UDP packets. In some implementations, a set of UDP packets has its own identifier (e.g., a UDP echo message identifier) that distinguishes it from other UDP echo messages. In some implementations, UDP packets do not include the packet identifier 308.

[0061] In various implementations, the UDP packet 300 includes the packet count total 310. For example, one packet (or a few packets for redundancy) in a UDP packet set includes a packet count total that indicates the number of packets in the set. For example, for a set of 100 packets, one or more of the packets includes a byte that indicates 100 total packets in the set. In some implementations, each packet in a set includes the packet count total 310.

[0062] In some implementations, the UDP packet 300 does not include a packet count total 310. For example, the UDP echo client and the UDP echo server have previously established a total packet count for UDP echo messages. In these implementations, the UDP packets can include the packet identifier 308 but not the packet count total 310, which has been previously agreed upon.

[0063] In many implementations, the UDP packet 300 includes padding data 312. In various instances, the padding data 312 includes randomly generated data according to the buffer size. For example, a UDP echo message includes a buffer size that is primarily padding data separated between or distributed among non-fragmented UDP packets in a set. In many instances, the buffer size corresponds to the maximum buffer size for a given packet protocol (e.g., 65,535 bytes for IP packets). In many cases, the padding data is different for each packet within a set of UDP packets of a UDP echo message.

[0064] In general, for UDP echo messages, the buffer size of padding data is much larger than the MTU size so multiple packets are needed to transmit the message between computing devices. Accordingly, the connectivity measurement system 206 can utilize padding data to improve the accuracy of measuring network connectivity. For example, if a UDP echo message includes a large and significant amount of padding data divided among a set of UDP packets, then the connectivity measurement system 206 can more accurately test and measure for low-grade packet loss (as low-grade packet loss is difficult to detect when only a few packets are sent). Another example is that the buffer size of a UDP echo message can match the size of the real-world data being transmitted between the two computing devices to yield more accurate and precise results.

[0065] In various implementations, the connectivity measurement system 206 fills up available space in UDP packets using the padding data 312 up to the MTU packet size. Because the padding data 312 itself is unimportant, the connectivity measurement system 206 can insert a break in the padding data from the buffer size at the end of a UDP packet and continue the padding data with the next UDP packet in a set until that packet is filled. In some instances, the connectivity measurement system 206 continues to generate separate, non-fragmented UDP packets in a set until the buffer size is exhausted and/or a packet count threshold is reached (e.g., a maximum of 40, 45, 70, or 100 packets).

[0066] While the connectivity measurement system 206 sends UDP echo messages that include sets of UDP packets packed with the padding data 312, in some implementations, the connectivity measurement system 206 uses smaller amounts of padding data (or no padding data). For instance, the connectivity measurement system 206 generates sets of UDP packets to match the packet sizes of real-world data being transmitted (matching both the total message size and individual packet size), which may include full or partial packets. This approach allows the connectivity measurement system 206 to capture results that more accurately indicate low-grade packet loss among the real-world data being transmitted.

[0067] In various implementations, the UDP packet 300 also includes data to perform checksum or other data confirmations. In some instances, the connectivity measurement system 206 relies on an application or higher-level protocols to perform checksums or forward error connection (FEC) procedures.

[0068] As mentioned earlier, utilizing UDP packets instead of ICMP packets offers several benefits. For example, compared to ICMP, UDP traffic is not deprioritized, dropped, or ignored as it is one of the data protocols being used by the two computing devices transmitting the data

for which the network connectivity is being measured. Consequently, utilizing UDP packets within UDP echo messages provides reliable network connectivity measurement conditions.

[0069] With the foundation in place for the connectivity measurement system 206 and UDP echo messages having sets of UDP packets, additional details will now be described regarding various functions of the connectivity measurement system 206. As mentioned earlier, FIG. 4 and FIG. 5 provide additional details of determining low-grade packet loss between computing devices based on receiving all or some of the UDP echo request message packets.

[0070] As shown, FIG. 4 and FIG. 5 include a UDP echo client 210 and a UDP echo server 220 that communicate with each other. While not shown, the UDP echo client 210 is located on a first computing device and the UDP echo server 220 is located on a second computing device. Additionally, the first and second computing devices may be part of a distributed computing environment located in different geographical regions.

[0071] For example, the UDP echo client 210 is located on a computing device and/or datacenter that sends data to a second computing device and/or datacenter where the UDP echo server 220 is located. As described above, in many instances, the UDP echo client 210 is associated with one instance of the connectivity measurement system 206, and/or the UDP echo server 220 is associated with another instance of the connectivity measurement system 206. As a result, the connectivity measurement system 206 monitors and determines network connectivity measurements between the two computing devices, including determining amounts of low-grade packet loss.

[0072] To further illustrate, consider that FIG. 4 shows a sequence flow diagram example of determining low-grade packet loss between computing devices when UDP echo request message packets are successfully received according to some implementations. In particular, FIG. 4

shows a series of acts performed by the connectivity measurement system 206 via the UDP echo client 210 and/or the UDP echo server 220.

[0073] FIG. 4 includes the act 402 of the UDP echo client 210 determining to send an echo request message to the UDP echo server 220. For example, the connectivity measurement system 206 determines to send a UDP echo request message as part of a regular measurement check between the computing devices. In some instances, a change in network conditions triggers the connectivity measurement system 206 to initiate the transmissions of a UDP echo message to the UDP echo server 220 for measuring network conditions. In various implementations, the connectivity measurement system 206 determines a low-grade packet loss score just before or after sending real-world data between the computing devices.

[0074] As mentioned above, sending a UDP echo request message includes sending a buffer size of padding data, which is often larger than the MTU size for packets. Accordingly, as shown, the UDP echo client 210 performs the act 404 of generating a set of separate UDP packets of the first message type.

[0075] To elaborate, the connectivity measurement system 206 sends large-sized UDP echo request messages to more accurately test the network connection at a highly-granular level by detecting low-grade packet loss. Additionally, in many instances, the connectivity measurement system 206 sends UDP echo request messages that match or exceed the size of the real-world data being transmitted between the computing devices.

[0076] As mentioned, the act 404 includes the connectivity measurement system 206 generating a set of separate UDP packets of the first message type. Because the UDP echo request message includes a buffer size of padding data that is larger than the MTU size of the network, the padding data needs to be divided into multiple packets. For example, with existing

systems, if a message is too big to be sent in a single packet, an operating system (OS) will fragment the message into multiple packets. Unfortunately, many systems do not support receiving fragmented packets and will drop or ignore them.

[0077] Instead of allowing the OS to process the UDP echo request message that is larger than the MTU, the connectivity measurement system 206 generates a set of separate, non-fragmented UDP packets. For instance, the connectivity measurement system 206 on the UDP echo client 210 divides the buffer size of padding data into portions smaller than the MTU and generates separate, independent, non-fragmented UDP packets. These related UDP packets are then grouped into a set that forms the UDP echo request message. In addition, in various instances, the connectivity measurement system 206 does not include a fragmented indication and/or indicates a "do not fragment" indicator.

[0078] In addition, the connectivity measurement system 206 assigns the UDP packets in the set of UDP packets of the UDP echo request message to a first message type indicating that the message is a UDP echo request message, as described earlier. For instance, as mentioned above, in generating the UDP packets in the set of UDP packets, the connectivity measurement system 206 includes a UDP header section and a data section, which includes the message type.

[0079] In some instances, the data section includes a packet identifier. For example, the connectivity measurement system 206 assigns each UDP packet in the set of separate UDP packets of a UDP echo request message with a unique identifier. For instance, for a set of 100 UDP packets, the connectivity measurement system 206 assigns the packets with identifiers from 0–99 or 1–100. In various implementations, the connectivity measurement system 206 assigns UDP packets with a GUID.

[0080] Additionally, in various implementations, the data section includes a packet count total, as described above. For example, one or more of the UDP packets in a set indicates the packet count total or the total number of packets in the UDP echo request message. In some implementations, the UDP echo client 210 and the UDP echo server 220 predetermine the packet count total of a UDP echo request message.

[0081] Further, the UDP packets include the padding data. As mentioned above, in one or more implementations, the connectivity measurement system 206 divides the padding data of the buffer size into separate UDP packets. In some instances, the connectivity measurement system 206 divides the buffer size to pad each UDP packet in the set of separate UDP packets with padding data until it reaches a predetermined size or until a threshold number of UDP packets is met (e.g., 100 packets).

[0082] In some implementations, the connectivity measurement system 206 fills up each UDP packet in the set of separate UDP packets with a smaller amount of padding data. For example, the connectivity measurement system 206 generates UDP packets that are smaller than the MTU. In some implementations, the connectivity measurement system 206 may still generate a large number of UDP packets in the set (e.g., 100 packets). In a few implementations, the connectivity measurement system 206 does not include padding data. For example, the connectivity measurement system 206 generates a set of separate UDP packets without padding data. This may be beneficial if the real-world data also generates UDP packets that are smaller than the MTU size or include little to no data.

[0083] Upon generating the UDP echo reply message 122 as the set of separate UDP packets having the first message types (e.g., a UDP echo request message type), the UDP echo client 210 sends the UDP echo request message packets to the UDP echo server 220, as shown in the act

406. In various implementations, the connectivity measurement system 206 sends the set of UDP packets as separate, individual UDP packets while also sending them concurrently, at the same time, and/or as quickly as possible (including sequential parallel transmissions). In various implementations, the connectivity measurement system 206 sends the transmissions within a message-sending threshold. Additionally, the echo request message packets may be sent in order by identifier or randomly.

[0084] As shown in the act 408, the UDP echo server 220 receives each of the echo request message packets of the first message type with a message-receiving threshold. For example, the UDP echo server 220 receives all of the UDP packets making up the UDP echo request message in the set of separate UDP packets with no dropped or lost packets. Because the UDP echo request message is sent as separate packets, the UDP echo server 220 individually receives each packet as a non-fragmented packet.

[0085] Further, the UDP echo server 220 receives the UDP packets within a message-receiving threshold, such as within 1 second, 5 seconds, or 10 seconds. In some implementations, the message-receiving threshold is based on the packet count total in the set of separate UDP packets. For instance, a set of 5 UDP packets may have a shorter message-receiving threshold than a set of 50 or 100 UDP packets. If the UDP echo server 220 receives a UDP packet after the message-receiving threshold, it may time out and not receive the packet.

[0086] As mentioned, the UDP echo server 220 receives packets of the first message type. For example, the UDP echo server 220 may refuse or drop UDP packets that do not have a message type indicating that the packet is part of a UDP echo request message. In this way, the connectivity measurement system 206 prevents the UDP echo server 220 from processing UDP

packets that do not have a message type or UDP packets that have the incorrect message type (such as packets generated by the UDP echo server 220).

[0087] In some instances, in response to receiving the UDP echo request message, the connectivity measurement system 206 can determine a low-grade packet loss score. For example, the act 410 shows the connectivity measurement system 206 on the UDP echo server 220 determining a low-grade packet loss score based on the UDP echo request message. The connectivity measurement system 206 can additionally or alternatively determine the low-grade packet loss score by the UDP echo client 210, as shown in the act 418. In some implementations, the connectivity measurement system 206 determines the low-grade packet loss score on another device within the first datacenter, the second datacenters, or elsewhere.

[0088] As shown, in some instances, the connectivity measurement system 206 associated with the UDP echo server 220 determines a low-grade packet loss score. In many implementations, the connectivity measurement system 206 determines a low-grade packet loss score based on the number of UDP packets received by the UDP echo server 220. For example, the connectivity measurement system 206 determines the low-grade packet loss by comparing the number of received UDP packets for the UDP echo request message with the packet count total.

[0089] In one or more implementations, the connectivity measurement system 206 determines a low-grade packet loss score of zero (i.e., zero or no low-grade packet loss) when the number of received UDP packets of the UDP echo request message matches the packet count total. A low-grade packet loss score of zero indicates no lost packets as the UDP packets of the UDP echo request message from the UDP echo client 210 were successfully received at the UDP echo server 220. Conversely, the connectivity measurement system 206 determines a low-grade

packet loss score above zero when one or more of the UDP packets of the UDP echo request message are lost, which is further described below in connection with FIG. 5.

[0090] In some instances, the connectivity measurement system 206 determines a low-grade packet loss score above zero even when all of the UDP packets are received at the UDP echo server 220. For instance, if some of the packets in the set of separate UDP packets making up the UDP echo request message take longer to arrive (e.g., after a first threshold but before the message-receiving threshold), the connectivity measurement system 206 determines a low-grade packet loss score above zero.

[0091] Similarly, in addition to the number of received UDP packets and packet count total, the connectivity measurement system 206 utilizes additional factors to determine the low-grade packet loss score. For example, the connectivity measurement system 206 utilizes an algorithm that includes factors such as the physical/geographical length between the UDP echo client 210 and the UDP echo server 220, the time of day the UDP echo request message is being sent, the level of network congestion at the time of transmission, the buffer size of the UDP echo message, and/or the message type. Accordingly, in some implementations, when using one of these factors, the connectivity measurement system 206 may determine to lower the low-grade packet loss score above when one or more UDP packets are lost.

[0092] As mentioned above, utilizing a larger buffer size of padding data leads to a higher number of UDP packets that are included in a set of separate UDP packets making up a UDP echo message. With the higher number of packets, which often match the MTU packet size, the connectivity measurement system 206 is able to more accurately detect and analyze low-grade packet loss in a network environment. This is particularly beneficial with systems where ICMP and fragmented packets are not well supported.

[0093] As shown in FIG. 4, the act 412 includes the UDP echo server 220 generating an echo reply message that includes a corresponding set of separate UDP packets of a second message type. For example, the connectivity measurement system 206 associated with the UDP echo server 220 generates the UDP echo reply message in response to receiving the UDP echo request message. Notably, the UDP echo reply message is assigned a different message type to signal that it is a reply message and not a request message.

[0094] In various implementations, the UDP echo reply message mirrors the UDP echo request message as it was received at the UDP echo server 220 back to the UDP echo client. For example, when all of the UDP packets in the set of separate UDP packets of the UDP echo request message are received by the UDP echo server 220, the UDP echo server 220 (i.e., the connectivity measurement system 206) generates a UDP echo reply message having the same buffer size, which leads to the same number of UDP packets making up the UDP echo reply message. In generating the UDP echo reply message, the connectivity measurement system 206 associated with the UDP echo server 220 generates an additional set of separate UDP packets that include non-fragmented packets.

[0095] In some implementations, the UDP echo server 220 generates a UDP echo reply message having the same padding data as received in the UDP echo request message. For example, in one or more implementations, the connectivity measurement system 206 generates the UDP echo reply message by duplicating the set of separate UDP packets received at the second computing device and changing the UDP echo request message type to a UDP echo reply message type. In these implementations, the connectivity measurement system 206 may duplicate the same identifiers for corresponding packets (e.g., reuse the assigned identifiers) or assign new unique identifiers.

[0096] In one or more implementations, the connectivity measurement system 206 generates a corresponding number of UDP packets but includes different padding data. In some implementations, the connectivity measurement system 206 utilizes the same buffer size and padding data, but the data is divided differently among the additional set of separate UDP packets.

[0097] Upon generating the UDP echo reply message represented by the set of corresponding UDP packets, the UDP echo server 220 sends the echo reply message packets to the UDP echo client 210, as shown by the act 414. For example, the UDP echo server 220 transmits the UDP echo reply message as separate, individual UDP packets while sending them as close together as possible.

[0098] As noted above, the UDP echo server 220 sends the UDP echo reply message as a second message type, which indicates that it is a reply message. If the message was mistakenly or maliciously received again by the UDP echo server 220, it disregards the message because it has the wrong message type. This way, the UDP echo server 220 does not fall into the trap of sending messages in an infinite loop to itself.

[0099] FIG. 4 also shows the act 416 of the UDP echo client 210 receiving the echo reply message of the second message type. Like the UDP echo server 220, the UDP echo client 210 determines that the UDP echo reply message is properly assigned to the second message type indicating a reply message before further processing the message. Otherwise, the UDP echo client 210 can ignore or disregard the message.

[0100] As shown, the act 418 shows the UDP echo client 210 determining a low-grade packet loss score based on the received packets. In various implementations, the connectivity measurement system 206 associated with the UDP echo client 210 determines the low-grade

packet loss score by comparing the number of UDP packets received in the UDP echo reply message with the packet count total, as described above. For example, the connectivity measurement system 206 compares the number of received UDP packets with the packet count total to determine whether any packets were lost or dropped.

[0101] In various implementations, the packet count total is the number of packets the UDP echo client 210 sent to the UDP echo server 220 in the UDP echo request message. In some implementations, the packet count total is the total number of packets the UDP echo server 220 sent back in the UDP echo reply message. In one or more implementations, the connectivity measurement system 206 determines the low-grade packet loss score based on, at least in part, comparing the packet count total of the UDP echo request message with the packet count total of the UDP echo reply message. For instance, if the packet count totals match, the connectivity measurement system 206 determines a low-grade packet loss score of zero even if less than all of the UDP packets were received back in the UDP echo reply message as all of the UDP packets were successfully received at the UDP echo server 220. In some implementations, a mismatch in packet count totals signals some amount of low-grade packet loss.

[0102] The low-grade packet loss score may represent different measurements depending on which network connections the connectivity measurement system 206 is evaluating. For instance, if the real-world data is being transmitted one way, the connectivity measurement system 206 may be less concerned with the return-trip low-grade packet loss or the total round-trip low-grade packet loss. Conversely, for real-world data being sent and returned, the connectivity measurement system 206 determines separate, additional, and/or combined low-grade packet loss scores.

[0103] Accordingly, in some implementations, the connectivity measurement system 206 determines a first or request low-grade packet loss score indicating low-grade packet loss from the UDP echo client 210 to the UDP echo server 220 based on the UDP echo request message. The connectivity measurement system 206 also determines a second or reply low-grade packet loss score indicating low-grade packet loss from the UDP echo server 220 to the UDP echo client 210 based on the UDP echo reply message. In additional implementations, the connectivity measurement system 206 determines a combined low-grade packet loss score for the round trip based on the UDP echo request message, the UDP echo reply message (e.g., average the two or take the higher of the two scores), and/or combined data (e.g., the total round-trip time (RTT) of the two UDP echo messages).

[0104] Depending on the low-grade packet loss score being determined, the connectivity measurement system 206 may utilize the different numbers of received UDP packets and/or the different packet count totals. In some implementations, the connectivity measurement system 206 may determine the scores at different computing devices. For example, the connectivity measurement system 206 associated with the UDP echo server 220 determines the request low-grade packet loss score, and the connectivity measurement system 206 associated with the UDP echo client 210 determines the reply low-grade packet loss score, which may be different if packets were only lost on the return trip. In some implementations, the connectivity measurement system 206 associated with the UDP echo client 210 determines the request low-grade packet loss score, the reply low-grade packet loss score, and/or a combined low-grade packet loss score.

[0105] In some implementations, rather than generating and sending back a UDP echo reply message with a corresponding number of UDP packets, the connectivity measurement system

206 sends back the same and/or a predetermined number as sent in the UDP echo request message. For example, both the UDP echo client 210 and the UDP echo server 220 send 45, 60, 100, or 150 packets regardless of how many packets the UDP echo server 220 receives.

[0106] In various implementations, the connectivity measurement system 206 sends a UDP echo reply message 122 that includes a single UDP packet (or more packets for redundancy depending on the low-grade packet loss score). For example, if the UDP echo server 220 determines a low-grade packet loss score of zero, the UDP echo server 220 sends back a UDP echo reply message of one or two packets indicating the low-grade packet loss score to the UDP echo client 210. As the low-grade packet loss score increases, the UDP echo server 220 can send additional UDP packets indicating the low-grade packet loss score to ensure packet loss does not affect the UDP echo reply message. In some implementations, the UDP echo server 220 sends back a UDP echo reply message of one or two packets indicating the number of packets received at the UDP echo server 220 for the UDP echo client 210 to use to determine a low-grade packet loss score.

[0107] While FIG. 4 illustrates a case where all of the UDP echo request message packets were successfully received by the UDP echo server 220, FIG. 5 discusses determining a low-grade packet loss score when one or more UDP packets are lost. In particular, FIG. 5 illustrates a sequence flow diagram example of determining low-grade packet loss between computing devices when one or more of the UDP echo request message packets are not successfully received according to some implementations.

[0108] FIG. 5 includes various acts described earlier in FIG. 4. For example, FIG. 5 includes the act 402 of the UDP echo client 210 determining to send an echo request message to the UDP echo server 220, the act 404 of the connectivity measurement system 206 generating a set of

separate UDP packets of a first message type, and the act 406 of the UDP echo client 210 sending the echo request message packets to the UDP echo server 220 as part of measuring network connectivity and determining low-grade packet loss.

[0109] As before, the UDP echo server 220 receives the UDP echo request message packets. As shown, the act 508 includes the UDP echo server 220 receiving most of the echo request message packets of the first message type within a message-receiving threshold. In some instances, the UDP echo server 220 does not receive all of the echo request message packets within the message-receiving threshold (e.g., receiving packets within 5 seconds of the UDP echo request message being sent or within 5 seconds of receiving the first of the echo request message packets). For example, one or more of the UDP packets are lost or dropped during transmission. As another example, the UDP echo server 220 receives one or more of the echo request message packets after the message-receiving threshold. Accordingly, in these instances, the UDP echo server 220 receives a number of echo request message packets that is less than the packet count total of the echo request message.

[0110] With existing systems, when a target server device does not receive all of the packets of an echo message, the echo message is incomplete and the target server will show a timeout as a result. Furthermore, if the echo message is a request message, the target server will not send a reply message to the source device. In contrast to existing systems, the connectivity measurement system 206 determines the number of UDP packets received for the UDP echo message, which will indicate when packets in the UDP echo reply message accounts were lost.

[0111] As mentioned earlier, in various implementations, the UDP packets of the UDP echo request message include an identifier, such as an index number. In these implementations, the UDP echo server 220 can determine which of the UDP packets were lost and/or if a group of

packets was lost (e.g., if sent together or sequentially). In some instances, the UDP echo server 220 can report which packets were lost in the UDP echo reply message.

[0112] After receiving the UDP echo reply message packets, the connectivity measurement system 206 associated with the UDP echo server 220 can determine a low-grade packet loss score. For example, the act 510 shows the UDP echo server 220 determining a low-grade packet loss score based on the number of received echo request packets. For example, the connectivity measurement system 206 compares the number of received echo request packets with the packet count total to determine that not all of the echo request packets were received and/or a low-grade packet loss score above zero for transmissions from the UDP echo client 210.

[0113] In various implementations, the connectivity measurement system 206 determines the low-grade packet loss score based on the number of lost packets out of the packet count total. For instance, the connectivity measurement system 206 first determines the number of lost packets by comparing the number of received packets to the packet count total to determine the number of lost packets (100 total packets – 97 received packets = 3 lost packets). In additional implementations, the connectivity measurement system 206 determines the low-grade packet loss score based on comparing and/or dividing the number of lost packets with the packet count total (e.g., 3 lost packets / 100 total packets = 3% packet loss).

[0114] As mentioned above, in various implementations, the connectivity measurement system 206 utilizes additional factors to determine the low-grade packet loss score. For example, the connectivity measurement system 206 utilizes an algorithm that includes factors such as the physical/geographical length between the UDP echo client 210 and the UDP echo server 220, the time of day the UDP echo request message is being sent, the level of network congestion at the time of transmission, the buffer size of the UDP echo message, and/or the message type.

Depending on the application and weighting, one or more of these factors could increase or decrease the low-grade packet loss score when one or more UDP packets are lost.

[0115] Upon generating a low-grade packet loss score or based on omitting this action until later (e.g., the act 418), the UDP echo server 220 generates an echo reply message with a corresponding number of separate UDP packets of a second message type, as shown in the act 512. In particular, the connectivity measurement system 206 associated with the UDP echo server 220 generates the echo reply message to include an additional set of separate UDP packets of the UDP that corresponds to received UDP packets from the set of separate UDP packets of the echo request message.

[0116] To further illustrate, in various implementations, based on the UDP echo server 220 receiving 42 of 45 UDP packets of a UDP echo request message, the UDP echo server 220 generates 42 corresponding packets for the UDP echo reply message. The UDP echo server 220 can generate the packets as described above (e.g., by duplicating the identifier and/or padding data in corresponding packets). In some instances, the UDP echo server 220 adjusts the packet count total to reflect that the additional set of separate UDP packets of the UDP echo reply message includes a lower number of packets than the UDP echo request message.

[0117] FIG. 5 also shows the act 414 of the UDP echo server 220 sending the echo reply message packets back to the UDP echo client 210. For example, the UDP echo server 220 sends the additional set of separate UDP packets of the UDP echo reply message to the UDP echo client 210 in response to the UDP echo request message. In some implementations, the UDP echo server 220 sends the UDP echo reply message as one, two, or a few UDP packets that indicate the low-grade packet loss score and/or the number of packets received at the UDP echo server 220, as explained above.

[0118] As also shown, FIG. 5 includes the act 416 of the UDP echo client 210 receiving the echo reply message of the second message type and the act 418 of determining a low-grade packet loss score based on the received packets. As described above, the connectivity measurement system 206 on the UDP echo client 210 may determine one or more network health metrics based on the UDP echo request message, the UDP echo reply message, and/or both UDP echo messages (e.g., an RTT).

[0119] Based on determining a low-grade packet loss score above zero, or a low-grade packet loss score indicating some amount of low-grade packet loss, the connectivity measurement system 206 can implement one or more further actions. In some implementations, the connectivity measurement system 206 initiates a further investigation of where the low-grade packet loss is occurring. In some implementations, the connectivity measurement system 206 instructs the service using the computing device associated with the UDP echo client 210 and/or the UDP echo server 220 to switch to a server on another device. In some implementations, the connectivity measurement system 206 initiates mitigating actions such as causing the real-world data to perform retransmissions, switch transmission protocols, add forward error correction, utilize a different path or packet size, and so on.

[0120] Turning now to FIG. 6, this figure illustrates an example flowchart that includes a series of acts for utilizing the connectivity measurement system 206 according to one or more implementations. In particular, FIG. 6 illustrates an example series of acts of computer-implemented methods for determining low-grade packet loss between computing systems and/or devices utilizing user datagram protocol (UDP) messages.

[0121] While FIG. 6 illustrates acts according to one or more implementations, alternative implementations may omit, add, reorder, and/or modify any of the acts shown. Further, the acts

of FIG. 6 can be performed as part of a method such as a computer-implemented method. Alternatively, a non-transitory computer-readable medium can include instructions that, when executed by a processor, cause a computing device to perform the acts of FIG. 6.

[0122] In further implementations, a system can perform (e.g., a processing system with a processor can cause instructions to be performed) the acts of FIG. 6. For example, in some instances, the system includes a first computing device such as a UDP echo client, and a second computing device such as a UDP server device (or vice versa). In addition, in various instances, the system includes a processing system having a processor and a computer memory including instructions that, when executed by the processing system, cause the system to perform the operations included in the acts of FIG. 6.

[0123] As shown, the series of acts 600 includes an act 610 of sending separate UDP echo request message packets from a first device to a second device. For example, the act 610 involves sending, by the first computing device, a UDP echo request message that includes a set of separate UDP packets to the second computing device to which the first computing device sends other communications. In some implementations, the act 610 includes generating the set of separate UDP packets by assigning a unique identifier to each UDP packet in the set of separate UDP packets. In various implementations, the set of separate UDP packets includes non-fragmented packets. In some implementations, the act 610 includes generating the set of separate UDP packets by padding each UDP packet in the set of separate UDP packets with padding data to reach a predetermined size. In additional implementations, the predetermined size is the maximum transmission unit (MTU) size of UDP packets receivable by that second computing device.

[0124] In various implementations, the act 610 includes generating the set of separate UDP packets by assigning a UDP echo request message type to each UDP packet in the set of separate UDP packets. In some implementations, the first computing device includes a UDP echo client that generates the UDP echo request message and the second computing device includes a UDP echo server that receives the number of received UDP packets of the UDP echo request message.

[0125] In some implementations, each UDP packet in the set of separate UDP packets is assigned a UDP echo request message type and a unique identifier. In one or more implementations, the first computing device is located in a first datacenter that is in a different geographic region from a second datacenter that includes the second computing device. In some implementations, the set of separate UDP packets includes 100 packets, and where the message-receiving threshold is five (5) seconds or less. In some implementations, each packet in the set of separate UDP packets includes header bytes, a UDP echo message type byte, an identifier byte, and padding data bytes.

[0126] As further shown, the series of acts 600 includes an act 620 of receiving a number of received UDP packets from the UDP echo request message packets at the second device. For instance, in example implementations, the act 620 involves receiving, by the second computing device, a number of received UDP packets from the set of separate UDP packets of the UDP echo request message from the first computing device within a message-receiving threshold. In some implementations, the act 620 includes receiving, from a first computing device, a UDP echo request message at a second computing device, where the UDP echo request message includes the number of received UDP packets from a set of separate UDP packets.

[0127] As further shown, the series of acts 600 includes an act 630 of determining a low-grade packet loss score based on the number of received UDP packets. For instance, in example

implementations, the act 630 involves determining a low-grade packet loss score between the first computing device and the second computing device based on the number of received UDP packets of the UDP echo request message received by the second computing device. In various implementations, the act 630 involves determining a low-grade packet loss score between the first computing device and the second computing device based on comparing the number of received UDP packets received from the UDP echo request message with a packet count total of the UDP echo request message.

[0128] In various implementations, the act 630 includes comparing the number of received UDP packets received of the UDP echo request message based on a packet count total of the UDP echo request message. In some implementations, determining the low-grade packet loss score as zero based on the number of received UDP packets received from the UDP echo request message matching the packet count total of the UDP echo request message or determining the low-grade packet loss score as greater than zero based on the number of received UDP packets received from the UDP echo request message being less than the packet count total of the UDP echo request message. In one or more implementations, the act 630 includes generating the set of separate UDP packets by including the packet count total in one or more UDP packets in the set of separate UDP packets.

[0129] In various implementations, the act 630 includes determining a low-grade packet loss score between the first computing device and the second computing device based on the number of received UDP packets of the UDP echo request message received by the second computing device. In some implementations, the act 630 includes sending, by the second computing device and to the first computing device, a UDP echo reply message that indicates the number of received UDP packets received from the UDP echo request message at the second computing

device or the low-grade packet loss score between the first computing device and the second computing device.

[0130] In some implementations, the series of acts 600 includes additional acts. For example, the series of acts 600 includes acts of sending, by the second computing device, a UDP echo reply message that includes an additional set of separate UDP packets to the first computing device. In additional implementations, the series of acts 600 includes receiving, by the first computing device, an additional number of received UDP packets from an additional set of separate UDP packets of a UDP echo reply message from the second computing device. In one or more implementations, the additional number of received UDP packets corresponds to a number of received UDP packets of the UDP echo request message that are received by the second computing device. In some instances, the additional number of received UDP packets is assigned a UDP echo reply message type and has unique identifiers corresponding to the set of separate UDP packets.

[0131] In various implementations, the series of acts 600 includes acts of generating the additional set of separate UDP packets by duplicating the set of separate UDP packets received at the second computing device and changing the UDP echo request message type to a UDP echo reply message type. In some implementations, the series of acts 600 includes sending, by the second computing device, a UDP echo reply message that includes an additional set of separate UDP packets to the first computing device, where the additional set of separate UDP packets includes an additional number of received UDP packets that corresponds to the number of received UDP packets received from the UDP echo request message.

[0132] In some implementations, the series of acts 600 includes receiving, by the first computing device, an additional number of received UDP packets from the additional set of

separate UDP packets of the UDP echo reply message from the second computing device. In various implementations, the series of acts 600 includes establishing a secure channel between the first computing device and the second computing device before sending the UDP echo request message.

[0133] FIG. 7 illustrates certain components that may be included within a computer system 700. The computer system 700 may be used to implement the various computing devices, components, and systems described herein (e.g., by performing computer-implemented instructions). As used herein, a "computing device" refers to electronic components that perform a set of operations based on a set of programmed instructions. Computing devices include groups of electronic components, client devices, server devices, etc.

[0134] In various implementations, the computer system 700 represents one or more of the client devices, server devices, or other computing devices described above. For example, the computer system 700 may refer to various types of network devices capable of accessing data on a network, a cloud computing system, or another system. For instance, a client device may refer to a mobile device such as a mobile telephone, a smartphone, a personal digital assistant (PDA), a tablet, a laptop, or a wearable computing device (e.g., a headset or smartwatch). A client device may also refer to a non-mobile device such as a desktop computer, a server node (e.g., from another cloud computing system), or another non-portable device.

[0135] The computer system 700 includes a processing system including a processor 701. The processor 701 may be a general-purpose single- or multi-chip microprocessor (e.g., an Advanced Reduced Instruction Set Computer (RISC) Machine (ARM)), a special-purpose microprocessor (e.g., a digital signal processor (DSP)), a microcontroller, a programmable gate array, etc. The processor 701 may be referred to as a central processing unit (CPU) and may

cause computer-implemented instructions to be performed. Although the processor 701 shown is just a single processor in the computer system 700 of FIG. 7, in an alternative configuration, a combination of processors (e.g., an ARM and DSP) could be used.

[0136] The computer system 700 also includes memory 703 in electronic communication with the processor 701. The memory 703 may be any electronic component capable of storing electronic information. For example, the memory 703 may be embodied as random-access memory (RAM), read-only memory (ROM), magnetic disk storage media, optical storage media, flash memory devices in RAM, on-board memory included with the processor, erasable programmable read-only memory (EPROM), electrically erasable programmable read-only memory (EEPROM), registers, and so forth, including combinations thereof.

[0137] The instructions 705 and the data 707 may be stored in the memory 703. The instructions 705 may be executable by the processor 701 to implement some or all of the functionality disclosed herein. Executing the instructions 705 may involve the use of the data 707 that is stored in the memory 703. Any of the various examples of modules and components described herein may be implemented, partially or wholly, as instructions 705 stored in memory 703 and executed by the processor 701. Any of the various examples of data described herein may be among the data 707 that is stored in memory 703 and used during the execution of the instructions 705 by the processor 701.

[0138] A computer system 700 may also include one or more communication interface(s) 709 for communicating with other electronic devices. The one or more communication interface(s) 709 may be based on wired communication technology, wireless communication technology, or both. Some examples of the one or more communication interface(s) 709 include a Universal Serial Bus (USB), an Ethernet adapter, a wireless adapter that operates according to

an Institute of Electrical and Electronics Engineers (IEEE) 702.11 wireless communication protocol, a Bluetooth® wireless communication adapter, and an infrared (IR) communication port.

[0139] A computer system 700 may also include one or more input device(s) 711 and one or more output device(s) 713. Some examples of the one or more input device(s) 711 include a keyboard, mouse, microphone, remote control device, button, joystick, trackball, touchpad, and light pen. Some examples of the one or more output device(s) 713 include a speaker and a printer. A specific type of output device that is typically included in a computer system 700 is a display device 715. The display device 715 used with implementations disclosed herein may utilize any suitable image projection technology, such as liquid crystal display (LCD), light-emitting diode (LED), gas plasma, electroluminescence, or the like. A display controller 717 may also be provided, for converting data 707 stored in the memory 703 into text, graphics, and/or moving images (as appropriate) shown on the display device 715.

[0140] The various components of the computer system 700 may be coupled together by one or more buses, which may include a power bus, a control signal bus, a status signal bus, a data bus, etc. For clarity, the various buses are illustrated in FIG. 7 as a bus system 719.

[0141] This disclosure describes a connectivity measurement system in the framework of a network. In this disclosure, a "network" refers to one or more data links that enable electronic data transport between computer systems, modules, and other electronic devices. A network may include public networks such as the Internet as well as private networks. When information is transferred or provided over a network or another communication connection (either hardwired, wireless, or both), the computer correctly views the connection as a transmission medium. Transmission media can include a network and/or data links that carry required program code in

the form of computer-executable instructions or data structures, which can be accessed by a general-purpose or special-purpose computer.

[0142] In addition, the network described herein may represent a network or a combination of networks (such as the Internet, a corporate intranet, a virtual private network (VPN), a local area network (LAN), a wireless local area network (WLAN), a cellular network, a wide area network (WAN), a metropolitan area network (MAN), or a combination of two or more such networks) over which one or more computing devices may access the various systems described in this disclosure. Indeed, the networks described herein may include one or multiple networks that use one or more communication platforms or technologies for transmitting data. For example, a network may include the Internet or other data link that enables transporting electronic data between respective client devices and components (e.g., server devices and/or virtual machines thereon) of the cloud computing system.

[0143] Further, upon reaching various computer system components, program code means in the form of computer-executable instructions or data structures can be transferred automatically from transmission media to non-transitory computer-readable storage media (devices), or vice versa. For example, computer-executable instructions or data structures received over a network or data link can be buffered in random-access memory (RAM) within a network interface module (NIC), and then it is eventually transferred to computer system RAM and/or to less volatile computer storage media (devices) at a computer system. Thus, it should be understood that non-transitory computer-readable storage media (devices) can be included in computer system components that also (or even primarily) utilize transmission media.

[0144] Computer-executable instructions include instructions and data that, when executed by a processor, cause a general-purpose computer, special-purpose computer, or special-purpose

processing device to perform a certain function or group of functions. In some implementations, computer-executable and/or computer-implemented instructions are executed by a general-purpose computer to turn the general-purpose computer into a special-purpose computer implementing elements of the disclosure. The computer-executable instructions may include, for example, binaries, intermediate format instructions such as assembly language, or even source code. Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the described features or acts described above. Rather, the described features and acts are disclosed as example forms of implementing the claims.

[0145] Those skilled in the art will appreciate that the disclosure may be practiced in network computing environments with many types of computer system configurations, including, personal computers, desktop computers, laptop computers, message processors, hand-held devices, multi-processor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, mobile telephones, PDAs, tablets, pagers, routers, switches, and the like. The disclosure may also be practiced in distributed system environments where local and remote computer systems, which are linked (either by hardwired data links, wireless data links, or a combination of hardwired and wireless data links) through a network, both perform tasks. In a distributed system environment, program modules may be located in both local and remote memory storage devices.

[0146] The techniques described herein may be implemented in hardware, software, firmware, or any combination thereof unless specifically described as being implemented in a specific manner. Any features described as modules, components, or the like may also be implemented together in an integrated logic device or separately as discrete but interoperable

logic devices. If implemented in software, the techniques may be realized at least in part by a non-transitory processor-readable storage medium, including instructions that, when executed by at least one processor, perform one or more of the methods described herein (including computer-implemented methods). The instructions may be organized into routines, programs, objects, components, data structures, etc., which may perform particular tasks and/or implement particular data types, and which may be combined or distributed as desired in various implementations.

[0147] Computer-readable media can be any available media that can be accessed by a general-purpose or special-purpose computer system. Computer-readable media that store computer-executable instructions are non-transitory computer-readable storage media (devices). Computer-readable media that carry computer-executable instructions are transmission media. Thus, by way of example, implementations of the disclosure can include at least two distinctly different kinds of computer-readable media: non-transitory computer-readable storage media (devices) and transmission media.

[0148] As used herein, non-transitory computer-readable storage media (devices) may include RAM, ROM, EEPROM, CD-ROM, solid-state drives (SSDs) (e.g., based on RAM), Flash memory, phase-change memory (PCM), other types of memory, other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other medium which can be used to store desired program code means in the form of computer-executable instructions or data structures and which can be accessed by a general-purpose or special-purpose computer.

[0149] The steps and/or actions of the methods described herein may be interchanged with one another without departing from the scope of the claims. In other words, unless a specific order of steps or actions is required for the proper operation of the method that is being described,

the order and/or use of specific steps and/or actions may be modified without departing from the scope of the claims.

[0150] The term "determining" encompasses a wide variety of actions and, therefore, "determining" can include calculating, computing, processing, deriving, investigating, looking up (e.g., looking up in a table, a data repository, or another data structure), ascertaining, and the like. Also, "determining" can include receiving (e.g., receiving information), accessing (e.g., accessing data in a memory), and the like. Also, "determining" can include resolving, selecting, choosing, establishing, and the like.

[0151] The terms "comprising," "including," and "having" are intended to be inclusive and mean that there may be additional elements other than the listed elements. Additionally, it should be understood that references to "one implementation" or "implementations" of the present disclosure are not intended to be interpreted as excluding the existence of additional implementations that also incorporate the recited features. For example, any element or feature described concerning an implementation herein may be combinable with any element or feature of any other implementation described herein, where compatible.

[0152] The present disclosure may be embodied in other specific forms without departing from its spirit or characteristics. The described implementations are to be considered illustrative and not restrictive. The scope of the disclosure is indicated by the appended claims rather than by the foregoing description. Changes that come within the meaning and range of equivalency of the claims are to be embraced within their scope.

What is claimed is:

1. A system for determining low-grade packet loss between computing devices utilizing user datagram protocol (UDP) messages, comprising:

a first computing device;

a second computing device;

a processing system comprising a processor; and

a computer memory comprising instructions that, when executed by the processing system, cause the system to perform operations comprising:

sending, by the first computing device, a UDP echo request message that includes a set of separate UDP packets to the second computing device to which the first computing device sends other communications;

receiving, by the second computing device, a number of received UDP packets from the set of separate UDP packets of the UDP echo request message from the first computing device within a message-receiving threshold; and

determining a low-grade packet loss score between the first computing device and the second computing device based on comparing the number of received UDP packets received from the UDP echo request message with a packet count total of the UDP echo request message.

2. The system of claim 1, wherein the set of separate UDP packets includes non-fragmented packets.

3. The system of claim 1, wherein the operations further comprise generating the set of separate UDP packets by assigning a unique identifier to each UDP packet in the set of separate UDP packets.

4. The system of claim 1, wherein the operations further comprise generating the set of separate UDP packets by padding each UDP packet in the set of separate UDP packets with padding data to reach a predetermined size.

5. The system of claim 4, wherein the predetermined size is a maximum transmission unit size of UDP packets receivable by that second computing device.

6. The system of claim 1, wherein the packet count total of the UDP echo request message is included in one or more of UDP packets in the set of separate UDP packets.

7. The system of claim 1, wherein the operations further comprise:

determining the low-grade packet loss score as zero based on the number of received UDP packets received from the UDP echo request message matching the packet count total of the UDP echo request message; or

determining the low-grade packet loss score as greater than zero based on the number of received UDP packets received from the UDP echo request message being less than the packet count total of the UDP echo request message.

8. The system of claim 7, wherein the operations further comprise generating the set of separate UDP packets by including the packet count total in one or more UDP packets in the set of separate UDP packets.

9. The system of claim 1, wherein the operations further comprise generating the set of separate UDP packets by assigning a UDP echo request message type to each UDP packet in the set of separate UDP packets.

10. The system of claim 9, wherein the operations further comprise:

sending, by the second computing device, a UDP echo reply message that includes an additional set of separate UDP packets to the first computing device; and

receiving, by the first computing device, an additional number of received UDP packets from the additional set of separate UDP packets of the UDP echo reply message from the second computing device.

11. The system of claim 10, wherein the operations further comprise generating the additional set of separate UDP packets by duplicating the set of separate UDP packets received at the second computing device and changing the UDP echo request message type to a UDP echo reply message type.

12. The system of claim 1, wherein:

the first computing device includes a UDP echo client that generates the UDP echo request message; and

the second computing device includes a UDP echo server that receives the number of received UDP packets of the UDP echo request message.

13. The system of claim 1, wherein the operations further comprise establishing a secure channel between the first computing device and the second computing device before sending the UDP echo request message.

14. The system of claim 1, wherein the set of separate UDP packets includes at least 100 packets, and wherein the message-receiving threshold is 5 seconds or less.

15. A computer-implemented method for determining low-grade packet loss between computing devices utilizing user datagram protocol (UDP) messages, comprising:

sending, by a first computing device, a UDP echo request message that includes a set of separate UDP packets to a second computing device, wherein each UDP packet in the set of separate UDP packets is assigned a UDP echo request message type and a unique identifier;

receiving, by the first computing device, an additional number of received UDP packets from an additional set of separate UDP packets of a UDP echo reply message from the second computing device, wherein the additional number of received UDP packets corresponds to a number of received UDP packets of the UDP echo request message that are received by the second computing device, and wherein the additional number of received UDP packets is assigned a UDP echo reply message type and has unique identifiers corresponding to the set of separate UDP packets; and

determining a low-grade packet loss score between the first computing device and the second computing device based on comparing the number of received UDP packets or the additional number of received UDP packets with a corresponding packet count total.

16. The computer-implemented method of claim 15, wherein each packet in the set of separate UDP packets includes header bytes, a UDP echo message type byte, an identifier byte, and padding data bytes.

17. The computer-implemented method of claim 15, wherein the first computing device is located in a first datacenter that is in a different geographic region from a second datacenter that includes the second computing device.

18. A computer-implemented method for determining low-grade packet loss between computing devices utilizing user datagram protocol (UDP) messages, comprising:

receiving, from a first computing device, a UDP echo request message at a second computing device, the UDP echo request message including a number count of received UDP packets from a set of separate UDP packets; and

determining a low-grade packet loss score between the first computing device and the second computing device based on comparing the number count of the received UDP packets received from the UDP echo request message by the second computing device with a packet count total of the UDP echo request message.

19. The computer-implemented method of claim 18, further comprising sending, by the second computing device, a UDP echo reply message that includes an additional set of separate UDP packets to the first computing device, wherein the additional set of separate UDP packets includes an additional number count of the received UDP packets that corresponds to the number count of the received UDP packets received from the UDP echo request message.

20. The computer-implemented method of claim 18, further comprising sending, by the second computing device and to the first computing device, a UDP echo reply message that indicates:

the number count of the received UDP packets received from the UDP echo request message at the second computing device; or

the low-grade packet loss score between the first computing device and the second computing device.

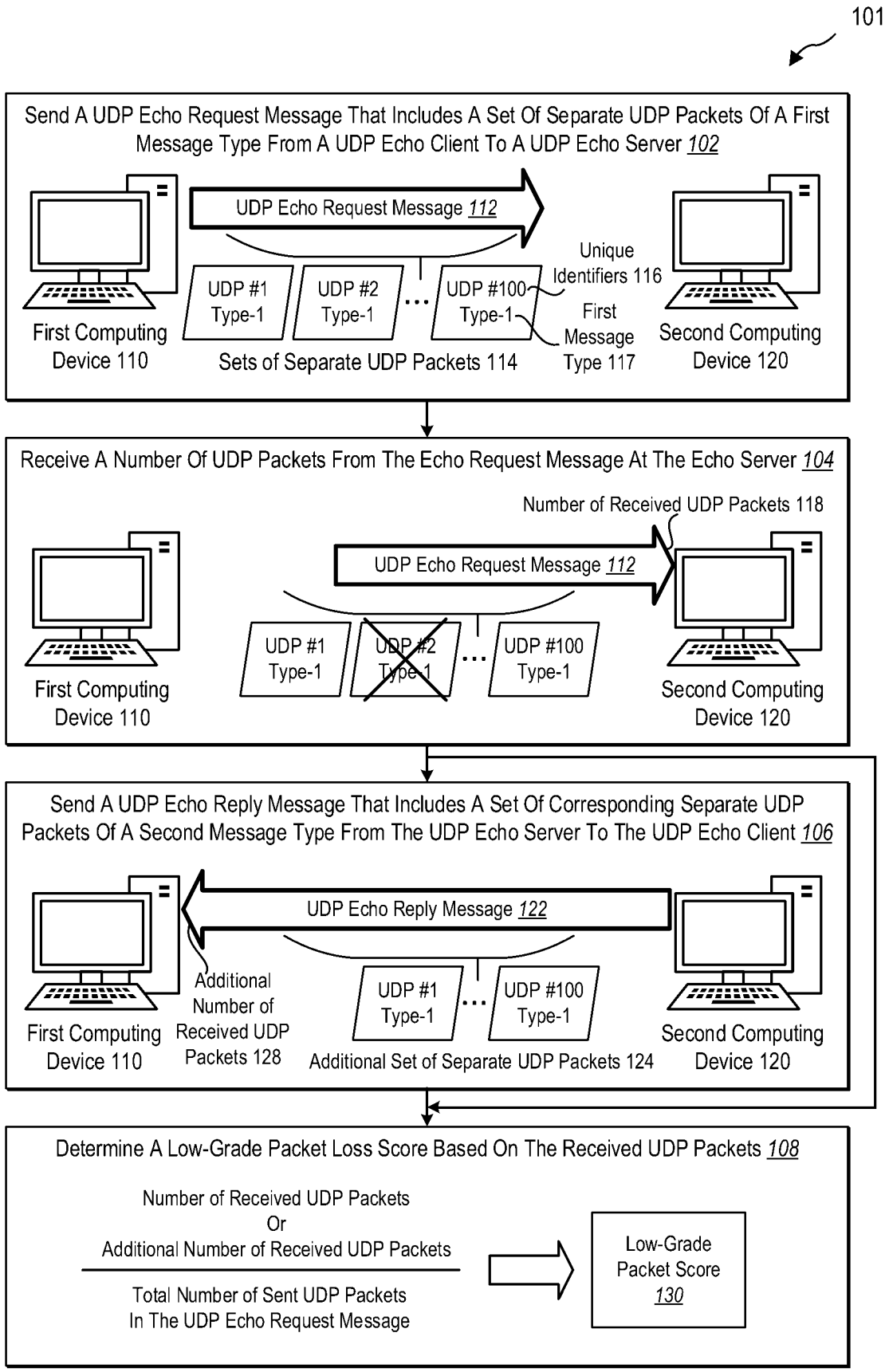
101
Send A UDP Echo Request Message That Includes A Set Of Separate UDP Packets Of A First Message Type From A UDP Echo Client To A UDP Echo Server 102
UDP Echo Request Message 112
Unique Identifiers 116
UDP #1 Type-1
UDP #2 Type-1
UDP #100 Type-1
First Message Type 117
First Computing Device 110
Sets of Separate UDP Packets 114
Second Computing Device 120
Receive A Number Of UDP Packets From The Echo Request Message At The Echo Server 104
Number of Received UDP Packets 118
UDP Echo Request Message 112
UDP #1 Type-1
UDP #2 Type-1
UDP #100 Type-1
First Computing Device 110
Second Computing Device 120
Send A UDP Echo Reply Message That Includes A Set Of Corresponding Separate UDP Packets Of A Second Message Type From The UDP Echo Server To The UDP Echo Client 106
UDP Echo Reply Message 122
Additional Number of Received UDP Packets 128
UDP #1 Type-1
UDP #100 Type-1
First Computing Device 110
Additional Set of Separate UDP Packets 124
Second Computing Device 120
Determine A Low-Grade Packet Loss Score Based On The Received UDP Packets 108
Number of Received UDP Packets
Or
Additional Number of Received UDP Packets
Total Number of Sent UDP Packets In The UDP Echo Request Message
Low-Grade Packet Score 130

FIG. 1

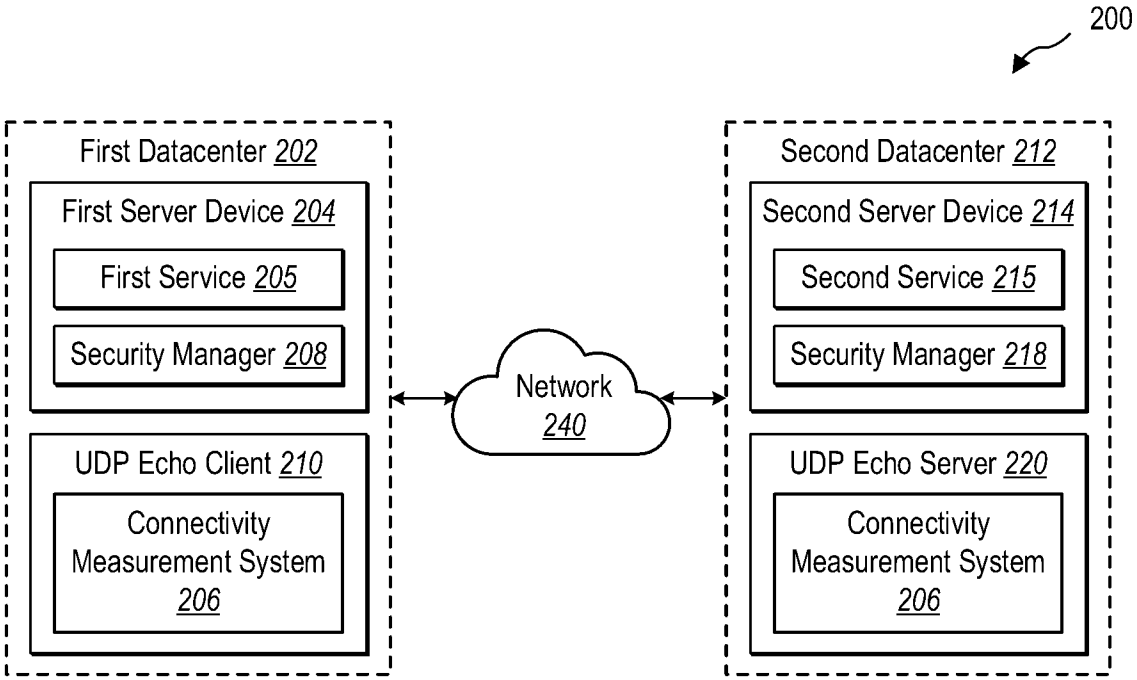
200
First Datacenter 202
First Server Device 204
First Service 205
Security Manager 208
UDP Echo Client 210
Connectivity Measurement System 206
Network 240
Second Datacenter 212
Second Server Device 214
Second Service 215
Security Manager 218
UDP Echo Server 220
Connectivity Measurement System 206

FIG. 2A

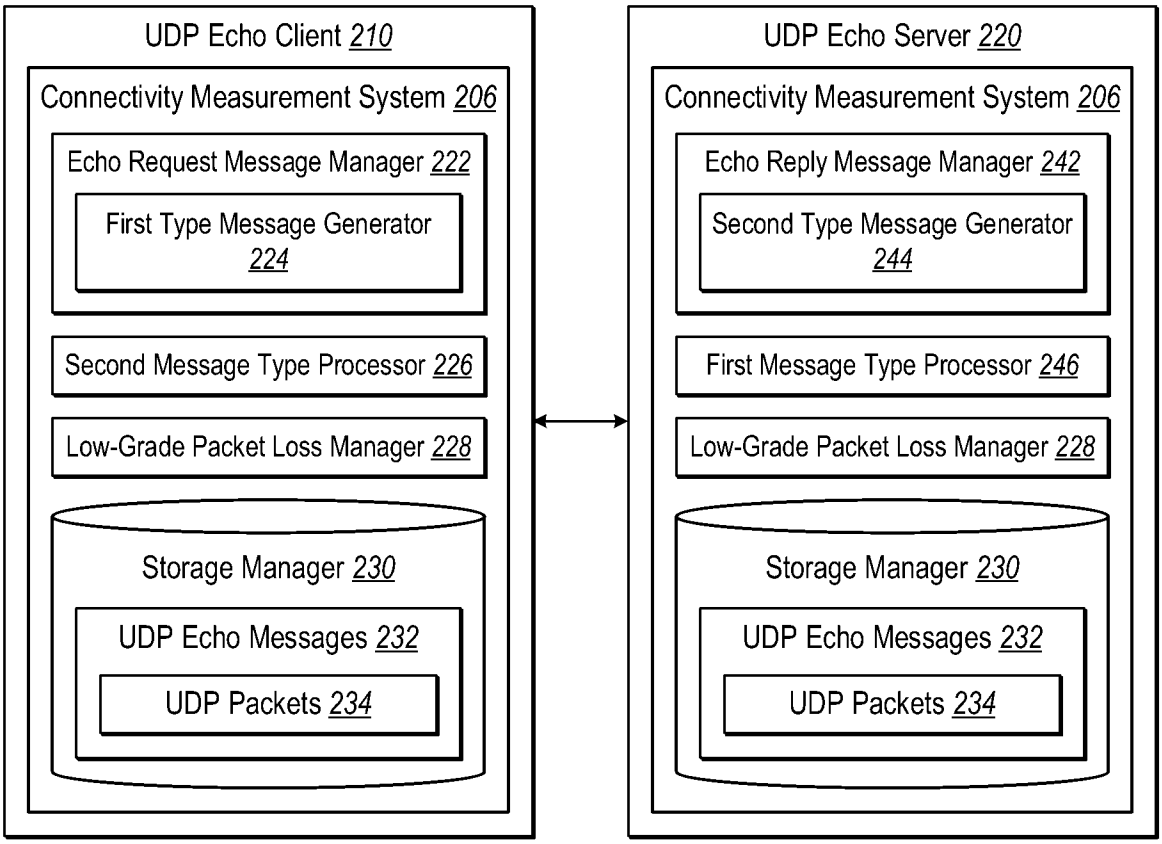
UDP Echo Client 210
Connectivity Measurement System 206
Echo Request Message Manager 222
First Type Message Generator 224
Second Message Type Processor 226
Low-Grade Packet Loss Manager 228
Storage Manager 230
UDP Echo Messages 232
UDP Packets 234
UDP Echo Server 220
Connectivity Measurement System 206
Echo Reply Message Manager 242
Second Type Message Generator 244
First Message Type Processor 246
Low-Grade Packet Loss Manager 228
Storage Manager 230
UDP Echo Messages 232
UDP Packets 234

FIG. 2B

UDP Packet 300

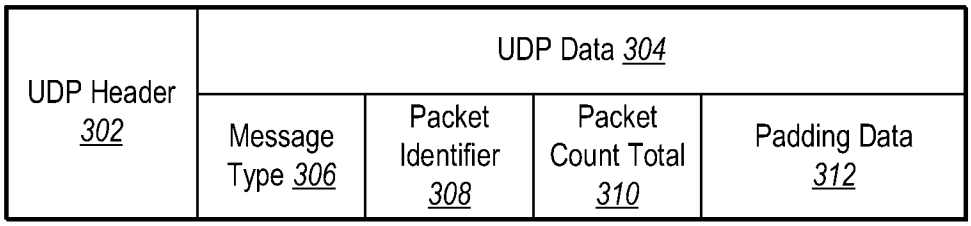

FIG. 3

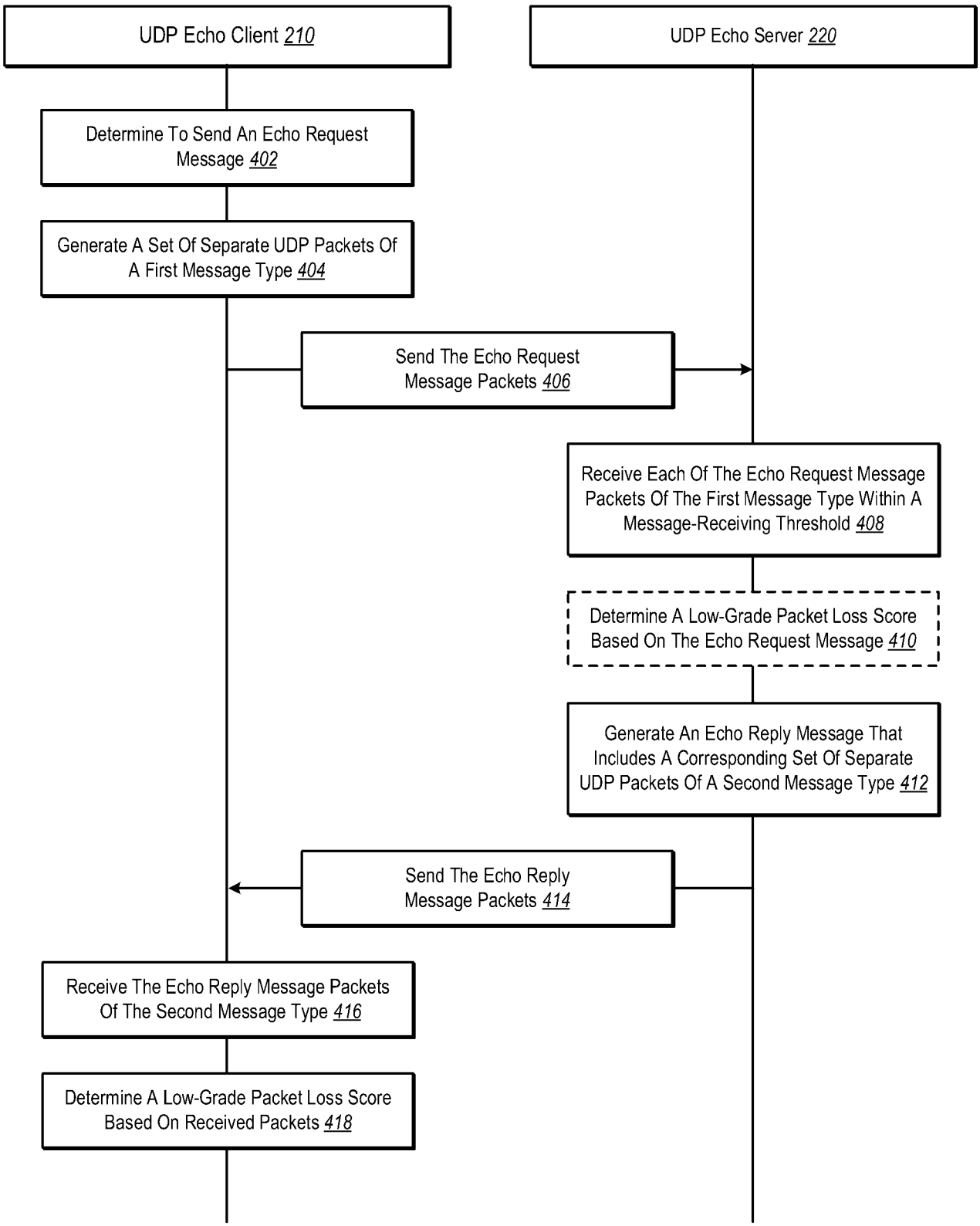
UDP Echo Client 210
UDP Echo Server 220
Determine To Send An Echo Request Message 402
Generate A Set Of Separate UDP Packets Of A First Message Type 404
Send The Echo Request Message Packets 406
Receive Each Of The Echo Request Message Packets Of The First Message Type Within A Message-Receiving Threshold 408
Determine A Low-Grade Packet Loss Score Based On The Echo Request Message 410
Generate An Echo Reply Message That Includes A Corresponding Set Of Separate UDP Packets Of A Second Message Type 412
Send The Echo Reply Message Packets 414
Receive The Echo Reply Message Packets Of The Second Message Type 416
Determine A Low-Grade Packet Loss Score Based On Received Packets 418

FIG. 4

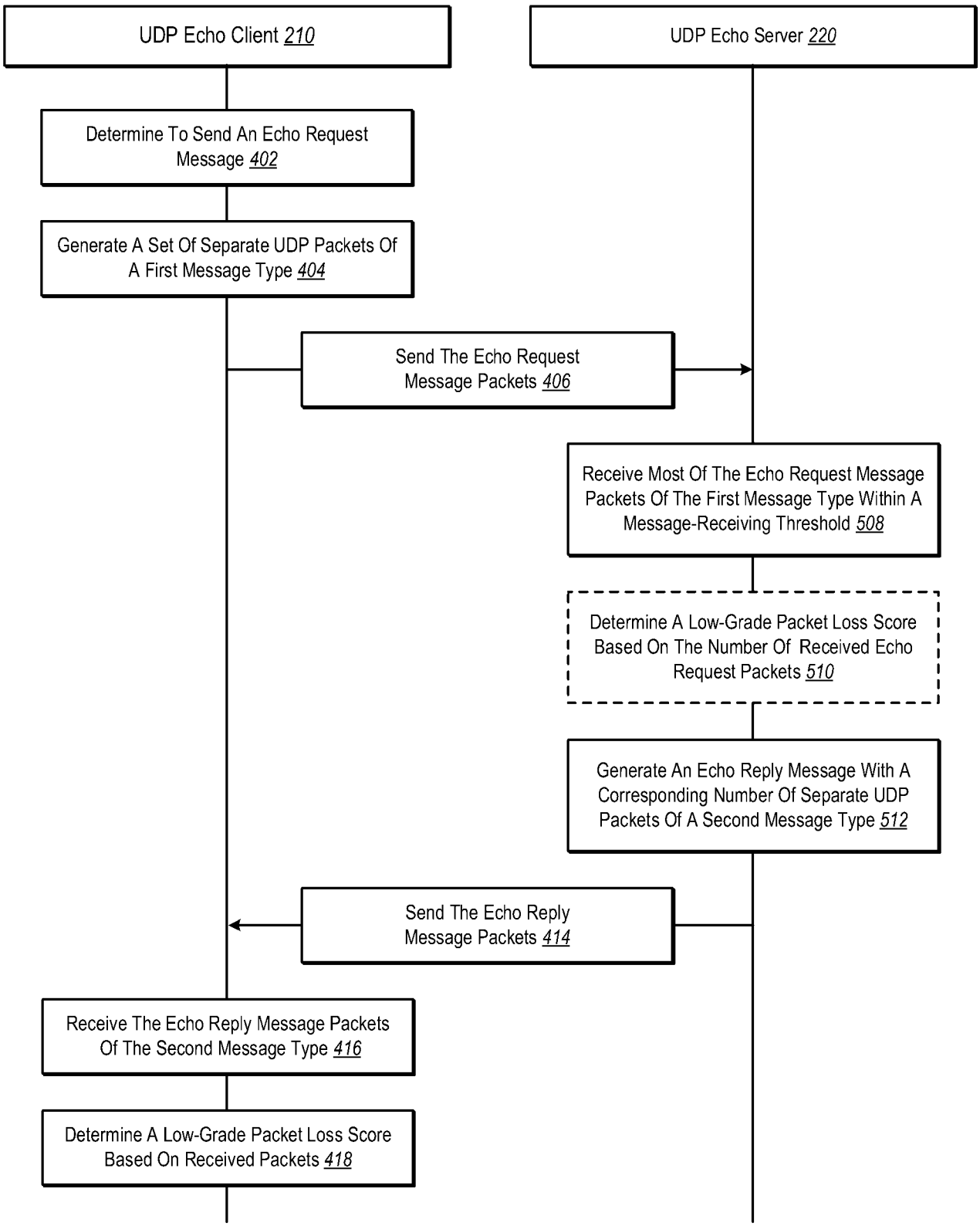
UDP Echo Client 210
UDP Echo Server 220
Determine To Send An Echo Request Message 402
Generate A Set Of Separate UDP Packets Of A First Message Type 404
Send The Echo Request Message Packets 406
Receive Most Of The Echo Request Message Packets Of The First Message Type Within A Message-Receiving Threshold 508
Determine A Low-Grade Packet Loss Score Based On The Number Of Received Echo Request Packets 510
Generate An Echo Reply Message With A Corresponding Number Of Separate UDP Packets Of A Second Message Type 512
Send The Echo Reply Message Packets 414
Receive The Echo Reply Message Packets Of The Second Message Type 416
Determine A Low-Grade Packet Loss Score Based On Received Packets 418

FIG. 5

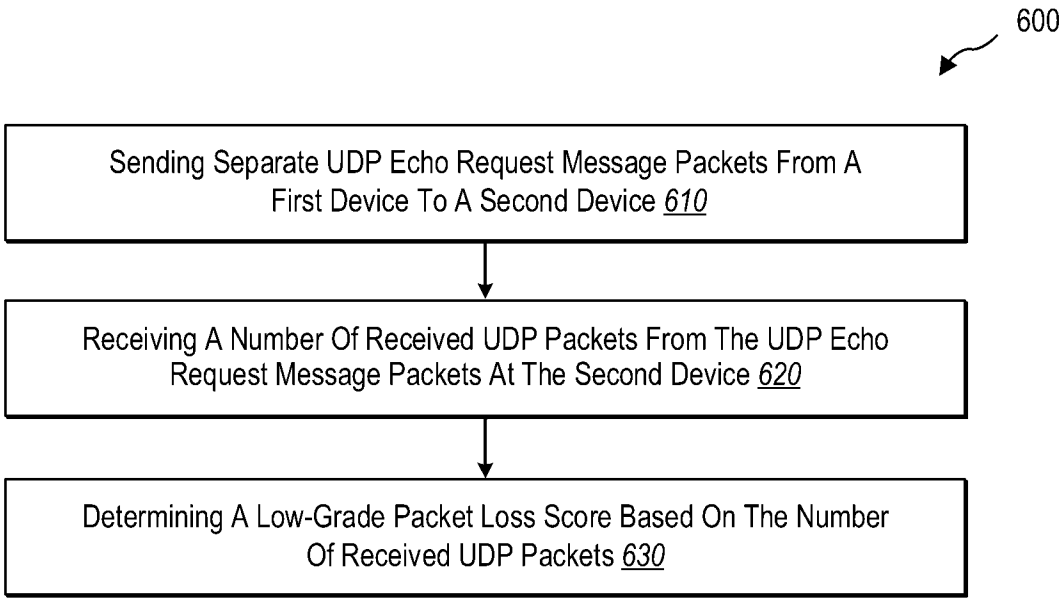
600
Sending Separate UDP Echo Request Message Packets From A First Device To A Second Device 610
Receiving A Number Of Received UDP Packets From The UDP Echo Request Message Packets At The Second Device 620
Determining A Low-Grade Packet Loss Score Based On The Number Of Received UDP Packets 630

FIG. 6

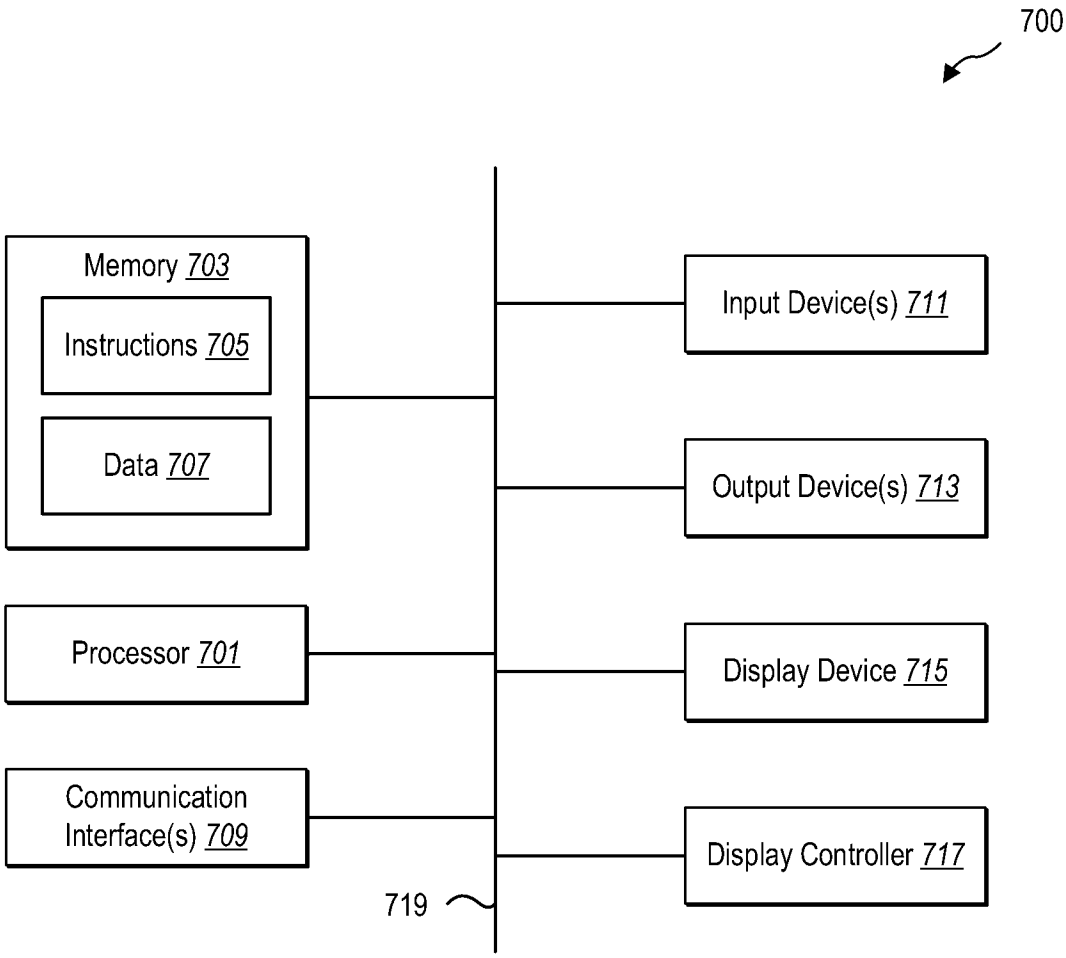
700
Memory 703
Instructions 705
Data 707
Processor 701
Communication Interface(s) 709
719
Input Device(s) 711
Output Device(s) 713
Display Device 715
Display Controller 717

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/CN2023/099605

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L43/0829 H04L43/10
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 H04W

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

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

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	JONES ET AL: "TR-143: Enabling network Throughput Performance Tests and Statistical Monitoring", INTERNET CITATION, 1 May 2008 (2008-05-01), pages 1-48, XP002602044, Retrieved from the Internet: URL:http://www.broadband-forum.org/technical/download/TR-143.pdf [retrieved on 2010-09-22] page 14 page 21 - page 22 page 26 - page 30 page 32; figure 3 ----- -/--	1-20

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

[x] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
16 November 2023	28/11/2023

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

1

Form PCT/ISA/210 (second sheet) (April 2005)

INTERNATIONAL SEARCH REPORT

International application No
PCT/CN2023/099605

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	WO 2011/124182 A2 (HUAWEI TECH CO LTD [CN]; XIONG SHAOCHENG [CN] ET AL.) 13 October 2011 (2011-10-13) page 5 - page 7	1-5,7, 9-12, 14-18
X	US 2014/086066 A1 (FILIPESCU EMILIAN-MIHAI [RO] ET AL) 27 March 2014 (2014-03-27) paragraph [0016] - paragraph [0022] paragraph [0031] - paragraph [0038] paragraph [0043]	1-12,14, 15,18

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/CN2023/099605

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2011124182 A2	13-10-2011	CN 102301765 A	28-12-2011
		WO 2011124182 A2	13-10-2011
US 2014086066 A1	27-03-2014	NONE	