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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
16 July 2009 (16.07.2009)

PCT

(10) International Publication Number
WO 2009/088333 A1

(51) International Patent Classification:
H04L 29/06 (2006.01) *H04L 29/08* (2006.01)

(21) International Application Number:
PCT/SE2008/050556

(22) International Filing Date: 13 May 2008 (13.05.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/020,489 11 January 2008 (11.01.2008) US

(71) Applicant *(for all designated States except US)*: **TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)** [SE/SE]; S-164 83 Stockholm (SE).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **BRODIN, Per-Erik** [SE/SE]; Skjutbanevägen 47, S-191 44 Sollentuna (SE). **SUN, Jiong** [CN/SE]; Synålsvägen 23 rum 132, S-168 73 Bromma (SE).

(74) Agent: **BERGENSTRÅHLE & LINDVALL AB**; Box 17704, S-118 93 Stockholm (SE).

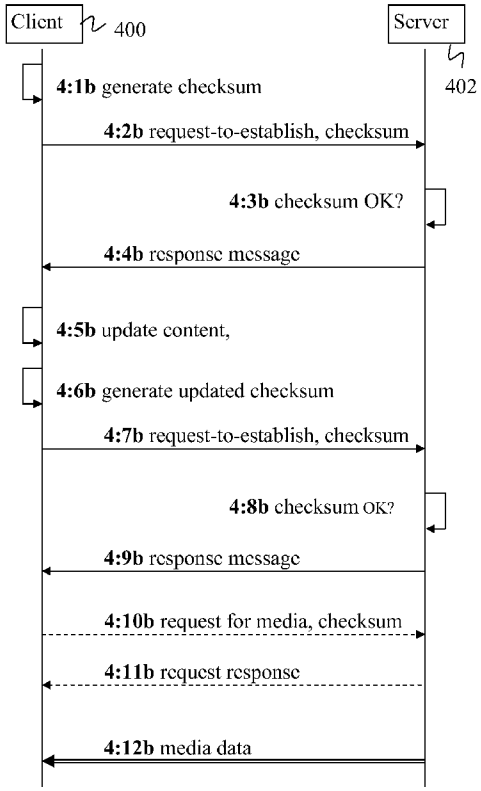
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM,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

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

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR ESTABLISHING A STREAMED MEDIA SESSION

Figure 4b



(57) Abstract: A method and an arrangement for enabling a communication session for streamed media between a client terminal and a server. A request-to-establish message is sent (202) from the client terminal, supplemented with a checksum generated (200) for available SDP information. The checksum is compared to a checksum generated for correct SDP information available at the server. A response message indicating the result of the comparison is received from the server (204). If the comparison indicates that the SDP information available at theserver is incorrect, the response message is supplemented with the correct SDP information, to update (210) the available SDP information. The client terminal is then sending a new request-to-message supplemented with the updated SDP information. On the other hand, if the available SDP information is correct, streamed media is started to be sent (208) from the server to the client terminal. Generating a checksum of SDP information available at the client terminal and supplement to a request-to-establishment message, enables for the server to check that the SDP information available at the client terminal is up to date, thereby achieving a fast and reliable establishment of a communication session for streamed media between a client terminal and a server.

WO 2009/088333 A1

Declarations under Rule 4.17:

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

— *of inventorship (Rule 4.17(iv))*

Published:

— *with international search report*

— *with information concerning request for restoration of the right of priority in respect of one or more priority claims*

METHOD AND APPARATUS FOR ESTABLISHING A STREAMED MEDIA SESSION

TECHNICAL FIELD

The present invention relates generally to establishment of Packet-switched Streaming Service (PSS) sessions, where streamed media is sent from a server to a client terminal. In particular, the present invention can be used where a mobile phone receives streamed media data from a media data server.

BACKGROUND

In packet-switched communication service systems, digital information is divided into a plurality of information packets which are distributed through a communication network from a sending party to a receiving party. Typically, the information packets are distributed using a plurality of communication paths between the sending party and the receiving party, and the original information is reconstructed at the receiving party from the received information packets. Although the information packets may be distributed using different communication paths, the sending party and the receiving party can be said to be connected by a virtual communication link. By setting up such a virtual communication link, a communication session for sending streamed media data will be established.

In this description, a client terminal is receiving streamed media over a virtual communication link. However, the client terminal and the server, respectively, may be any type of communication equipment suitable for transmitting or receiving streamed media on any type of communication link. For instance, the client terminal can be a mobile terminal in a UMTS (Universal Mobile Telecommunications System) or a computer connected to a computer communication network, etc.

With reference to FIGURE 1, a procedure where a client terminal 100 is setting up a communication session for streamed

media data with a server 102, according to prior art will now be schematically described. In a first **step 1:1**, the client terminal 100 initiates a communication session for streamed media data by sending a request to the server 102. Typically, the request is a <SETUP> message, according to known standards, and is therefore not necessary to describe in more detail here. In a following **step 1:2**, the server 102 confirms and accepts the request by sending an <OK> message to the client terminal 100. Then in a further **step 1:3**, the client terminal 100 sends another request to the server 102, to initiate the sending of media data from the server 102 to the client terminal. Typically, the start command is a <PLAY> message. In a subsequent **step 1:4**, the server 102 confirms and accepts the request to send media data, by sending another <OK> message to the client terminal 100. In a final **step 1:5**, the server starts sending the streamed media data from the server 102 to the communication terminal 100. The command messages are conventional messages according to standards for streamed media data services, which are not described in detail here. Typically, the command messages are supplemented with various information parameters. However, to simplify the understanding of the process, these information parameters are not shown in the schematic figure 1.

In general, it is necessary to provide session description information before the session can be executed. The term "session description information" refers to information regarding e.g. available communication paths between a sending party and a receiving party, information capacity for the communication paths, etc. Session description information is typically stored both at the server and at the client terminal.

For distribution of session description information to client terminals in packet-switched communication service systems two methods are generally applied today: Either the session

description information will be distributed from the server during the establishment of the communication session, or prior to the establishment of the communication session.

Distribution of session description information prior to establishment of a communication session can be performed when a client terminal wants to access a communication server at a later occasion, such as when a person subscribes to a service which allows downloading of streamed music from a server at a later occasion. The session description information can then be obtained via a communication path or via an alternative communication channel, such as an HTTP connection. An advantage of distributing the session description information prior to establishment of a communication session is that the time for establishing the communication session will be shortened.

Distribution of session description information during establishment of a communication session can be performed by beginning the establishment of the communication session by sending a request for session description information to the server, and then in response receiving the session description information. An advantage of distributing the session description information during the establishment of a communication session is that the session description information is up-to-date.

However, there are certain problems associated with the existing solutions outlined above. The user of a client terminal who wants to establish a communication session for streamed media data has to decide between spending a relatively long time on establishing the communication session, or running the risk of using session description information that is out of date. Using out of date session description information can give rise to unexpected errors if a selected communication path does not exist. On the other hand, the client terminal might not have knowledge of all available communication paths.

The session description information might be corrupted or modified during the transmission from the server to the client terminal, or during the storage at the client terminal.

SUMMARY

It is an object of the present invention to address at least some of the problems outlined above. In particular, it is an object to provide a solution which allows a relatively fast and reliable establishment of a communication session for streamed media between a client terminal and a server. These objects and others may be achieved primarily by a solution according to the attached independent claims.

According to different aspects, a client terminal and a method performed in the client terminal are provided for enabling a communication session for streamed media between the client terminal and a server.

In a method, a request-to-establish message is sent to the server, supplemented with session description information available at the client terminal. A response message is then received from the server, indicating whether the available session description information is correct or incorrect, the response message being supplemented with the correct session description information if the available session description information is incorrect. If the available session description information is correct, the session is executed by receiving media data from the server using the available session description information.

On the other hand, if the response message indicates that the available session description information is incorrect, the session description information at the client terminal may be updated with the received correct session description

information, and a new request-to-establish message may be sent from the client terminal to the server, supplemented with the correct updated session description information. Moreover, the session description information supplemented with the request-to-establish message may be realised by a checksum generated based on the entire available session description information or some specific part(s) of it, e.g. media information and/or attribute information. Furthermore, the streamed media may be received according to the RTSP and the session description information may be formatted in accordance with the SDP.

A client terminal comprises a session description information unit and a communication unit. The session description information unit is adapted to store session description information available at the client terminal. The communication unit is adapted to send a request-to-establish message to the server via a communication link, supplemented with the available session description information, and to receive a response message indicating if the sent session description information is correct or incorrect. The session description unit is further adapted to update its stored session description information with the received correct information when received from the server.

Furthermore, the communication unit may also be adapted to receive streamed media from a server. Moreover, the client terminal may comprise a checksum generator adapted to generate a checksum based on the entire available session description information or some specific part(s) of it. The communication unit may then be adapted to send the generated checksum instead of the available session description information. The session description information unit may also be adapted to update the entire stored session description information, or just the incorrect part(s).

According to further aspects, a server and a method performed in the server are provided for enabling a communication session for streamed media between a client terminal and the server.

In a method, a request-to-establish message is received from the client terminal, supplemented with a checksum based on session description information available at the client terminal. It is then determined whether the received checksum is correct or incorrect, by comparing the checksum to a correct checksum generated based on correct session description information at the server. A response message is then sent to the client terminal, indicating whether the checksum and thereby the session description information available at the client terminal is correct or incorrect, the response message being supplemented with the correct session description information if the session description information available at the client terminal is incorrect. If the session description information available at the client terminal is correct, the session is executed by sending media data to the client terminal using the correct session description information.

On the other hand, if the result of the comparison indicates that the available session description information is incorrect, the server waits for a new request-to-establish message from the client terminal, supplemented with a new checksum generated based on the correct updated session description information.

A server comprises a session description information unit, a communication unit, and a determining unit. The session description information unit is adapted to store correct session description information available at the server. The communication unit is adapted to receive a request-to-establish message from the client terminal via a communication link,

supplemented with session description information available at the client terminal, and to send a response message indicating if the received session description information is correct or incorrect. The determining unit is adapted to determine whether the received session description information is correct or incorrect, by comparing the received session description information to the correct one, and to output the result to the session description information unit. Moreover, the session description information unit is adapted to generate the response message, and supplement the response message with the correct session description information if the determining unit indicates that the received session description information is incorrect.

Furthermore, the server may comprise a checksum generator adapted to generate a checksum based on correct session description information available at the server, the communication unit may be adapted to receive a checksum based on the session description information available at the client terminal, and the determining unit may be adapted to compare the received checksum to the generated correct checksum.

According to another aspect, a method performed in a client terminal and a server is provided for enabling a communication session for streamed media between the client terminal and the server. A checksum based on at least a subset of session description information available at the client terminal is generated in the client terminal. Then a request-to-establish message is sent from the client terminal to the server, supplemented with the checksum generated in the client terminal. It is then determined in the server whether the received checksum is correct or incorrect, by comparing the checksum to a correct checksum generated based on correct session description information available at the server. A response message is then sent to the client terminal, indicating whether the checksum and

thereby the session description information available at the client terminal is correct or incorrect, the response message being supplemented with the correct session description information if the session description information available at the client terminal is incorrect. If the session description information available at the client terminal is correct, the session is executed by sending media data from the server to the client terminal using the correct session description information.

On the other hand, if the result of the comparison indicates that the session description information available at the server is incorrect, the client terminal updates its available session description information with the correct received session description information. Finally, the process is repeated, by generating a new checksum in the client terminal based on the updated session description information, and sending a new request-to-establish message to the server, supplemented with the new checksum.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will now be described in more detail by means of exemplary embodiments and with reference to the accompanying drawings, in which:

Figure 1 is a signalling diagram illustrating a scenario where a client terminal is setting up a communication session for streamed media with a server, according to prior art.

Figure 2 is a flow chart illustrating a procedure for enabling a communication session for streamed media data performed in a client terminal, in accordance with one embodiment.

Figure 3 is a flow chart illustrating a procedure for enabling a communication session for streamed media data performed in a server, in accordance with another embodiment.

Figure 4a is a signalling diagram illustrating a procedure for enabling a communication session for streamed media, in accordance with yet another embodiment.

Figure 4b is a signalling diagram illustrating a procedure for enabling a communication session for streamed media, in accordance with yet another embodiment.

Figure 5 is a block diagram illustrating a client terminal in more detail, in accordance with a further embodiment.

Figure 6 is a block diagram illustrating a server in more detail, in accordance with a further embodiment.

DETAILED DESCRIPTION

Briefly described, the present invention can be used such that a client terminal and a server can achieve a relatively faster and more reliable establishment of a communication session for streamed media. A checksum based on available session description information is generated in a client terminal before being sent to a server. The server then generates a corresponding checksum based on correct session description information available at the server. The correct checksum is compared in the server to the received checksum. Finally, the server sends a response message to the client terminal, the response message indicating if the received checksum is correct or incorrect. If the comparison of the checksums indicates that the received checksum is incorrect, the server further supplements the response message with the correct session description information.

Throughout this description, the term "checksum" refers to any result of a logical or arithmetic operation performed on a

bit stream, for enabling a check of the content of the bit stream. For instance, a hash-function, as MD5 (Message-Digest algorithm 5), SHA1, or a Cyclic Redundancy Check algorithm, or a check digit may be applied for generating the checksum. The functions and algorithms above are generally known in the art and are not necessary to describe in detail. Real Time Streaming Protocol is a commonly used protocol for transmission of streamed media data, and is hereinafter referred to as "RTSP". Session Description Protocol is a commonly used protocol for formatting session description information, regarding streamed media, and is hereinafter referred to as "SDP". The session description information comprises information regarding e.g. media and attributes. The protocols RTSP and SDP are not necessary to describe in detail here to understand the embodiments below.

With reference to FIGURE 2, illustrating a flow chart with steps executed in a client terminal, a procedure for enabling a communication session for streamed media data between the client terminal and a server in accordance with one embodiment will now be described. In a first **step 200**, an SDP checksum is generated based on SDP information available at the client terminal, by applying a checksum algorithm on the SDP information. The SDP checksum may be generated based on the available entire SDP information, or based on some specific parts of the SDP information. For instance, media information and/or attribute information of the SDP information may be used. In a next **step 202**, the generated SDP checksum is sent together with a request-to-establish message from the client terminal to the server, to be compared to a correct SDP checksum generated in the server for SDP information available at the server. The request-to-establish message may be realised by a <SETUP> message.

Then in a following **step 204**, a response message is received from the server, indicating whether the SDP checksum

sent to the server is correct or not. Typically, if the sent SDP checksum is correct, the response message may be realised by an <OK> message. On the other hand, if the sent checksum is incorrect, the response message comprises a message indicating that the SDP checksum is incorrect. If the comparison performed by the server indicates that the SDP checksum received from the client terminal is incorrect; the response message will further be supplemented with the correct SDP information from the server. The received correct SDP information may comprise the entire SDP information or just the correction of the incorrect parts of the SDP information. In a further **step 206**, it is decided whether the available SDP information at the client terminal should be updated or not. The decision is based on the response message received in step 204. If the response message indicates that the sent SDP checksum is incorrect and is further supplemented with the correct SDP information, the available SDP information at the client terminal will be updated with the received correct SDP.

If it is decided in step 206 not to update the available SDP information at the client terminal, a communication session for streamed media is established between the client terminal and the server in a following **step 208**, the communication session being based on the SDP information already available at the client terminal. The client terminal then starts to receive streamed media from the server.

If it is instead decided in step 206 to update the SDP information, then in another **step 210** the available SDP information is updated with the correct SDP information received with the response message in step 204. Finally, the steps 200-206 are repeated, using the updated SDP information now available at the client terminal.

In an optional **step 212**, illustrated with a dashed line, the client may also request to establish one or more additional

communication sessions, simultaneously with an established communication session. In that case, the client then repeats the steps 200-206, regarding the additional communication session(s). For instance, if an audio session is established, a video session or a further audio session may be established.

Optionally, the client terminal may also send an initiating request message for a desired service supplemented with the SDP information available at the client terminal, and then receive a response message regarding the desired service, before the server starts sending the streamed media data in step 208. The initiating request message may be realised by a <PLAY> message. The messages: <SETUP>, <OK>, <PLAY>, etc., are commonly used messages while establishing sessions for streamed media, and are not necessary to describe in more detail here.

Although this exemplary embodiment describes a procedure where a client terminal requests to establish a communication session and waits for a response message before requesting a desired streamed media service, the invention is not limited thereto. The described procedure can, as realised by one skilled in the art, easily be adapted to be applied to any communication session for streamed media where a client terminal and a server exchanges session description information at the establishment of the session or during the session, e.g. for establishing a pipelined communication session, or a during a fast content switching session.

With reference to FIGURE 3, illustrating a flow chart with steps executed in a server, a method of enabling a communication session for streamed media between a client terminal and the server in accordance with one embodiment will now be described. In a first **step 300**, a request-to-establish message, supplemented with an SDP checksum is received from the client terminal. The received SDP checksum has been generated based on SDP information

available at the client terminal. In a next **step 302**, a correct SDP checksum is generated based on correct SDP information available at the server. The received SDP checksum is then, in a following **step 304**, compared to the generated correct SDP checksum.

If the result of the comparison indicates that the received SDP checksum is correct, then in a further **step 306**, a response message is sent to the client terminal, confirming the received SDP checksum. In a following **step 308**, the communication session is then established and the server can start sending streamed media over the established communication session applying the correct SDP information.

On the other hand, if the result of the comparison indicates that the received SDP checksum is incorrect, then in a further **step 310**, a response message supplemented with the correct SDP information is sent to the client terminal, to update the SDP information available at the client terminal. The correct SDP information may comprise the entire SDP information or just the corrected parts. The server then repeats the steps 300-304, based on the updated SDP information.

With reference to the FIGURES 4a and 4b, illustrating signalling diagrams with messages exchanged between a client terminal 400 and a server 402, procedures of enabling a communication session for streamed media between the client terminal and the server in accordance with two further embodiments will now be described. Figure 4a, illustrates the case where the SDP information available at the client terminal is correct, when figure 4b instead illustrates the case where the SDP information is incorrect.

According to the procedure illustrated in FIGURE 4a, in a first **step 4:1a**, the client terminal 400 who wants to establish a communication session with the server 402 for streamed media,

generates an SDP checksum based on SDP information available at the client terminal. The SDP checksum may be generated based on the entire SDP information, or on some specific parts of the SDP information, e.g. media information and/or attribute information. In a next **step 4:2a**, the client terminal sends a request-to-establish message to the server to initiate the establishment of the communication session, the request-to-establish message being supplemented with the SDP checksum generated in step 4:1a. As described above, the request-to-establish message may be realised by a <SETUP> message. The complete message may be: <SETUP>, <SDP-Available: "SDP checksum">.

In a following **step 4:3a**, performed in the server, another SDP checksum is generated based on the correct SDP information available at the server. The received SDP checksum is then compared to the correct SDP checksum.

In a further **step 4:4a**, a response message indicating the result from the comparison performed in step 4:3a is sent from the server to the client terminal. In this embodiment the compared checksums are identical, and the response message will therefore confirm that the available SDP information at the client terminal is correct. The response message may be realised by an <OK> message.

Optionally, the client terminal may also send an initiating request message for a desired service to the server, supplemented with the SDP information available at the client terminal, and receive a response message regarding the desired service, before the server starts sending the streamed media data. This is performed in the intermediate **steps 4:5a** and **4:6a**, respectively, illustrated by dashed lines. The initiating request message for a desired service may be realised by a <PLAY> message, supplemented with the SDP checksum based on the SDP information available at the client terminal. The complete message may be: <PLAY>, <SDP-

Available: "SDP checksum">. The response message may be realised by an <OK> message.

In a final **step 4:7a**, a communication session is established and the server starts sending the streamed media to the client terminal.

According to the procedure illustrated in FIGURE 4b, in a first **step 4:1b**, the client terminal 400 who wants to establish a communication session with a server 402 for streamed media, generates an SDP checksum based on SDP information available at the client terminal. The SDP checksum may be generated based on the entire SDP information, or on some specific parts of the SDP information. In a next **step 4:2b**, the client terminal sends a request-to-establish message to the server to initiate the establishment of the communication session, the request-to-establish message being supplemented with the SDP checksum generated in step 4:1b. As described above, the request-to-establish message may be realised by a <SETUP> message. In a following **step 4:3b**, performed in the server, another SDP checksum is generated based on the correct SDP information available at the server. The received SDP checksum is then compared to the correct SDP checksum. The steps 4:1b-4:3b corresponds to the steps 4:1a-4:3a in the above described embodiment, but are performed for incorrect SDP information available at the client terminal.

In a further **step 4:4b**, a response message indicating the result from the comparison performed in step 4:3b is sent from the server to the client terminal. In this embodiment the compared checksums are not identical, and the response message will therefore indicate that the available SDP information at the client terminal is incorrect. The response message is further supplemented with the correct SDP information from the server. The sent correct SDP information may comprise the entire SDP

information or just the corrected parts. The complete message may be: <Incorrect SDP checksum>, <Content length 295>, <"SDP">.

In a following **step 4:5b**, the client terminal updates the available SDP information with the received correct SDP information. The client terminal makes then a new attempt to establish the communication session using the updated SDP information, by in a further **step 4:6b**, generating an SDP checksum based on the updated SDP information, before sending a new request-to-establish message to the server supplemented with the generated SDP checksum, in a following **step 4:7b**. In another **step 4:8b**, performed in the server, another SDP checksum is generated based on the correct SDP information available at the server. The received SDP checksum is then compared to the correct SDP checksum.

In a further **step 4:9b**, a response message indicating the result from the comparison performed in step 4:8b is sent to the client terminal. In this embodiment the compared checksums are identical, and the response message will therefore confirm that the updated available SDP information at the client terminal is correct. The response message may be realised by an <OK> message. The steps 4:6b-4:9b corresponds to the steps 4:1b-4:4b, but are performed for updated SDP information at the client terminal.

As described in the embodiment above, the client terminal may in addition also send an initiating request message for a desired service to the server supplemented with the SDP information available at the client terminal, and receive a response message regarding the desired service, before the server starts sending the streamed media. This is performed in the intermediate **steps 4:10b** and **4:11b**, respectively, illustrated by dashed lines. Analogously to steps 4:5a described in the embodiment above, the initiating request message for a desired service may be realised by a <PLAY> message, supplemented with

the SDP checksum based on the updated SDP information available at the client terminal. Furthermore, the response message may be realised by an <OK> message.

In a final **step 4:12b**, a communication session is established and the server starts sending the streamed media to the client terminal.

The above described procedures can easily be adapted to be applied to any communication session for streamed media where a client terminal and a server exchanges session description information at the establishment of the session or during the session.

In an alternative embodiment, different from the above described, the entire available SDP information at the client terminal, or a part of the entire SDP information may be sent to the server instead of an SDP checksum. In the server in this embodiment the entire SDP information or a part of the SDP information is compared to the SDP information available in the server.

With reference to FIGURE 5, an embodiment of a client terminal 500 for enabling a communication session 512 for streamed media with a server will now be described. The client terminal 500 comprises a session description information unit 502, a checksum generator 504, a communication unit 506, and media receiving unit 510. The session description information unit 502 is adapted to output its available SDP information to the checksum generator 504 when the client terminal 500 establishes a communication session with a server for streamed media. The checksum generator 504 is adapted to generate an SDP checksum based on the received SDP information and to output the generated SDP checksum to the communication unit 506. The checksum generator 504 may generate the SDP checksum based on the entire available SDP information, or just some part(s) of it,

e.g. from media information and/or attribute information. The communication unit 506 is adapted to send a request-to-establish message, supplemented with the received SDP checksum to the server over a communication link 508.

The communication unit 506 is further adapted to receive a response message from the server over the communication link 508, in response to the request-to-establish message, the response message indicating whether the sent SDP checksum is correct or incorrect. The SDP checksum can be incorrect due to various reasons, e.g. the SDP checksum is affected by transmission errors during the transmission from the client terminal 500 to the server, or the SDP checksum can be generated based on SDP information being out of date, etc.

The communication unit 506 is further adapted to establish a communication session 512 with the server and to start receiving streamed media from the server over the established communication session 512, if the response message indicates that the sent SDP checksum is correct, the communication session being based on the available SDP information in the session description information unit 502. Furthermore, the communication unit 506 is adapted to supply the received streamed media to the media receiver 510. The client terminal 500 is also adapted to send an initiating request message for a desired service to the server over the communication link 508, before the server starts to send the streamed media over the established communication session 512. The initiating request message may be supplemented with the generated SDP checksum.

Furthermore, the communication unit 506 is adapted to receive the correct SDP information from the server over the communication link 508, if the received response message indicates that the sent SDP information is incorrect. The communication unit 506 is further adapted to output the correct

SDP information to the session description information unit 502, which is further adapted to update its available stored SDP information, and output the updated SDP information to the checksum generator 504. The checksum generator 504 is further adapted to generate an updated SDP checksum based on the updated SDP information, and the communication unit 506 is further adapted to send a new request-to-establish message supplemented with the updated SDP checksum.

The client terminal may also be adapted to request to establish one or more additional communication sessions, simultaneously with an established communication session. For instance, if an audio session is established, a video session or a further audio session may be established.

Optionally, the session description information unit 502 may be adapted to output the entire SDP information or just some part(s) of it to the communication unit 506. The communication unit 506 may also be adapted to send the received SDP information the server.

With reference to FIGURE 6, an embodiment of a server 600 for enabling a communication session 614 for streamed media with a client terminal will now be described. The server 600 comprises a session description information unit 602, a checksum generator 604, a communication unit 606, a determining unit 608, and a media sending unit 610. When the client terminal establishes a communication session 614 with the server for streamed media, the communication unit 606 is adapted to receive a request-to-establish message supplemented with an SDP checksum from the client terminal over a communication link 612, the received SDP checksum being generated based on the SDP information available at the client terminal. The communication unit 606 is further adapted to supply the received SDP checksum to a first input of the determining unit 608. The session description unit 602 is

adapted to output the correct SDP information to the checksum generator 604, which is adapted to generate a correct SDP checksum and supply to a second input of the determining unit 608. The determining unit 608 is adapted to compare the SDP checksum received from the client terminal to the correct SDP checksum, and output the result from the comparison to the session description information unit 602.

The session description unit 602 is further adapted to initiate the communication unit 606 to send a response message to the client terminal when receiving the result from the comparison, the response message indicating the result from the comparison. Furthermore, the session description information unit 602 is adapted to supply the correct SDP information to the communication unit 606, to be sent over the communication link 612 supplemented to the response message, if the result of the comparison indicates that the SDP checksum received from the client terminal is incorrect.

The communication unit 606 is further adapted to establish the communication session 614 and to start sending streamed media to the client terminal over the established communication session 614, based on the available SDP information in the session description information unit 602, if the result of the comparison instead indicates that the received SDP checksum is correct. Moreover, the communication unit 606 is adapted to receive the media to be sent from the media sender 610.

Optionally, the server 600 may also be adapted to receive an initiating request message from the client terminal over the communication link 612, before the server starts to send the streamed media over the established communication session 614. The initiating request message may be supplemented with the SDP checksum generated in the client terminal.

Optionally, the session description information unit 602

may be adapted to output the entire SDP information or just some part(s) of it to the determining unit 608. The communication unit 606 may also be adapted to receive the entire SDP information or just some part(s) of it from the client terminal, and supply to the determining unit 608. The determining unit may also be adapted to compare SDP information instead of SDP checksums.

The described client terminal 500 and server 600 can easily, as is realised by one skilled in the art, be adapted to be applied for any communication session for streamed media data where a client terminal and a server exchanges session description information at the establishment of the session or during the session, e.g. for establishing a pipelined communication session, or a during a fast content switching session.

Furthermore, it is to be understood that the client terminal and the server described above in this description also comprises additional conventional means providing functionality, such as e.g. various control units and memories, necessary for enabling common functions and features to operate properly. However, for simplicity reasons, any means or functionality which is not necessary for the understanding of the proposed establishing of a communication session for streamed media has been omitted in the figures, and will not be discussed in any further detail in this description.

By means of the present invention, a fast and reliable method for enabling a communication session for streamed media data from a server to a client terminal is obtained. Using the above described solution, the enabling of the communication session can be performed by applying session description information distributed to the client terminal on beforehand. Moreover, the user of the client terminal does not run the risk of using session description information, being out of date or

being affected by various failures.

While the invention has been described with reference to specific exemplary embodiments, the description is in general only intended to illustrate the inventive concept and should not be taken as limiting the scope of invention. Although a client terminal and a media server have been used throughout when describing the above embodiments, any other communication terminal suitable for communicating streamed media, may be used in the manner described.

The invention is generally defined by the following independent claims.

CLAIMS

1. A method for enabling a communication session for streamed media between a client terminal and a server, comprising the following steps being executed in the client terminal:

a) sending (202) a request-to-establish message to the server, supplemented with session description information available at the client terminal, and

b) receiving (204) a response message from the server indicating whether the session description information is correct or incorrect, the response message being supplemented with the correct session description information if the sent session description information is incorrect,

wherein, if the response message indicates that the sent session description information is correct, the session is executed by receiving (208) media data from the server.

2. The method according to claim 1, wherein, if the response message indicates that the sent session description information is incorrect, the following further steps are executed:

c) updating (210) the session description information available at the client terminal based on the correct session description information received with the response message, and

d) repeating the steps a-b using the updated session description information to enable execution of the session.

3. The method according to claim 1 or 2, wherein the further step a' is executed before step a:

   a') generating (200) a checksum based on at least a subset of session description information available at the client terminal,

   and the session description information sent in step a) is the generated checksum.

4. The method according to claim 3, wherein the checksum is generated based on media information in the session description information and/or attribute information in the session description information.

5. The method according to claim 3, wherein the checksum is generated based on the entire session description information, the entire session description information is received from the server if the checksum is incorrect, and further the entire session description information is updated in step c.

6. The method according to claim 1 or 2, wherein the session description information supplemented to the request-to-establish message is the entire session description information or some part(s) of the session description information.

7. The method according to any of claims 1-6, wherein the streamed media is received according to the RTSP and the session description information is formatted in accordance with the SDP.

8. A method for enabling a communication session for streamed media between a client terminal and a server, comprising the following steps being executed in the server:

a) receiving (300) a request-to-establish message supplemented with a checksum based on at least a subset of session description information available at the client terminal,

b) determining (304) whether the received checksum is correct or incorrect,

c) sending (306, 310) a response message indicating whether the checksum is correct or incorrect to the client terminal, and further if the checksum is incorrect supplementing (310) the response message with the correct session description information available at the server,

wherein, if the response message indicates that the checksum is correct, the session is executed by sending (308) media data to the client terminal, and

wherein, if the response message indicates that the checksum is incorrect, the further step is executed:

d) repeating the steps a-c using the updated session description information to enable execution of the session.

9. A method for enabling a communication session for streamed media between a client terminal and a server, comprising the following steps:

a) generating (4:1a, 4:1b, 4:6b) in the client terminal a checksum based on at least a subset of session description information available at the client terminal,

b) sending (4:2a, 4:2b, 4:7b) a request-to-establish message, supplemented with the checksum from the client terminal to the server,

c) determining (4:3a, 4:3b, 4:8b) in the server whether the received checksum is correct or incorrect, sending (4:4a, 4:4b, 4:9b) a response message from the server to the client terminal, and further if the checksum is incorrect supplementing the response message with the correct session description information available at the server,

wherein, if the response message indicates that the checksum is correct, the session is executed by sending (4:7a, 4:12b) media data from the server to the client, and

wherein, if the response message indicates that the checksum is incorrect the following steps are executed:

d) updating (4:5b) the session description information available at the client terminal based on the correct session description information received with the response message, and

e) repeating the steps a-c using the updated session description information.

10. The method according to claim 9, wherein the determination if the checksum is correct or incorrect is performed by generating a correct checksum based on at least a subset of correct session description information available at the server, and comparing the received checksum to the generated checksum.

11. A client terminal (500) for enabling a communication session (512) for streamed media with a server, comprising:

a) a session description information unit (502) adapted to store session description information available at the client terminal,

b) a communication unit (506) adapted to send a request-to-establish message supplemented with the stored session description information via a communication link (508) to the server, and further adapted to receive a response message from the server via the communication link (508),

the session description information unit (502) being further adapted to update the stored session description information when correct session description information is received.

12. The client terminal according to claim 11, further comprising a media receiving unit (510), and wherein the communication unit (506) is further adapted to receive streamed media from the server and supply to the media receiving unit (510).

13. The client terminal according to claim 11 or 12, wherein:

- the communication unit (506) is further adapted to receive the streamed media according to the RTSP,
- the session description information unit (502) is further adapted to store the session description information formatted in accordance with the SDP.

14. The client terminal according to any of claims 11-13, further comprising a checksum generator (504) is further adapted to generate a checksum based on at least media information in the session description information and/or attribute information in the session description information.

15. The client terminal according to claim 14, wherein the checksum generator (504) is further adapted to generate the checksum based on the entire session description information available at the client terminal, the session description

information unit (502) is further adapted to store the entire session description information and to update the entire session description information available at the client terminal when the response message is provided with the correct session description information.

16. A server (600) for enabling a communication session (614) for streamed media with a client terminal, comprising:

a) a session description information unit (602) adapted to store session description information,

b) a communication unit (606) adapted to receive a request-to-establish message supplemented with session description information available at the client terminal via a communication link (610), and further adapted to send a response message to the client terminal via the communication link, and

c) a determining unit (608) adapted to determine if the received session description information is correct, and output the result to the session description information unit (602),

the session description information unit (602) being further adapted to generate the response message, and supplement the response message with the correct session description information if the determining unit (608) indicates that the received session description information is incorrect.

17. The server according to claim 16, further comprising a checksum generator (604) adapted to generate a correct checksum based on at least a subset of the session description information stored in the session description information unit

(602), and wherein the determining unit (608) is further adapted to perform the determination by comparing the received checksum to the correct checksum.

18. The server according to claim 17, wherein the checksum generator (604) is further adapted to generate the checksum based on media information in the session description information and/or attribute information in the session description information.

19. The server according to any of claims 17-18, wherein the checksum generator (604) is further adapted to generate the checksum based on the entire session description information available at the server, the session description information unit (602) is further adapted to store the entire session description information, and to supply the entire session description information to the communication unit (606) to be supplemented to the response message.

20. The server according to any of claims 16-19, further comprising a media sending unit (612), wherein the communication unit (606) is further adapted to receive streamed media from the media sending unit (612) and send the received streamed to the client terminal.

21. The server according to any of claims 16-20, wherein:

- the communication unit (606) is further adapted to communicate the streamed media according to the RTSP, and

- the session description information unit (602) is further adapted to store the content of session description information being formatted in accordance with the SDP.

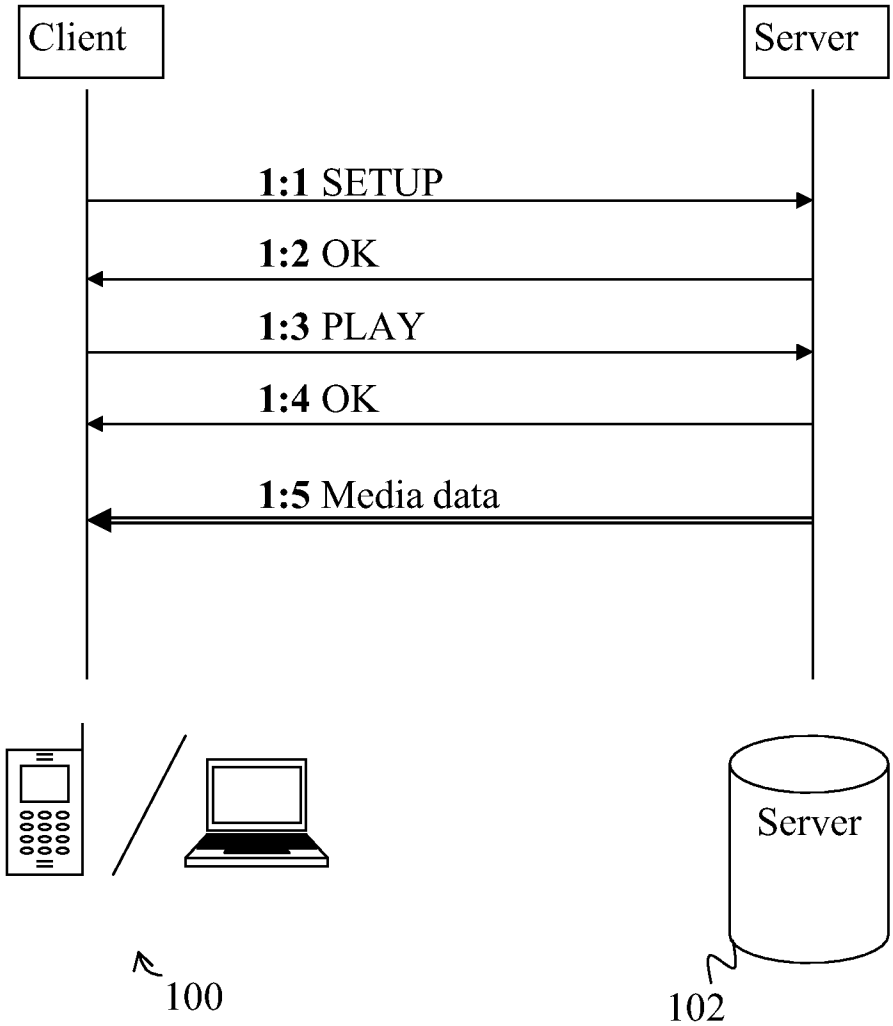


Figure 1 (prior art)

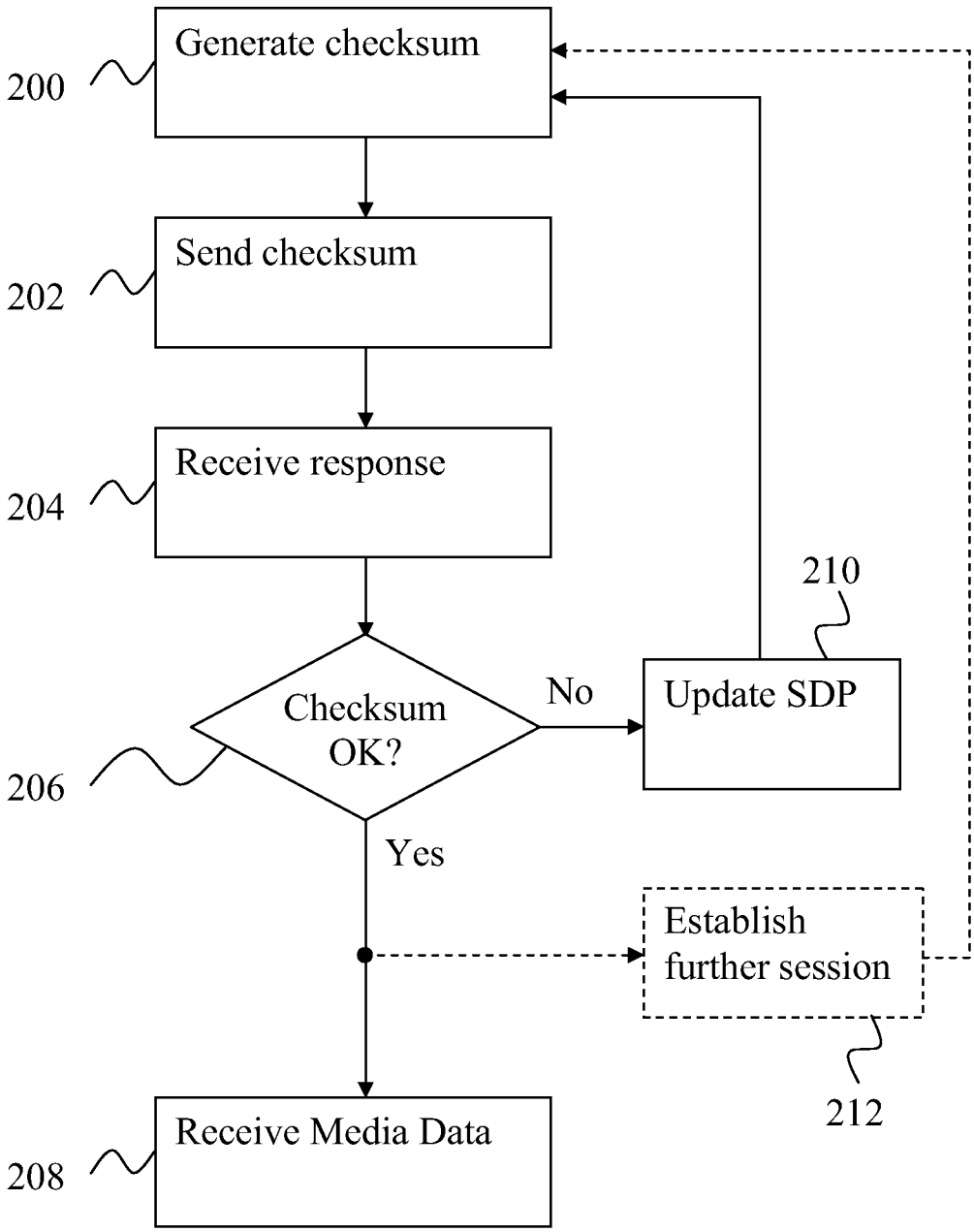

Generate checksum
200
Send checksum
202
Receive response
204
210
Checksum OK?
206
No
Update SDP
Yes
Establish further session
212
Receive Media Data
208

Figure 2

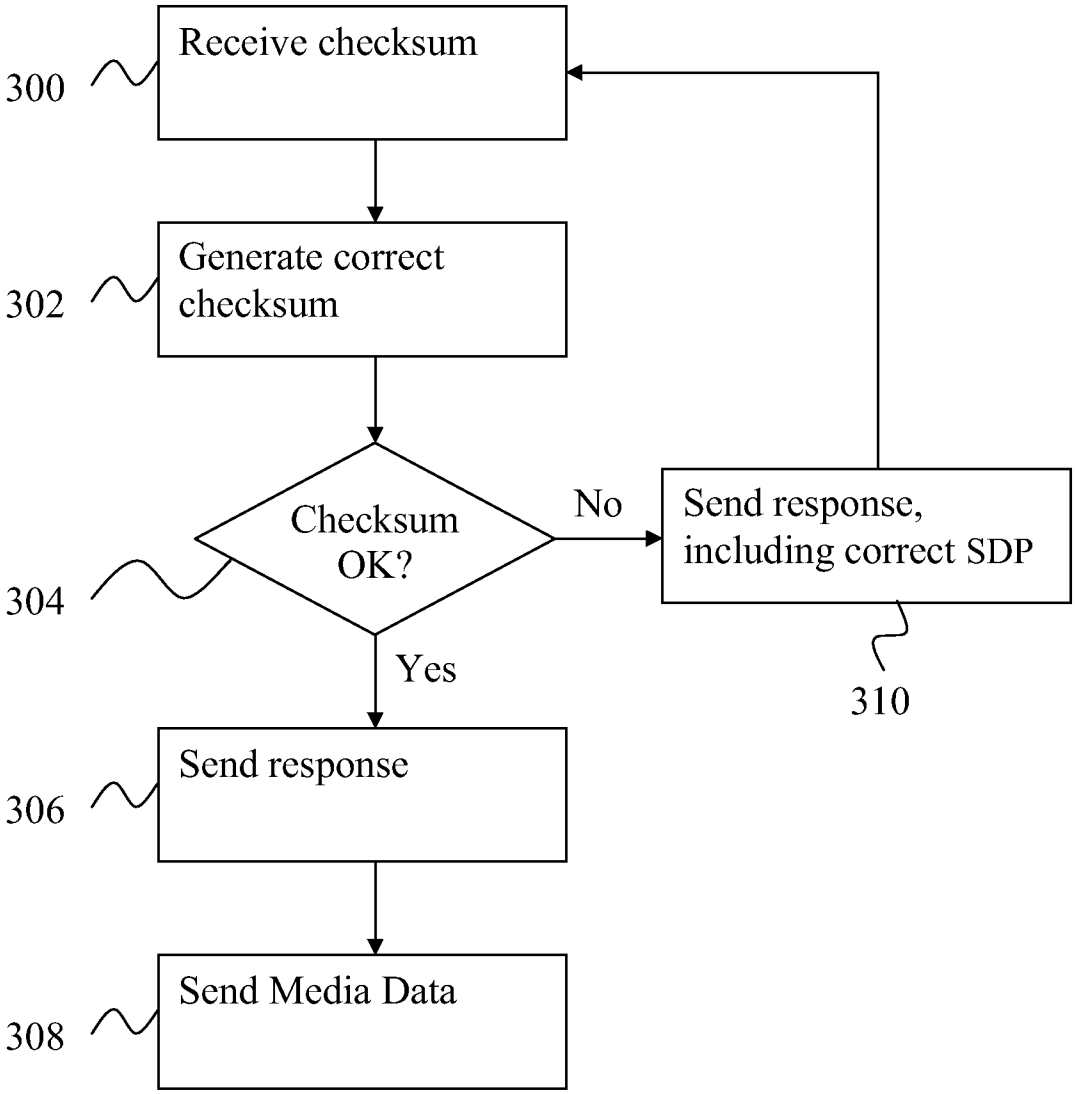

Receive checksum
300
Generate correct checksum
302
Checksum OK?
304
No
Send response, including correct SDP
310
Yes
Send response
306
Send Media Data
308

Figure 3

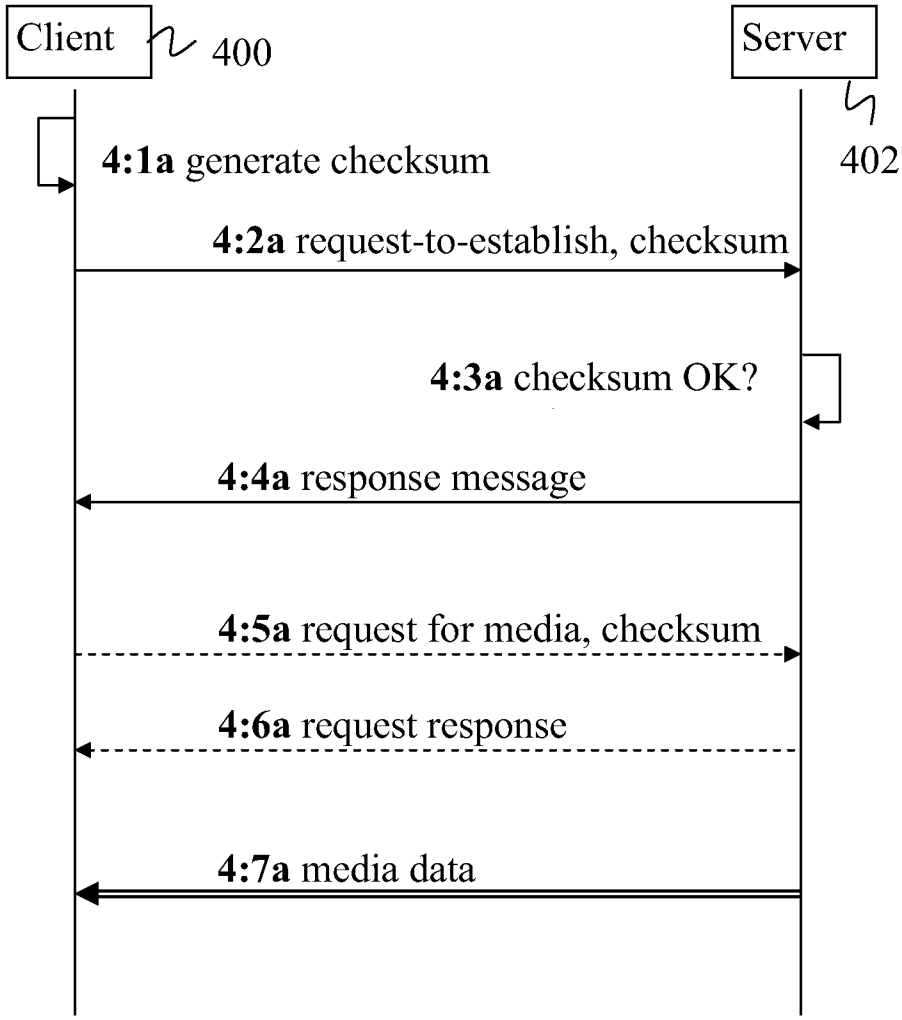

Client
400
Server
402
4:1a generate checksum
4:2a request-to-establish, checksum
4:3a checksum OK?
4:4a response message
4:5a request for media, checksum
4:6a request response
4:7a media data

Figure 4a

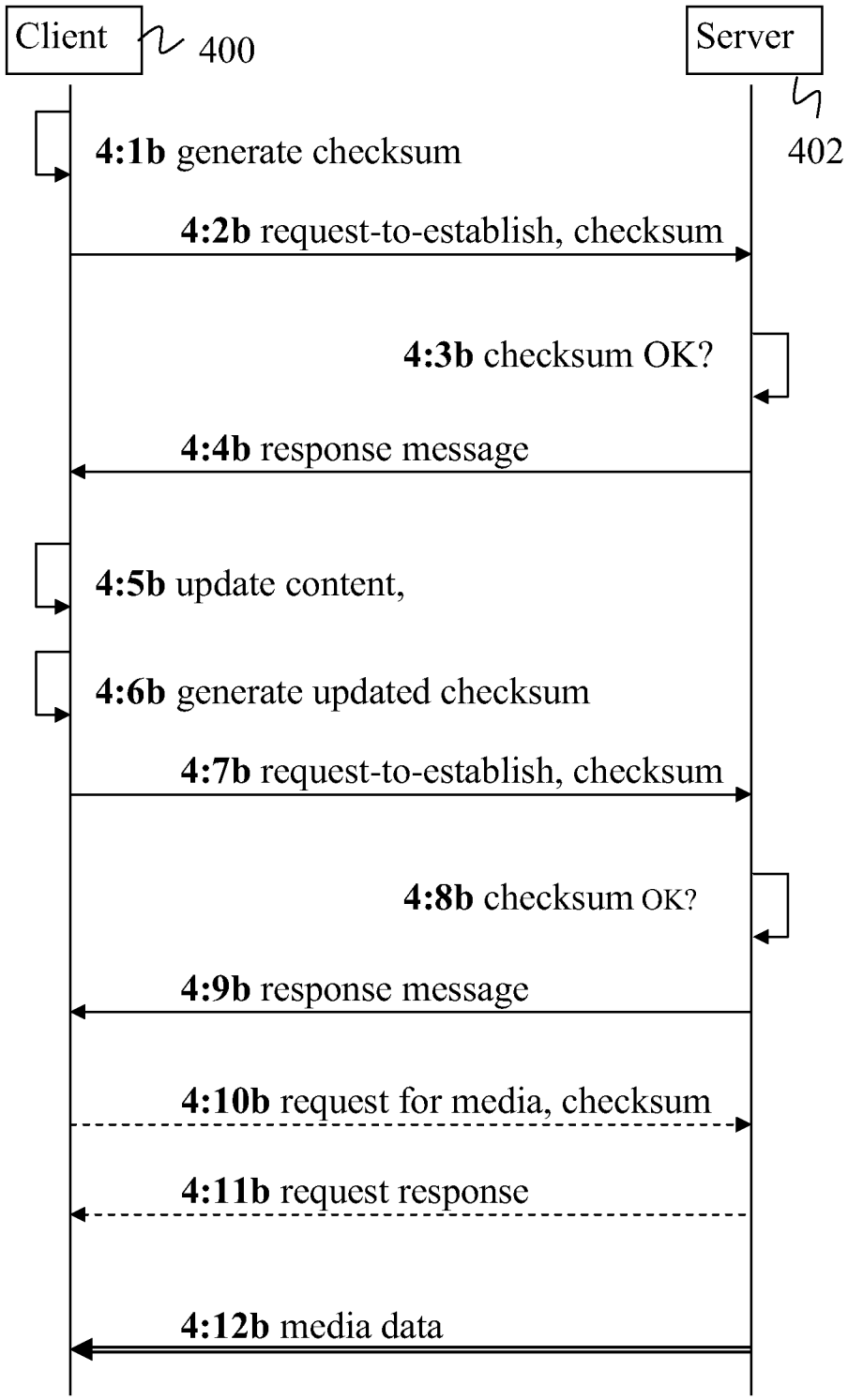

Client
400
Server
402
4:1b generate checksum
4:2b request-to-establish, checksum
4:3b checksum OK?
4:4b response message
4:5b update content,
4:6b generate updated checksum
4:7b request-to-establish, checksum
4:8b checksum OK?
4:9b response message
4:10b request for media, checksum
4:11b request response
4:12b media data

Figure 4b

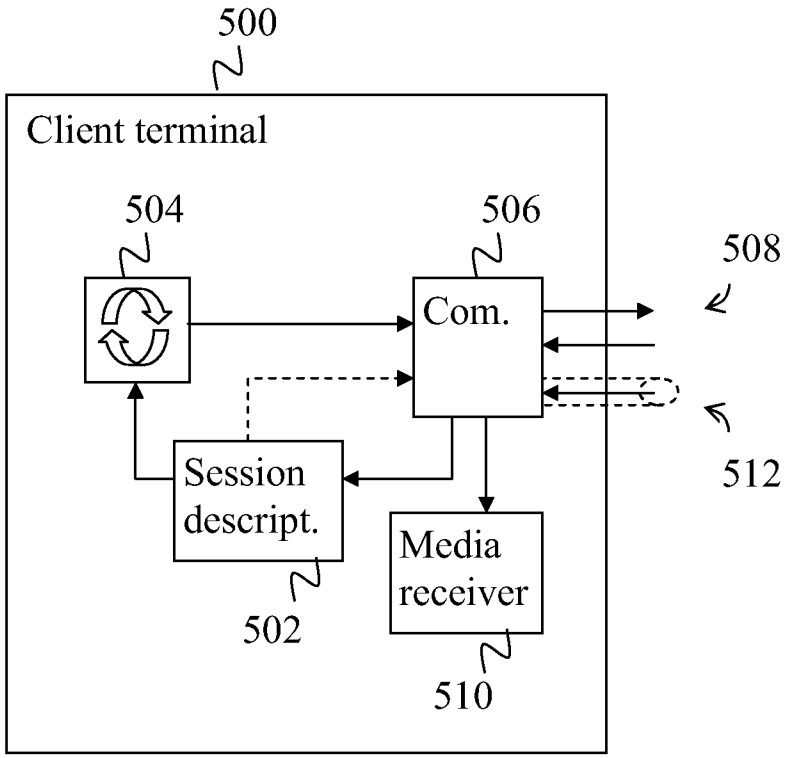

500
Client terminal
504
506
508
Com.
512
Session descript.
Media receiver
502
510

Figure 5

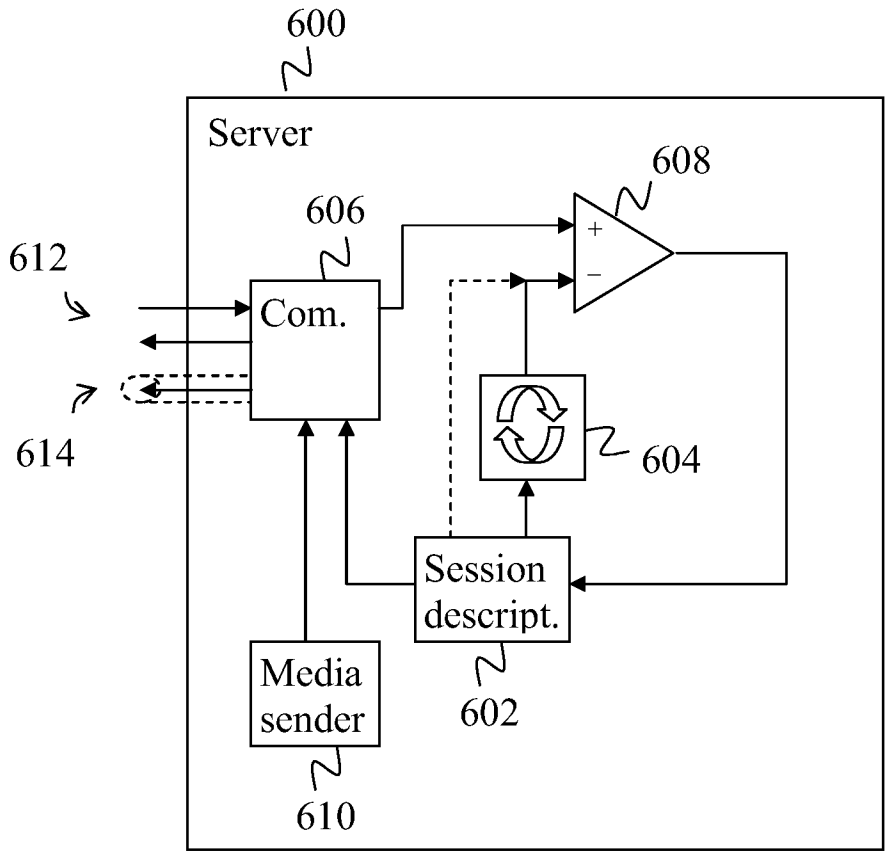

600
Server
608
606
612
Com.
614
604
Session
descript.
Media
sender
602
610

Figure 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/SE2008/050556

A. CLASSIFICATION OF SUBJECT MATTER
INV. H04L29/06 H04L29/08

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

B. FIELDS SEARCHED

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

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

Electronic data base consulted during the international search (name of data base and, where practical, 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.
A	WO 02/51178 A (SONERA OYJ [FI]; KOSKELA MARKKU [FI]) 27 June 2002 (2002-06-27) the whole document	1-21
A	US 2004/148400 A1 (MOSTAFA MIRAJ [FI]) 29 July 2004 (2004-07-29) page 20, line 11 - line 29	1-21
A	WO 02/080590 A (QUALCOMM INC [US]) 10 October 2002 (2002-10-10) paragraph [1108]	1-21
A	WO 01/33418 A (NOKIA NETWORKS OY [FI]; NOKIA INC [US]) 10 May 2001 (2001-05-10) paragraph [0147]	1-11

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents :

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

"E" earlier document 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
30 September 2008	07/10/2008

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, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Megalou-Nash, M

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/SE2008/050556

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 0251178 A	27-06-2002	AU 1926502 A	01-07-2002
		FI 20002821 A	22-06-2002
US 2004148400 A1	29-07-2004	NONE	
WO 02080590 A	10-10-2002	BR PI0208433 A	21-02-2006
		CN 1507730 A	23-06-2004
		EP 1374528 A2	02-01-2004
		JP 2005507573 T	17-03-2005
		TW 236822 B	21-07-2005
WO 0133418 A	10-05-2001	AT 401610 T	15-08-2008
		AU 7032800 A	14-05-2001
		EP 1226518 A2	31-07-2002
		US 7139813 B1	21-11-2006