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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 March 2017 (30.03.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/052507 A1

(51) International Patent Classification:
*H04L 29/06* (2006.01) *H04L 12/801* (2013.01)

(21) International Application Number:
PCT/US2015/051379

(22) International Filing Date:
22 September 2015 (22.09.2015)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP** [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).

(72) Inventors: **VIQUEZ CALDERON, Claudio Enrique**; Ultra Park, La Aurora, Heredia, Building 8, San Jose (CR). **VALVERDE GARRO, Diego**; Ultra Park, La Aurora, Heredia, Building 8, San Jose (CR). **HERNANDEZ VARGAS, Jose Daniel**; Ultra Park, La Aurora, Heredia, Building 8, San Jose (CR). **SANCHEZ MELENDEZ, Osvaldo Andres**; Ultra Park, La Aurora, Heredia, Building 8, San Jose (CR).

(74) Agents: **FERGUSON, Christopher, Ward** et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (US).

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

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

**Declarations under Rule 4.17**:

— *as to the identity of the inventor (Rule 4.17(i))*

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

*[Continued on next page]*

(54) Title: ENCRYPTED DATA PACKET

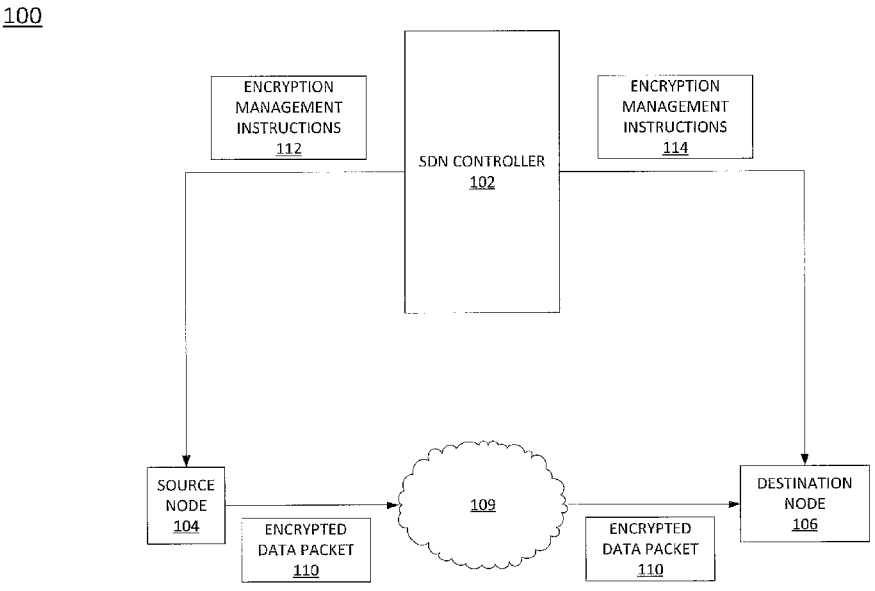



FIG. 1

(57) Abstract: In example implementations, a method includes a software defined network (SDN) controller that selects an encryption key. The SDN controller then sends a first instruction to a source node to modify a flow table of the source node to include an action that includes the encryption key. A second instruction is sent by the SDN controller to a destination node to modify a flow table of the destination node to include an action that includes the encryption key. The SDN controller can then control a data packet that is encrypted by the source node with the encryption key to be sent from the source node to the destination node, wherein the data packet is to be decrypted with the encryption key by the destination node.

WO 2017/052507 A1

**Published**:

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

# ENCRYPTED DATA PACKET

## BACKGROUND

**[0001]** A software defined network (SDN) is an approach to computer networking that allows networks to be managed through higher level abstraction of the network. For example, in an SDN network, the data plane and the control plane are separated. An SDN controller may be used to manage each node in the network and manage the SDN network by controlling data traffic. SDN networks may use communication protocols to allow the control plane to communicate with the data plane.

## BRIEF DESCRIPTION OF THE DRAWINGS

**[0002]** FIG. 1 is a block diagram of an example communication network of the present disclosure;

**[0003]** FIG. 2 is a block diagram of an example node of the present disclosure;

**[0004]** FIG. 3 is a block diagram of an example SDN controller of the present disclosure;

**[0005]** FIG. 4 is a flow diagram of an example method for encrypting a data packet; and

**[0006]** FIG. 5 is a flow diagram of another example method for encrypting a data packet.

## DETAILED DESCRIPTION

**[0007]** The present disclosure broadly discloses a software defined network (SDN) controller that is modified to perform and control data encryption in SDN networks. As discussed above, SDN networks use an SDN controller to separate the data plane and control plane. SDN controllers are currently used to perform routing functions, but do not perform or control encryption functions.

**[0008]** Examples of the present disclosure provide a modification to the SDN controller and nodes in the SDN network to implement encryption management and control by the SDN controller. FIG. 1 illustrates an example SDN network 100. The SDN network 100 may include an SDN controller 102, a source node 104 and a destination node 106. It should be noted that although only a single SDN controller 102, a single source node 104 and a single destination node 106 are illustrated in FIG. 1, any number of SDN controllers, source nodes and destination nodes may be deployed in the SDN network 100. The SDN network 100 may use an Open Flow communication protocol to allow the SDN controller 102, the source node 104 and the destination node 106 to communicate with one another.

**[0009]** In one implementation, the source node 104 may send encrypted data packets 110 over an Internet Protocol (IP) network 109 to the destination node 106. It should be noted that the IP network 109 has been simplified for ease of explanation. For example, the IP network 109 may include additional network elements (e.g., routers, gateways, switches, firewalls, and the like) and access networks (e.g., a broadband access network, a cellular access network, and the like) that are not shown.

**[0010]** FIG. 2 illustrates a block diagram of an example of the source node 104 of the present disclosure. It should be noted that the destination node 106 may include similar hardware and modifications. The source node 104 may include a processor 202. In one example, the processor may be an application specific integrated circuit (ASIC) 202. The ASIC 202 may include a flow table 204 that is used with Open Flow communication protocol. It should be noted that although the flow table 204 is illustrated as being entirely in the ASIC 202, the flow table 204 may be partially or completely stored in different portions of

the SDN network 100 (e.g., the SDN controller 102).

**[0011]** In one example, the flow table 204 may include a plurality of match criteria 206-1 to 206-n (herein after referred to collectively as match criteria 206 or individually as a match criteria 206) and a plurality of actions 208-1 to 208-n (herein after referred to collectively as actions 208 or individually as an action 208). The match criteria 206 may include a tuple that is matched by a tuple of the data packet 201. If the tuple of the match criteria 206 match the tuple of the data packet 201, the action 208 that corresponds with the match criteria 206 may be performed. The tuple may include parameters, such as, a media access control (MAC) address, a source Internet Protocol (IP) address, a destination IP address, or any other parameters that can be found in a header field of the data packet 201.

**[0012]** Typically, the flow table 204 may include match criteria 206 to perform a routing action. However, the present disclosure modifies the flow table 204 to include a new action 208 to perform encryption of a data packet 201.

**[0013]** Referring back to FIG. 1, in one example, the SDN controller 102 may select an encryption key and an encryption function and send a first instruction 112 to the source node 104. The first instruction 112 may be an encryption management instruction that causes the source node 104 to modify the flow table 204 to include the encryption key and the encryption function that is selected in the action 208 associated with a match criteria 206. In other words, the SDN controller sends the actual encryption key that is used and stored in the flow table 204 and an identification of the encryption function that is selected to implement the correct encryption function.

**[0014]** In some implementations, the flow table 204 may include different encryption keys and different encryption functions in different actions 208 for different match criteria 206. For example, match criteria 206-1 may include a first encryption key and first encryption function in the action 208-1 and the match criteria 206-2 may include a second encryption key and a second encryption function in the action 208-2. Thus, the SDN controller 102 may manage and control encryption for a variety of different data packets 201 using a variety of different encryption keys and different encryption functions.

[0015] In one example, the source node 104 may also include encryption functions 210. The encryption functions 210 may be implemented as portion, or separate circuit/hardware configuration, in the ASIC 202. The encryption functions 210 may store the methods or techniques to allow the ASIC 202 to perform an encryption on the data packet 201 using the encryption function that is selected by the SDN controller 102 and the encryption key that is sent by the SDN controller 102. In one example, any type of encryption key or encryption function may be used. For example, the encryption functions may include a mask, a rotation, an addition, an XRO, and the like.

[0016] The SDN controller 102 may send a second instruction 114 to the destination node 106. The second instruction 114 may be an encryption management instruction that includes the same encryption key and same encryption function as the encryption key and the encryption function that were selected by the SDN controller 102 and sent to the source node 104. The destination node 106 may also be configured similar to the source node 104 illustrated in FIG. 2. In other words, the destination node 106 may also include an ASIC 202 that stores a flow table 204 and has encryption functions 210. The second instruction 114 may cause the destination node 106 to modify its flow table to include a match criteria and action that has the encryption key and the encryption function from the second instruction 114.

[0017] Subsequently, when the data packet 201 that matches the match criteria 206 arrives at the source node 104, the source node 104 may encrypt the data packet 201 into an encrypted data packet 110. The encrypted data packet 110 may be sent over the IP network 109 to the destination node 106. The destination node 106 may then match the encrypted data packet 110 to a match criteria in its flow table and decrypt the encrypted data packet 110 with the encryption key sent from the SDN controller 102.

[0018] It should be noted that when a plurality of source nodes 104 and a plurality of destination nodes 106 are deployed, that each source node 104 and each destination node 106 may have different match criteria 206 associated with actions 208 that each include different encryption keys and different selected encryption functions in the flow table 204 of each source node 104 and

each destination node 106. In other words, the SDN controller 102 has an overview of all the source nodes 104 and destination nodes 106 in the SDN network 100. As a result, the SDN controller 102 may send different encryption keys and select different encryption functions for different match criteria 206 for source nodes 104. Said another way, each flow table 204 of each source node 104 and destination node 106 may not have the same number of encryption keys and encryption functions or the same type of encryption keys and encryption functions.

**[0019]** In other words, the encryption keys and the encryption functions that are selected by the SDN controller 102 can be selectively distributed to source nodes 104 and destination nodes 106 by the SDN controller 102 based upon how data packets 201 are routed within the SDN network 100. As a result, memory space can be saved on the source nodes 104 and the destination nodes 106 as unused encryption methods need not be stored in the encryption functions 210 of respective source nodes 104 and destination nodes 106.

**[0020]** FIG. 3 illustrates a block diagram of an example SDN controller 102 of the present disclosure. In one example, the SDN controller 102 may include an input/output (I/O) interface 302. The I/O interface 302 may allow for connections to external devices (e.g., a monitor, a keyboard, and the like) for programming or configuring parameters of the SDN controller.

**[0021]** In one example, the SDN controller 102 may include a processor 304. The processor 304 may be a central processing unit (CPU), an application specific integrated controller (ASIC), a micro controller, and the like. The processor 304 may be in communication with the I/O interface 302 and a non-transitory computer readable storage medium 306. The processor 304 may execute the instructions stored in the non-transitory computer readable storage medium 306.

**[0022]** In one example, the non-transitory computer readable storage medium 306 may include instructions 308, 310, 312 and 314. The instructions 308 include instructions to select an encryption key and an encryption function. The instructions 310 include instructions to send a first instruction to a source node to modify a flow table of the source node to include an action that includes

the encryption key and the encryption function. The instructions 312 include instructions to send a second instruction to a destination node to modify a flow table of the destination node to include an action that includes the encryption key and the encryption function. The instructions 314 include instructions to route a data packet that is encrypted by the source node with the encryption key to be sent from the source node to the destination node, wherein the data packet is decrypted with the encryption key by the destination node.

**[0023]** FIG. 4 illustrates a flow diagram of an example method 400 for encrypting a data packet. In one example, the blocks of the method 400 may be performed by the SDN controller 102.

**[0024]** At block 402, the method 402 begins. At block 404, the method 400 selects an encryption key and an encryption function. For example, the encryption key and the encryption function may be selected based on security levels of certain types of data or security levels between certain source node and destination node combinations. For example, certain data packets may have a match criteria and an action having a low level encryption key and a low level encryption function, while more secure data packets may have a match criteria and an action having a high level encryption key and a high level encryption function.

**[0025]** In other implementations, certain customers may pay for a higher level of security. Thus, certain source nodes and/or destination nodes may require a higher level of encryption. The SDN controller 102 may select a strong encryption key and encryption function for those source nodes and destination nodes, while providing a weaker encryption key and encryption function for other source nodes and destination nodes.

**[0026]** At block 406, the method 400 sends a first instruction to a source node to modify a flow table of the source node to include a first action that includes the encryption key and the encryption function. For example, using an Open Flow communication protocol, the SDN controller may send the first instruction to the source node. The source node may modify its flow table in response to the first instruction.

**[0027]** At block 408, the method 400 sends a second instruction to a

destination node to modify a flow table of the destination node to include a second action that includes the encryption key and the encryption function. For example, using an Open Flow communication protocol, the SDN controller may send the second instruction to the destination node. The destination node may modify its flow table in response to the second instruction.

**[0028]** At block 410, the method 400 routes a data packet that is encrypted by the source node with the encryption key to be sent from the source node to the destination node, wherein the data packet is to be decrypted with the encryption key by the destination node. For example, a data packet that matches the match criteria for an action that requires encryption may be received by the source node. The SDN controller may manage the routes for data packets. Thus, after the data packet is encrypted, the encrypted data packet may be sent to the destination node as instructed by the flow table in the source node that was configured by routing instructions from the SDN controller. At block 412, the method 400 ends.

**[0029]** FIG. 5 illustrates a flow diagram of another example method 500 for encrypting a data packet. In one example, the blocks of the method 500 may be performed by the source node 104.

**[0030]** At block 502, the method 500 begins. At block 504, the method 500 receives an instruction from an SDN controller with an encryption key and an encryption function that are selected by the SDN controller. For example, the SDN controller may select an encryption key and an encryption function based on a type of data packet that the source node receives or based on a security level associated with the source node.

**[0031]** At block 506, the method 500 modifies a flow table to include a match criteria and an action to include the encryption key and the encryption function. For example, the match criteria may be added with the parameters provided in the instructions from the SDN controller. The match criteria may include, a MAC address, a source IP address, a destination IP address, or any other parameter that can be found in a header file of the data packet.

**[0032]** The action may include an encryption of the data packet with the encryption key and the encryption function. The encryption key may include, a

mask, a rotation, an addition, an XOR, and the like.

**[0033]** At block 508, the method 500 receives a data packet having a tuple that matches the match criteria. For example, the source node may identify the tuple associated with the data packet and compare the tuple to the tuple in the match criteria. If the parameters in the tuple of the data packet match the parameters of the tuple in the match criteria, then the action may be executed.

**[0034]** At block 510, the method 500 encrypts the data packet with the encryption key. In one implementation, the action associated with match criteria may be to encrypt the data packet with the encryption key using the encryption function. Thus, the source node may encrypt the data packet and then transmit the data packet across the IP network to the destination node.

**[0035]** In one example, the destination node may then decrypt the encrypted data packet using the encryption key and the encryption function received from the SDN controller via a second instruction to the destination node. The method 500 may be repeated for each data packet that arrives at the source node. At block 512, the method 500 ends.

**[0036]** It will be appreciated that variants of the above-disclosed and other features and functions, or alternatives thereof, may be combined into many other different systems or applications. Various presently unforeseen or unanticipated alternatives, modifications, variations, or improvements therein may be subsequently made by those skilled in the art which are also intended to be encompassed by the following claims.

CLAIMS

1. A method, comprising:

selecting, by a software defined network (SDN) controller, an encryption key and an encryption function;

sending, by the SDN controller, a first instruction to a source node to modify a flow table of the source node to include a first action that includes the encryption key and the encryption function;

sending, by the SDN controller, a second instruction to a destination node to modify a flow table of the destination node to include a second action that includes the encryption key and the encryption function; and

routing, by the SDN controller, a data packet that is encrypted by the source node with the encryption key to be sent from the source node to the destination node, wherein the data packet is to be decrypted with the encryption key by the destination node.

2. The method of claim 1, wherein the first instruction and the second instruction are sent via an Open Flow communication protocol.

3. The method of claim 1, wherein the first action is associated with a match criteria in the flow table of the source node.

4. The method of claim 1, wherein modification of the flow table of the source node causes an encryption functions of the source node to encrypt the data packet in accordance with the encryption function that is selected by the SDN controller using the encryption key sent by the SDN controller.

5. The method of claim 1, wherein the encryption function comprises a mask, a rotation, an addition, or an XOR.

6. An apparatus, comprising:

a processor; and

a non-transitory computer-readable storage medium comprising instructions that, when executed by the processor, cause the processor to:

select an encryption key and an encryption function;

send a first instruction to a source node to modify a flow table of the source node to include a first action that includes the encryption key and the encryption function;

send a second instruction to a destination node to modify a flow table of the destination node to include a second action that includes the encryption key and the encryption function; and

control a data packet that is encrypted by the source node with the encryption key to be sent from the source node to the destination node, wherein the data packet is to be decrypted with the encryption key by the destination node.

7. The apparatus of claim 6, wherein the first instruction and the second instruction are sent via an Open Flow communication protocol.

8. The apparatus of claim 6, wherein the first action is associated with a match criteria in the flow table of the source node.

9. A method, comprising:

receiving an instruction from a software defined network (SDN) controller with an encryption key and an encryption function that are selected by the SDN controller;

modifying a flow table to include a match criteria and an action to include the encryption key and the encryption function;

receiving a data packet having a tuple that matches the match criteria; and

encrypting the data packet with the encryption key.

10. The method of claim 9, wherein the encrypting is performed by an encryption functions.

11. The method of claim 9, wherein the flow table is stored in a programmable networking application specific integrated circuit (ASIC).

12. The method of claim 9, wherein the instruction is received via an Open Flow communication protocol.

13. The method of claim 9, wherein the instruction from the SDN controller further comprises parameters for the match criteria.

14. The method of claim 9, wherein the encryption function comprises a mask, a rotation, an addition, or an XOR.

15. The method of claim 9, wherein the data packet that is encrypted is to be decrypted by a destination node with the encryption key sent to the destination node by the SDN controller.

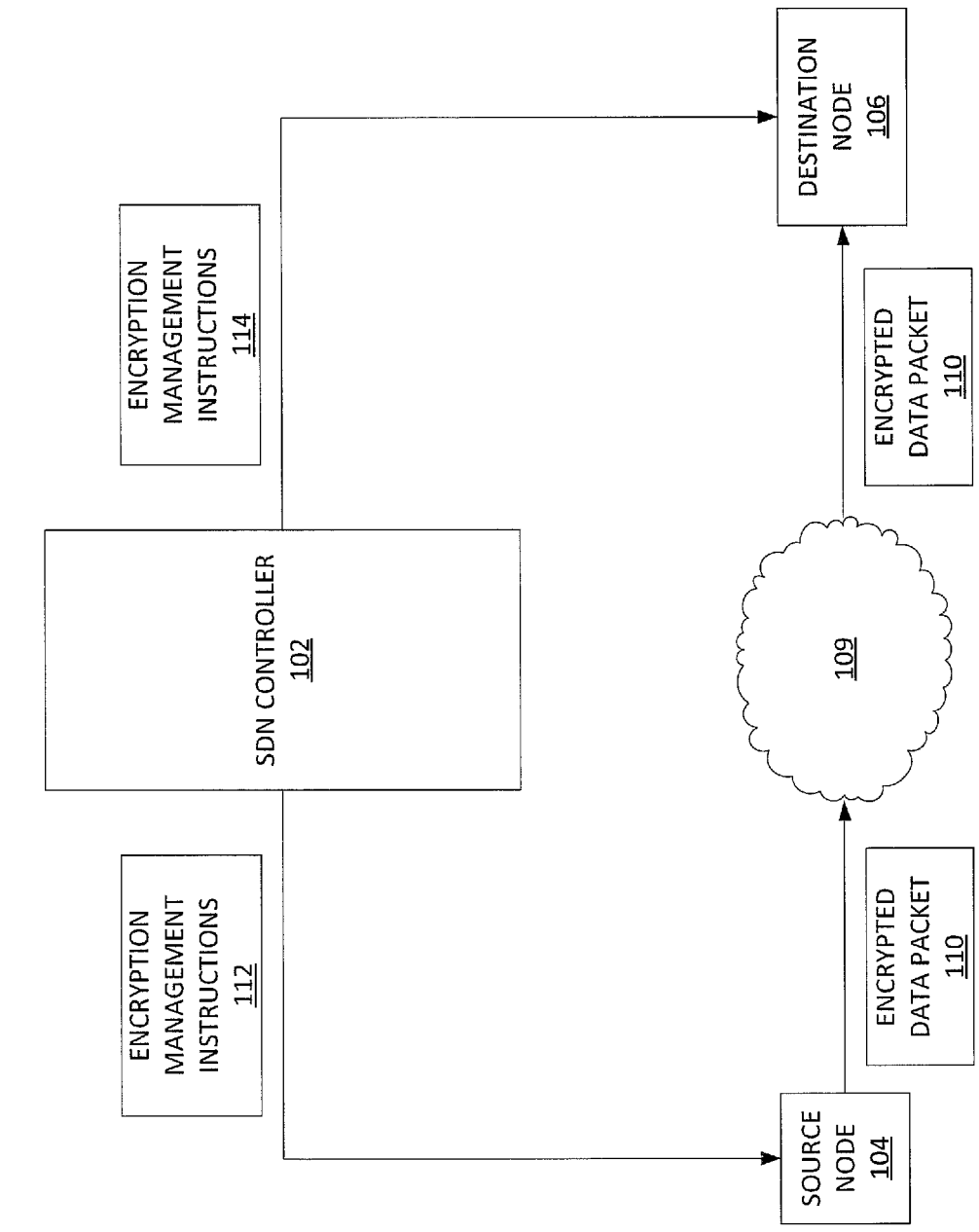

100
SOURCE NODE 104
ENCRYPTED DATA PACKET 110
109
ENCRYPTED DATA PACKET 110
DESTINATION NODE 106
SDN CONTROLLER 102
ENCRYPTION MANAGEMENT INSTRUCTIONS 112
ENCRYPTION MANAGEMENT INSTRUCTIONS 114

FIG. 1

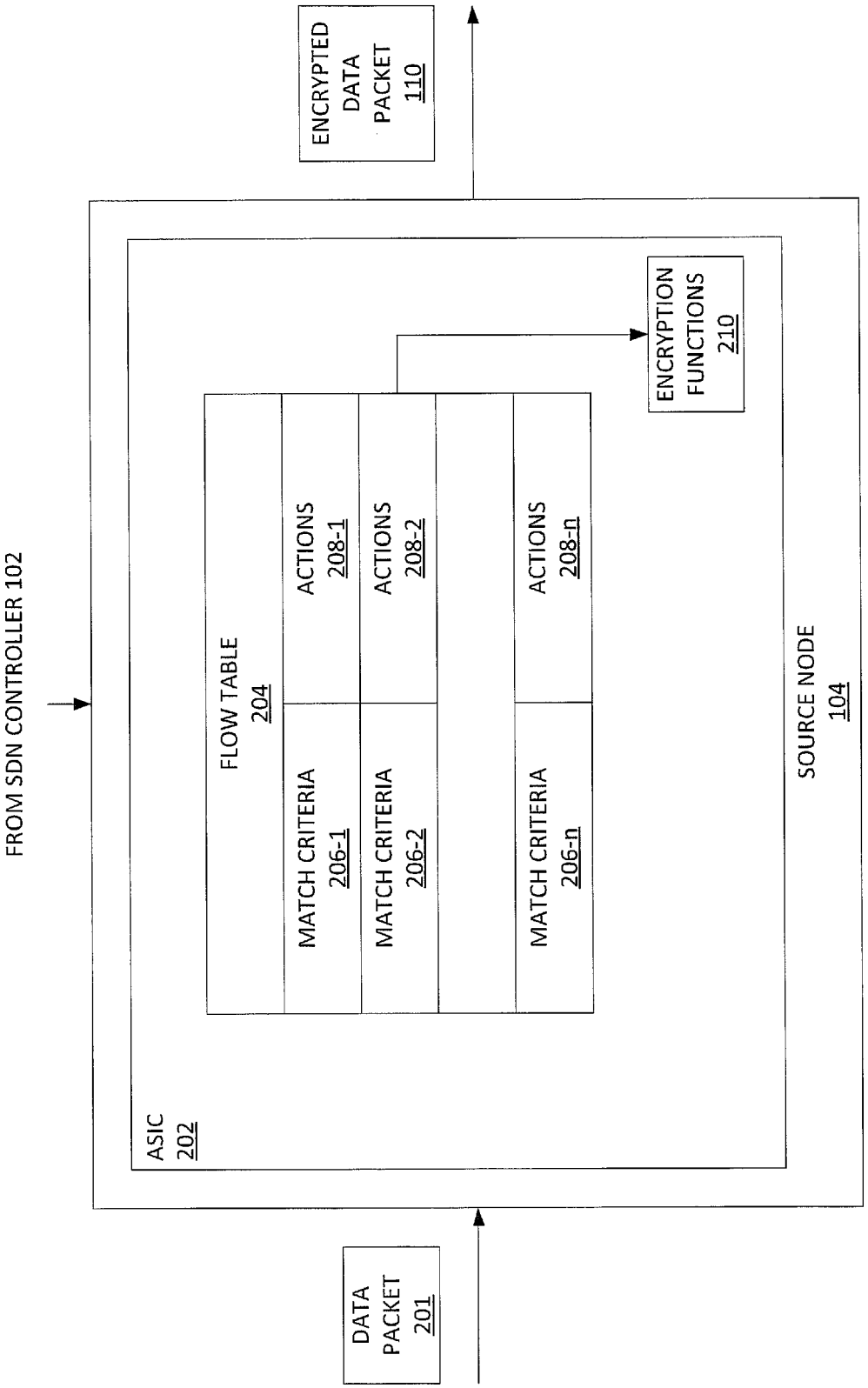

ENCRYPTED DATA PACKET 110
FROM SDN CONTROLLER 102
ENCRYPTION FUNCTIONS 210
FLOW TABLE 204
ACTIONS 208-1
ACTIONS 208-2
ACTIONS 208-n
MATCH CRITERIA 206-1
MATCH CRITERIA 206-2
MATCH CRITERIA 206-n
ASIC 202
SOURCE NODE 104
DATA PACKET 201

FIG. 2

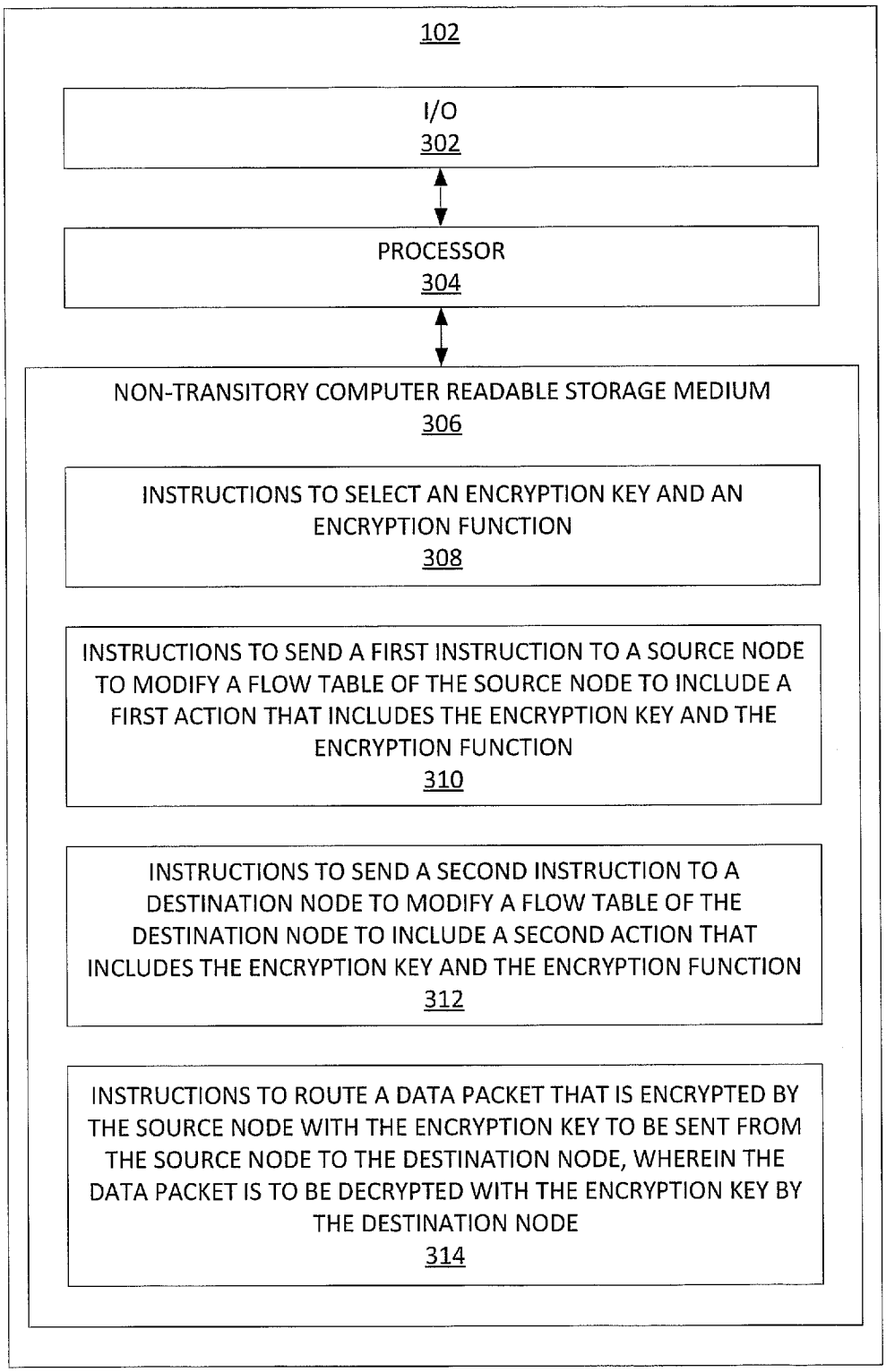

102
I/O
302
PROCESSOR
304
NON-TRANSITORY COMPUTER READABLE STORAGE MEDIUM
306
INSTRUCTIONS TO SELECT AN ENCRYPTION KEY AND AN ENCRYPTION FUNCTION
308
INSTRUCTIONS TO SEND A FIRST INSTRUCTION TO A SOURCE NODE TO MODIFY A FLOW TABLE OF THE SOURCE NODE TO INCLUDE A FIRST ACTION THAT INCLUDES THE ENCRYPTION KEY AND THE ENCRYPTION FUNCTION
310
INSTRUCTIONS TO SEND A SECOND INSTRUCTION TO A DESTINATION NODE TO MODIFY A FLOW TABLE OF THE DESTINATION NODE TO INCLUDE A SECOND ACTION THAT INCLUDES THE ENCRYPTION KEY AND THE ENCRYPTION FUNCTION
312
INSTRUCTIONS TO ROUTE A DATA PACKET THAT IS ENCRYPTED BY THE SOURCE NODE WITH THE ENCRYPTION KEY TO BE SENT FROM THE SOURCE NODE TO THE DESTINATION NODE, WHEREIN THE DATA PACKET IS TO BE DECRYPTED WITH THE ENCRYPTION KEY BY THE DESTINATION NODE
314

FIG. 3

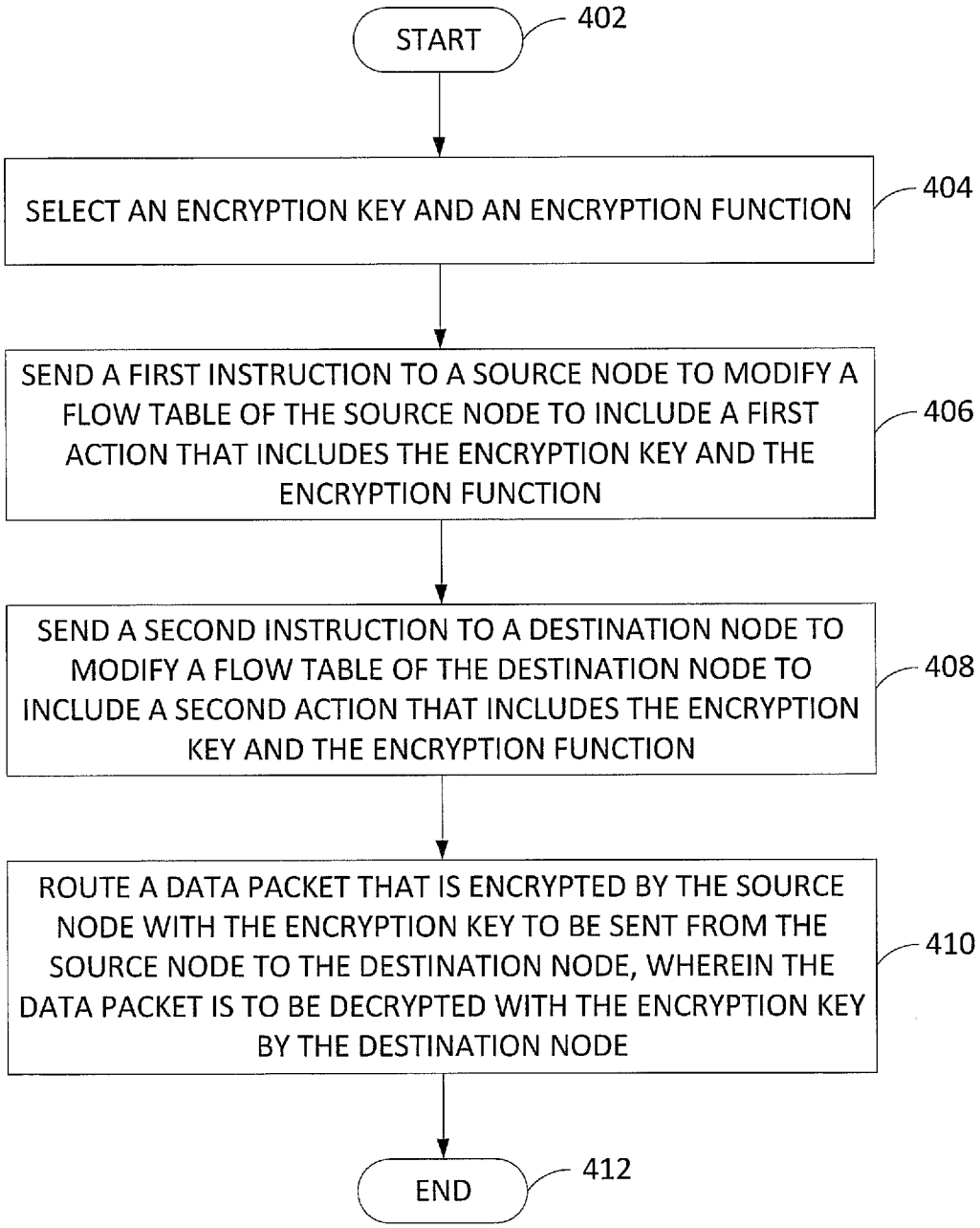

400
START
402
SELECT AN ENCRYPTION KEY AND AN ENCRYPTION FUNCTION
404
SEND A FIRST INSTRUCTION TO A SOURCE NODE TO MODIFY A FLOW TABLE OF THE SOURCE NODE TO INCLUDE A FIRST ACTION THAT INCLUDES THE ENCRYPTION KEY AND THE ENCRYPTION FUNCTION
406
SEND A SECOND INSTRUCTION TO A DESTINATION NODE TO MODIFY A FLOW TABLE OF THE DESTINATION NODE TO INCLUDE A SECOND ACTION THAT INCLUDES THE ENCRYPTION KEY AND THE ENCRYPTION FUNCTION
408
ROUTE A DATA PACKET THAT IS ENCRYPTED BY THE SOURCE NODE WITH THE ENCRYPTION KEY TO BE SENT FROM THE SOURCE NODE TO THE DESTINATION NODE, WHEREIN THE DATA PACKET IS TO BE DECRYPTED WITH THE ENCRYPTION KEY BY THE DESTINATION NODE
410
END
412

FIG. 4

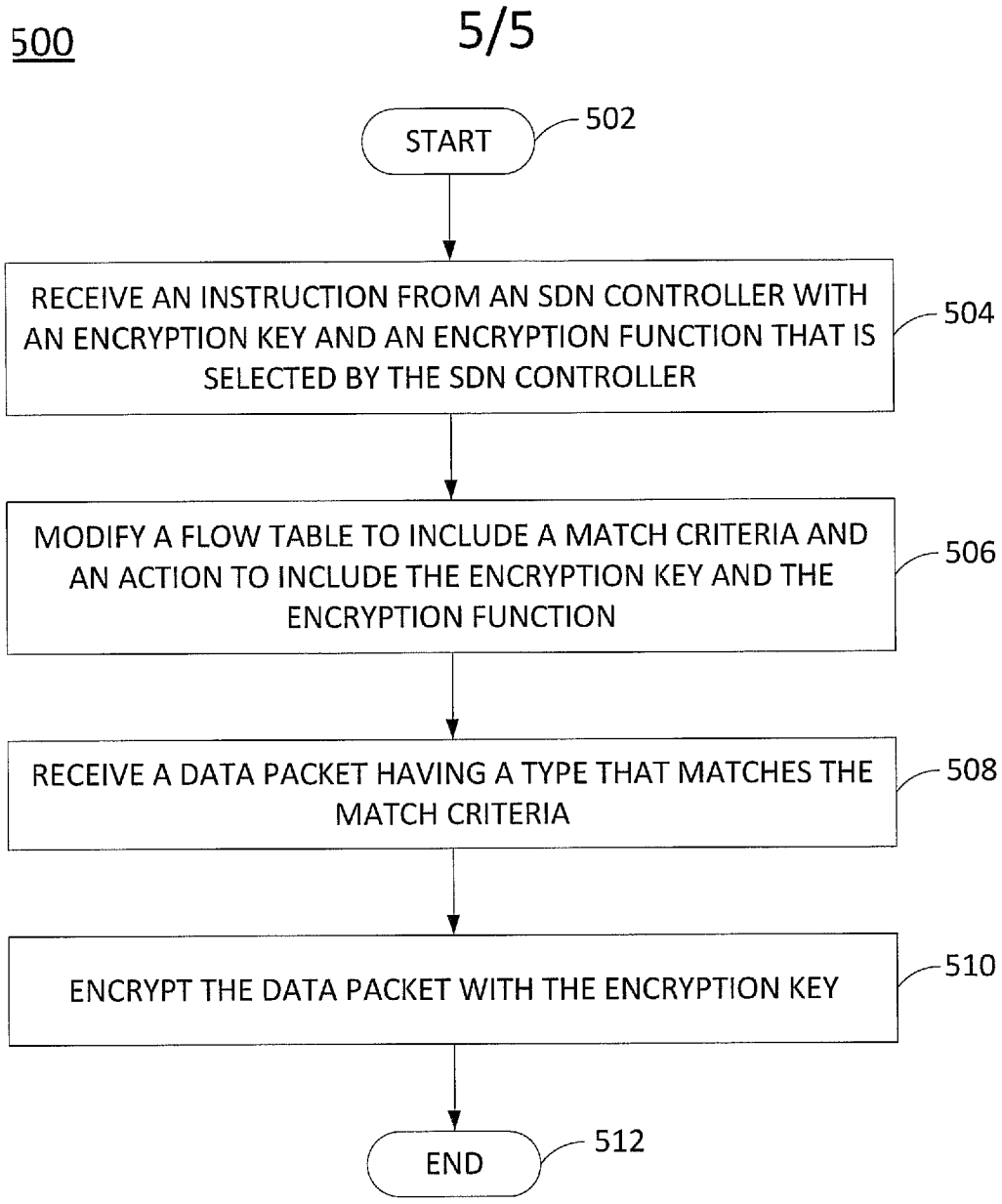

500
START
502
RECEIVE AN INSTRUCTION FROM AN SDN CONTROLLER WITH AN ENCRYPTION KEY AND AN ENCRYPTION FUNCTION THAT IS SELECTED BY THE SDN CONTROLLER
504
MODIFY A FLOW TABLE TO INCLUDE A MATCH CRITERIA AND AN ACTION TO INCLUDE THE ENCRYPTION KEY AND THE ENCRYPTION FUNCTION
506
RECEIVE A DATA PACKET HAVING A TYPE THAT MATCHES THE MATCH CRITERIA
508
ENCRYPT THE DATA PACKET WITH THE ENCRYPTION KEY
510
END
512

FIG. 5

**INTERNATIONAL SEARCH REPORT**

International application No.
**PCT/US2015/051379**

**A. CLASSIFICATION OF SUBJECT MATTER**

**H04L 29/06(2006.01)i, H04L 12/801(2013.01)i**

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 29/06; H04W 24/02; H04L 12/701; H04W 88/14; H04L 12/743; H04L 12/741; H04L 9/00; H04L 12/56; H04L 12/801

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: SDN, controller, encryption, key, function, flow, table, source, destination, node, matching, packet

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | WO 2014-131462 A1 (NOKIA SOLUTIONS AND NETWORKS OY) 04 September 2014<br>See page 8, lines 5-15; page 13, lines 6-8; page 14, lines 3-5; page 20, lines 16-20; page 25, lines 13-15; and figure 6. | 1-4,6-13,15 |
| Y | | 5,14 |
| Y | US 6862354 B1 (DAVID A. MCGREW et al.) 01 March 2005<br>See column 1, lines 15-25; column 6, lines 57-58; and figure 1A. | 5,14 |
| A | US 2015-0172189 A1 (TELEFONAKTIEBOLAGET L M ERICSSON) 18 June 2015<br>See paragraphs [0029], [0051]-[0057]; and figure 1. | 1-15 |
| A | WO 2015-131757 A1 (HANGZHOU H3C TECHNOLOGIES CO., LTD.) 11 September 2015<br>See paragraphs [0012]-[0023]; and figure 1. | 1-15 |
| A | US 2013-0223442 A1 (RAJESH NARAYANAN et al.) 29 August 2013<br>See paragraphs [0042]-[0043]; and figure 10. | 1-15 |

☐ 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 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 |
|---|---|
| 31 May 2016 (31.05.2016) | **29 July 2016 (29.07.2016)** |


Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-481-8578

Authorized officer
KIM, Seong Woo
Telephone No. +82-42-481-3348

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2015/051379**

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| WO 2014-131462 A1 | 04/09/2014 | None | |
| US 6862354 B1 | 01/03/2005 | US 7095850 B1 | 22/08/2006 |
| US 2015-0172189 A1 | 18/06/2015 | WO 2015-092565 A1 | 25/06/2015 |
| WO 2015-131757 A1 | 11/09/2015 | CN 104901825 A | 09/09/2015 |
| US 2013-0223442 A1 | 29/08/2013 | None | |