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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 October 2016 (06.10.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/155589 A1

(51) International Patent Classification:
H04L 12/761 (2013.01)

(21) International Application Number:
PCT/CN2016/077480

(22) International Filing Date:
28 March 2016 (28.03.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
201510137621.9 27 March 2015 (27.03.2015) CN

(71) Applicant: **HANGZHOU H3C TECHNOLOGIES CO., LTD.** [CN/CN]; 466 Changhe Road, Binjiang District, Hangzhou, Zhejiang 310052 (CN).

(72) Inventors: **WANG, Ju**; Room 730, Oriental Electronic BLD., NO.2, Chuangye Road, Shangdi Information Industry Base, Haidian District, Beijing 100085 (CN). **YAN, Dehan**; Room 730, Oriental Electronic BLD., NO.2, Chuangye Road, Shangdi Information Industry Base, Haidian District, Beijing 100085 (CN).

(74) Agent: **DEQI INTELLECTUAL PROPERTY LAW CORPORATION**; 7/F, Xueyuan International Tower, NO.1 Zhichun Road, Haidian District, Beijing 100083 (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, 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).

Published:

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

(54) **Title**: FORWARDING MULTICAST PACKETS

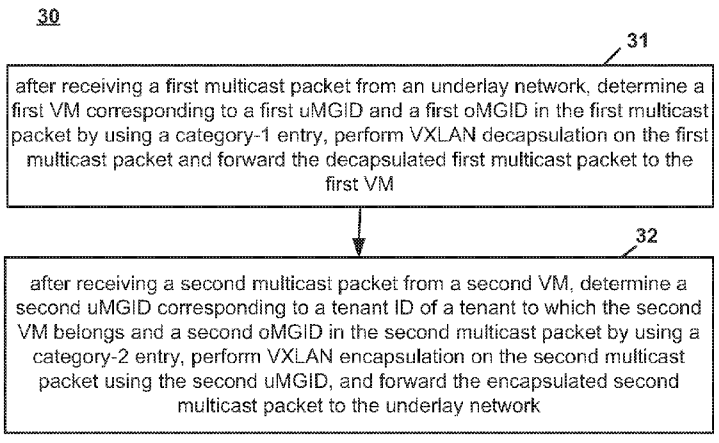


FIG. 3

(57) **Abstract**: In an example, after receiving a first multicast packet from an underlay network, a virtual switch (vSwitch) determines a first virtual machine (VM) corresponding to a first underlay network multicast group identity and a first overlay network multicast group identity in the first multicast packet by using a category-1 entry, performs VXLAN decapsulation on the first multicast packet and sends the decapsulated first multicast packet to the first VM. After receiving a second multicast packet from a second VM, the vSwitch determines a second underlay network multicast group identity corresponding to a tenant ID of a tenant to which the second VM belongs and a second overlay network multicast group identity in the second multicast packet by using a category-2 entry, performs VXLAN encapsulation on the second multicast packet using the second underlay network multicast group identity, and forwards the encapsulated second multicast packet to the underlay network.

WO 2016/155589 A1

FORWARDING MULTICAST PACKETS

Background

[0001] Virtual eXtensible Local Area Network (VXLAN) is a layer 2 VPN technique based on IP network and using "MAC in UDP" encapsulation. VXLAN enables a layer 2 virtual network to overlay a layer 3 physical network. The layer 2 virtual network is referred to as an overlay network, and the layer 3 physical network is referred to as an underlay network.

Brief Description of the Drawings

[0002] Features of the present disclosure are illustrated by way of example and not limited in the following figures.

[0003] FIG. 1 is a schematic diagram illustrating a network in accordance with examples of the present disclosure;

[0004] FIG. 2 is a schematic diagram illustrating a device in accordance with examples of the present disclosure;

[0005] Fig.3 is a flowchart illustrating a method of forwarding multicast packets in accordance with examples of the present disclosure;

[0006] FIG. 4 is a flowchart illustrating a method of a vSwitch obtaining a category-1 entry in accordance with examples of the present disclosure;

[0007] FIG. 5 is a flowchart illustrating a method of a vSwitch deleting a category-1 entry in accordance with examples of the present disclosure;

[0008] FIG. 6 is a flowchart illustrating a method of a vSwitch deleting a category-2 entry in accordance with examples of the present disclosure;

[0009] FIG. 7 is a flowchart illustrating a method of joining a multicast group in accordance with examples of the present disclosure;

[0010] FIG. 8 is a flowchart illustrating a method of leaving a multicast group in accordance with examples of the present disclosure; and

[0011] FIG. 9 is a schematic diagram illustrating an apparatus of forwarding multicast packets in accordance with examples of the present disclosure.

Detailed Descriptions

[0012] Various examples of the present disclosure may implement multicast in an overlay network by using a Software Defined Network (SDN) and an underlay network.

[0013] SDN separates the control plane of network devices from the data plane. SDN may include: an SDN controller and switching devices. According to some examples, the switching device may be a virtual switch (vSwitch).

[0014] FIG. 1 is a schematic diagram illustrating a network 10 in accordance with examples of the present disclosure. The network 10 may include an overlay network and an underlay network.

[0015] According to various examples, the overlay network may be implemented using SDN. The overlay network may include an SDN controller and plural vSwitches, such as vSwitch 1, vSwitch 2 and vSwitch n as shown in FIG. 1. In an example, the vSwitches may be configured to be VXLAN tunnel end points (VTEP). In various examples, each vSwitch may store a set of category-1 entries (C1 entries) and a set of category-2 entries (C2 entries) for forwarding multicast packets. In an example, a C1 entry or a C2 entry may comprise a tenant identity, an overlay network multicast group identity , an underlay network multicast group identity and an egress port.

[0016] FIG. 2 is a schematic diagram illustrating a device 20 in accordance with examples of the present disclosure. The device 20 may implement functions of a vSwitch. As shown in FIG. 2, the device 20 may include a processor 202, a non-transitory storage medium, e.g., a memory 204, a network communication interface 208, and an internal bus 210 connecting the components. The memory 204 may include logical instructions for controlling forwarding of multicast packets. The logical instructions may cause the processor 202 to:

[0017] after receiving a first multicast packet from an underlay network, determine a first virtual machine (VM) corresponding to a first underlay network multicast group identity (simply referred to as uMGID) and a first overlay network multicast group identity (simply referred to as oMGID) in the first multicast packet by using a category-1 entry, perform VXLAN decapsulation on the first multicast packet and forward the decapsulated first multicast packet to the first VM;

[0018] after receiving a second multicast packet from a second VM, determine a second uMGID corresponding to a tenant ID of a tenant to which the second VM belongs and a second oMGID in the second multicast packet by using a category-2 entry, perform VXLAN encapsulation on the second multicast packet using the second uMGID, and forward the encapsulated second multicast packet to the underlay network.

[0019] According to various examples, a vSwitch can forward multicast packets using a category-1 entry and a category-2 entry, thereby implement multicast in an overlay network by using an underlay network. FIG. 3 is a flowchart illustrating a method 30 of forwarding multicast packets in accordance with examples of the present disclosure. The method 30 may include the following procedures.

[0020] At block 31, a vSwitch may receive a first multicast packet from an underlay network, determine a first VM corresponding to a first uMGID and a first oMGID in the first multicast packet by using a category-1 multicast entry (simply referred to as C1 entry), perform VXLAN decapsulation on the first multicast packet and send the decapsulated first multicast packet to the first VM.

[0021] At block 32, the vSwitch may receive a second multicast packet from a second VM, determine a second uMGID corresponding to a tenant ID of a tenant to which the second VM belongs and a second oMGID in the second multicast packet by using a category-2 multicast entry (simply referred to as C2 entry), perform VXLAN encapsulation on the second multicast packet using the second uMGID, and forward the encapsulated second multicast packet to the underlay network.

[0022] According to some examples, after receiving the second multicast packet from the second VM, the vSwitch may also determine a third VM corresponding to the tenant ID of the tenant of the second VM and the second oMGID in the second multicast packet by using a C1 entry, and forward the second multicast packet to the third VM if the third VM is not the second VM. For example, the vSwitch may search for a C1 entry which includes: the tenant ID of the tenant of the second VM and the second oMGID, and an egress port which is not the port from which the second multicast packet is received.

[0023] In an example, a C1 entry may include: a tenant ID, an oMGID, an uMGID, and an egress port. The vSwitch may search for a C1 entry which includes the first

uMGID and the first oMGID, and determine the first VM using the egress port in the C1 entry found. In an example, the egress port is a port of the vSwitch which connects the vSwitch to the first VM.

[0024] According to some examples, a C2 entry may include: a tenant ID, an oMGID, an uMGID, and an egress port. The vSwitch may search for a C2 entry which includes the tenant ID of the tenant of the second VM and the second oMGID, determine the second uMGID to be the uMGID in the C2 entry found, and forward the second multicast packet which was processed through VXLAN encapsulation to the underlay network through the egress port in the C2 entry.

[0025] In an example, the vSwitch may obtain from the second multicast packet the VM ID of the second VM that sent the second multicast packet. According to some examples, static configuration in a vSwitch may include information of VMs connected to the vSwitch, e.g., a port of the vSwitch which connects the vSwitch to a VM, a VM ID of the VM, a tenant ID of a tenant to which the VM belongs, a VXLAN ID of a VXLAN to which the VM belongs, or the like. Therefore, after receiving a packet from a VM, the vSwitch may identify the VM using the port from which the packet is received, and determine the tenant ID of the tenant to which the VM belongs.

[0026] The C1 entries and the C2 entries may be obtained through different methods.

[0027] According to various examples, the vSwitch may obtain the C1 entries from the SDN controller. FIG. 4 is a flowchart illustrating a method of a vSwitch obtaining a C1 entry in accordance with examples of the present disclosure. As shown in Fig. 4, the method 40 may include the following procedures.

[0028] At block 41, a first multicast group join packet may be received from a first VM, and provided to the SDN controller which may search for a first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and a first oMGID in the first multicast group join packet and generate a C1 entry. In an example, the multicast group join packet may include a group join message conforming to a multicast protocol, e.g., an Internet Group Management Protocol (IGMP) join message.

[0029] At block 42, the C1 entry sent by the SDN controller may be received and stored.

[0030] According to an example, after receiving the first multicast group join packet from the first VM, the vSwitch may create a second multicast group join packet which may include the first uMGID, and send the second multicast group join packet through a port connected to the underlay network so as to join the underlay network multicast group.

[0031] According to various examples, the vSwitch may obtain the C2 entries from the SDN controller. The vSwitch may receive a C2 entry sent by the SDN controller. The C2 entry may be created by the SDN controller in response to a multicast group join packet sent by another vSwitch. That is, after receiving a multicast group join packet, the SDN controller may generate a relation which associates a tenant ID and an oMGID with an uMGID. The relation may be sent to the vSwitch that sent the multicast group join packet in the form of a C1 entry, and be synchronized into other vSwitches in the form of a C2 entry.

[0032] According to various examples, the vSwitch may delete a C1 entry after receiving a multicast group leave packet from a VM. FIG. 5 is a flowchart illustrating a method of a vSwitch deleting a C1 entry in accordance with examples of the present disclosure. As shown in Fig. 5, the method 50 may include the following procedures. In an example, the multicast group leave packet may include a group leave message conforming to a multicast protocol, e.g., an Internet Group Management Protocol (IGMP) leave message.

[0033] At block 51, a first multicast group leave packet may be received from a first VM, and provided to the SDN controller which may search for a first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and a first oMGID in the first multicast group leave packet and generate a C1 entry for deletion.

[0034] At block 52, the first C1 entry for deletion sent by the SDN controller may be received.

[0035] At block 53, the first C1 entry for deletion may be searched for in stored C1 entries, and the C1 entry found may be deleted.

[0036] In some examples, after deleting the C1 entry, the vSwitch may also search in remaining C1 entries for a C1 entry which includes the first uMGID in the deleted C1 entry. If no C1 entry is found, the vSwitch may create and send a second multicast

group leave packet to leave the underlay network multicast group corresponding to the first uMGID.

[0037] According to various examples, the vSwitch may delete a C2 entry according to an instruction from the SDN controller. FIG. 6 is a flowchart illustrating a method of a vSwitch deleting a C2 entry in accordance with examples of the present disclosure. As shown in Fig. 6, the method 60 may include the following procedures.

[0038] At block 61, a C2 entry for deletion sent by the SDN controller may be received. The C2 entry for deletion may be created by the SDN controller in response to a multicast group leave packet sent by another vSwitch.

[0039] At block 62, the C2 entry for deletion may be searched for in stored C2 entries, and the C2 entry found may be deleted.

[0040] That is, after receiving a multicast group leave packet, the SDN controller may generate a relation for deletion which associates a tenant ID and an oMGID with an uMGID. The relation for deletion may be sent to the vSwitch that sent the multicast group leave packet in the form of a C1 entry for deletion, and may also be synchronized into other vSwitches in the form of a C2 entry for deletion.

[0041] The process of obtaining the C1 entry and the C2 entry may be described in the following with reference to a multicast group join process.

[0042] FIG. 7 is a flowchart illustrating a method of joining a multicast group in accordance with examples of the present disclosure. As shown in FIG.7, the method 70 may include the following procedures.

[0043] At block 71, a VM in an overlay network, e.g., VM 11 in FIG. 1, may send a multicast group join packet which includes an identity of an overlay network multicast group which the VM wants to join, and the identity may be denoted as G1.

[0044] At block 72, a vSwitch, e.g., vSwitch 1 as shown in FIG. 1, may receive the multicast group join packet sent by VM 11, and send the multicast group join packet to the SDN controller.

[0045] At block 73, the SDN controller may receive the multicast group join packet sent by vSwitch 1, search for a relation which matches a tenant identity of a tenant to which VM 11 belongs and the oMGID in the multicast group join packet, perform the

procedure in block 74 if the relation is not found, and perform the procedure in block 75 if the relation is found.

[0046] The SDN controller may manage VMs connected to each vSwitch, e.g., allocate a tenant ID, a VXLAN identity (VNI) or the like for each VM connected to each vSwitch. Therefore, in block 73, after receiving the multicast group join packet sent by vSwitch 1, the SDN controller may determine the tenant ID (e.g., virtual private cloud identity (VPCID) 1) of the tenant to which VM 11 belongs according to the VM ID in the multicast group join packet. Then the SDN controller may search in stored relations for a relation using the VPCID1 and G1 as keywords, perform the procedure in block 74 if the relation is not found, and perform the procedure in block 75 if the relation is found.

[0047] At block 74, the SDN controller may create a relation which includes the tenant ID and the oMGID.

[0048] In an example, the relation created in block 74 may include: VPCID1, G1, an uMGID (denoted as G1') corresponding to G1, and a counting value. The counting value may be set to be a pre-determined value (e.g., the pre-determined value may be 1 according to an example).

[0049] Table 1 shows an example of the relation created in block 74.

tenant ID	oMGID	uMGID	counting value
			pre-determined value

Table 1

[0050] According to various examples, the tables illustrated herein are merely examples, and may include other information according to practical needs.

[0051] In various examples, there may be various applications running on the SDN controller. One of the applications, which is generally referred to as broadcast/multicast (BM) application, may implement centralized management of multicast groups in the overlay network and multicast groups in the underlay network. In the procedure of block 74, the SDN controller may allocate an uMGID to an oMGID in the multicast group join packet using the BM application.

[0052] The SDN controller may select an unoccupied uMGID from pre-configured uMGIDs and allocate the selected uMGID to the oMGID.

[0053] For example, oMGIDs pre-configured in the SDN controller may range from 1 to 100, and uMGIDs pre-configured in the SDN controller may range from 101 to 200. Supposing the oMGID in the multicast group join packet is 10, an unoccupied uMGID may be selected from the range 101 to 200, e.g., 110, and allocated to oMGID 10.

[0054] According an example, the SDN controller may abide by the following rules when allocating the uMGID to the oMGID.

[0055] Rule 1: For multicast group join packets sent by VMs of different tenants, different uMGIDs may be allocated as long as the packets are sent by VMs from different tenants even if the packets include the same oMGID.

[0056] For example, the SDN controller may receive a multicast group join packet sent by VM 11 of tenant 1, and the packet includes multicast group 1 as an oMGID. The SDN controller may allocate multicast group 101 to the multicast group 1 of tenant 1. After that, the SDN controller may receive a multicast group join packet sent by VM 12 of tenant 2, and the packet includes multicast group 1 as an oMGID. The SDN controller may allocate a different uMGID to multicast group 1 in the multicast group join packet sent by VM 12, and may allocate multicast group 102 or the like to multicast group 1 of tenant 2. As such, it can be ensured that different uMGIDs are allocated for multicast group join packets sent by VMs from different tenants even if the multicast group join packets include the same oMGID.

[0057] Rule 2: The same uMGID may be allocated for multicast group join packets which includes the same oMGID sent by different VMs of the same tenant even if the VMs have different VNI.

[0058] For example, the SDN controller may receive a multicast group join packet sent by VM 1 of tenant 1, and the packet includes an oMGID of multicast group 1. Supposing the VNI of VM 1 is VNI 1, the SDN controller may allocate multicast group 101 as the uMGID for the multicast group 1. After that, SDN controller may receive a multicast group join packet sent by VM 21 of tenant 1, the packet includes an oMGID of multicast group 1, and the VNI of VM 21 is VNI 2. The SDN controller may allocate multicast group 101 as the uMGID allocated to multicast group 1 for the multicast group join packet sent by VM 21.

[0059] According to the above two rules, the SDN controller may allocate an uMGID to an oMGID in a multicast group join packet.

[0060] At block 75, the counting value in the found relation is increased by 1.

[0061] The SDN controller may make use of stored relations instead of creating a new relation which is the same with a stored relation.

[0062] At block 76, subsequent to block 74 or 75, the SDN controller may create a C1 entry using the VPCID 1, G1 and G1' in the relation and the virtual port that connects the vSwitch 1 to VM 11 which serves as an egress port in the C1 entry, and send the C1 entry to the vSwitch 1. For each vSwitch other than the vSwitch 1, the SDN controller may create a C2 entry using the VPCID 1, G1, G1' and a port connecting the vSwitch to the underlay network which serves as an egress port in the C2 entry, and send the C2 entry to the vSwitch.

[0063] The SDN controller may send the C1 entry to the vSwitch 1 and a C2 entry to each vSwitch other than vSwitch 1 as flow table entries.

[0064] At block 77, subsequent to block 76, vSwitch 1 may receive and store the C1 entry sent by the SDN controller.

[0065] The C1 entry may include the information listed in block 76, e.g., an egress port, a tenant ID, an oMGID, and an uMGID. Table 2 is an example of a C1 entry.

egress port	tenant ID	oMGID	uMGID

Table 2

[0066] At block 78, subsequent to block 76, a vSwitch other than vSwitch 1 may receive and store a C2 entry sent by the SDN controller, and set a counting value corresponding to the C2 entry. The counting value may be set to be a pre-determined value.

[0067] The C2 entry may include the information listed in block 76, e.g., a tenant ID, an oMGID, an uMGID and an egress port. Table 3 illustrates an example of a C2 entry.

tenant ID	oMGID	uMGID	egress port



Table 3

[0068] In an example, in order to avoid creating an entry which is the same with a stored entry, before storing the C2 entry in block 78, the method may also include:

[0069] searching for a stored C2 entry which is the same with the received C2 entry;

[0070] increasing the counting value corresponding to the found C2 entry by 1 if the C2 entry is found; and

[0071] storing the received C2 entry and setting a counting value corresponding to the C2 entry if the C2 entry is not found. The initial value of the counting value may be a pre-determined value.

[0072] The above are examples of how a vSwitch in an SDN may obtain a C1 entry and a C2 entry. After receiving multicast packets, each vSwitch may forward the multicast packets using respective C1 entries and C2 entries (block 79). The forwarding process may be as shown in FIG. 4, and not repeated herein.

[0073] The process of deleting C1 entries and C2 entries may be described in the following with reference to a multicast group leave process.

[0074] FIG. 8 is a flowchart illustrating a method of leaving a multicast group in accordance with examples of the present disclosure. As shown in FIG.8, the method 80 may include the following procedures.

[0075] At block 81, a VM in an overlay network, e.g., VM 21 in FIG. 1, may send a multicast group leave packet which includes an identity of an overlay network multicast group which the VM wants to leave, and the identity may be denoted as G2.

[0076] At block 82, vSwitch 2 connected with VM 21 may receive the multicast group leave packet and send the multicast group leave packet to the SDN controller.

[0077] At block 83, the SDN controller may receive the multicast group leave packet sent by vSwitch 2, search in stored relations for a relation which includes: a tenant ID of the tenant of VM 21 (e.g., VPCID 2), G2. The relation may also include an uMGID (denoted by G2') corresponding to G2.

[0078] At block 84, the SDN controller may create a C1 entry for deletion using VPCID 2, G2, G2' and a vport connecting vSwitch 2 to VM 21 which serves as an egress port, and send the C1 entry for deletion to vSwitch 2. For each vSwitch other than vSwitch 2, the SDN controller may create a C2 entry for deletion using VPCID 2, G2, G2' and a port connecting the vSwitch to the underlay network which serves as an egress port, and send the C2 entry for deletion to the vSwitch. The SDN controller may decrease a counting value in the relation by 1, judge whether the decreased counting value equals the pre-determined value, and delete the relation if the decreased counting value equals the pre-determined value.

[0079] According to an example, before deleting the relation at block 84, the SDN controller may mark the uMGID in the relation as unoccupied, i.e., making the uMGID available for other oMGID.

[0080] At block 85, subsequent to block 84, vSwitch 2 may receive the C1 entry for deletion sent by the SDN controller, search for a C1 entry identical to the C1 entry for deletion and delete the C1 entry found. After that, vSwitch 2 may check whether vSwitch 2 stores a C1 entry which includes: the uMGID in the C1 entry for deletion, terminate the process if the C1 entry is found, or create a multicast group leave packet which includes the uMGID in the C1 entry for deletion to leave the underlay network multicast group if vSwitch 2 does not store a C1 entry which includes the uMGID.

[0081] At block 86, subsequent to block 84, a vSwitch other than vSwitch 2 may receive the C2 entry for deletion sent by the SDN controller, search for a C2 entry identical to the C2 entry for deletion. If a C2 entry identical to the C2 entry for deletion is found, the vSwitch may decrease a counting value corresponding to the found C2 entry by 1, and check whether the counting value equals the pre-determined value after 1 is deducted from the counting value. If the counting value equals the pre-determined value, the vSwitch may delete the C2 entry found. If the counting value does not equal the pre-determined value, the vSwitch may terminate the process.

[0082] Hence, the process as shown in Fig. 8 is completed.

[0083] FIG. 9 is a schematic diagram illustrating an apparatus 90 of forwarding multicast packets in accordance with examples of the present disclosure. As shown in

FIG. 9, the apparatus 90 may include: a sending unit 91, a C1 entry unit 92, a C2 entry unit 93, and a packet unit 94.

[0084] The sending unit 91 may send a multicast group join packet from a VM to an SDN controller.

[0085] The C1 entry unit 92 may receive and store a C1 entry sent by the SDN controller in response to the multicast group join packet. The C1 entry may include: an egress port, a VNI of a VXLAN that the egress port belongs to, a tenant ID, an oMGID, and an uMGID. The C1 entry unit 92 may create a multicast group join packet corresponding to the uMGID to join a multicast group corresponding to the uMGID.

[0086] The C2 entry unit 93 may receive and store a C2 entry sent by the SDN controller in response to a multicast group join packet sent by another vSwitch, and set a counting value corresponding to the C2 entry. The counting value is set to be a pre-determined value. The C2 entry may include a tenant ID, an oMGID, an uMGID, and an egress port.

[0087] The packet unit 94 may receive a multicast packet from a VM, search in stored C1 entries and C2 entries for an entry for forwarding the multicast packet, and forward the multicast packet using an entry found. The packet unit 94 may also receive a multicast packet from the underlay network, search in the stored C1 entries for an entry for forwarding the multicast packet, and forward the multicast packet using an entry found.

[0088] In an example, before storing the C2 entry and setting the counting value corresponding to the C2 entry, the C2 entry unit 93 may search stored C2 entries for an entry identical to the received C2 entry, increase a counting value corresponding to the found entry by 1 if the entry identical to the received C2 entry is found, or perform the procedure of storing the C2 entry and setting the counting value corresponding to the C2 entry if the entry identical to the received C2 entry is not found.

[0089] In an example, the sending unit 91 may send a multicast group leave packet received from a VM to the SDN controller.

[0090] The C1 entry 92 may also receive a C1 entry for deletion sent by the SDN controller in response to a multicast group leave packet sent by the vSwitch, search in stored C1 entries for a C1 entry identical to the received C1 entry for deletion, and delete

the C1 entry found. The C1 entry 92 may also check whether remaining C1 entries include a C1 entry which includes: the uMGID in the received C1 entry for deletion.

[0091] If the stored C1 entries include a C1 entry including the uMGID, the C1 entry unit 92 may terminate the process. If the remaining C1 entries do not include a C1 entry which includes the uMGID, the C1 entry unit 92 may create a multicast group leave packet which includes the uMGID in the C1 entry for deletion to leave the multicast group in the underlay network corresponding to the uMGID in the C1 entry for deletion.

[0092] According to examples, the C2 entry unit 93 may receive a C2 entry for deletion sent by the SDN controller in response to a multicast group leave packet sent by another vSwitch. The C2 entry unit 93 may search in stored C2 entries for a C2 entry identical to the received C2 entry for deletion. The C2 entry unit 93 may decrease a counting value corresponding to the found C2 entry by 1, and check whether the counting value equals the pre-determined value after 1 is deducted from the counting value. If the counting value equals the pre-determined value, the C2 entry unit 93 may delete the C2 entry found. If the counting value does not equal the pre-determined value, the C2 entry unit 93 may terminate the process.

[0093] According to some examples, the packet unit 94 may search in stored C1 entries and C2 entries for an entry for forwarding a multicast packet, and forward the multicast packet using the found entry. The process of forwarding the multicast packet may include the following procedures.

[0094] The packet unit 94 may search in the stored C1 entries for an entry which includes: a tenant ID corresponding to the multicast packet, an oMGID in the multicast packet. If the C1 entry is found and an egress port in the C1 entry is not the port through which the multicast packet is received, the packet unit 94 may forward the multicast packet according to a VNI and the egress port in the C1 entry.

[0095] The packet unit 94 may search in stored C2 entries for a C2 entry which includes: a tenant ID corresponding to the multicast packet, an oMGID in the multicast packet. If the C2 entry is found, the packet unit 94 may forward the multicast packet using an uMGID and an egress in the C2 entry.

[0096] According to some examples, the packet unit 94 may search in the stored C1 entries for an entry which includes: an uMGID, and forward the multicast packet using an oMGID, a VNI and an egress port in the found C1 entry if the C1 entry is found.

[0097] It should be understood that in the above processes and structures, not all of the procedures and modules are necessary. Certain procedures or modules may not be present in certain implementations according to the needs of the user. The order of the procedures is not fixed, and can be adjusted. The modules are defined based on function simply for facilitating description. In implementation, a module may be implemented by multiple modules, and functions of multiple modules may be implemented by the same module. The modules may reside in the same device or distribute in different devices. The "first", "second" in the above descriptions are merely for distinguishing two similar objects, and have no substantial meanings.

[0098] The scope of the claims should not be limited by the embodiments set forth in the examples, but should be given the broadest interpretation consistent with the description as a whole.

Claims

1. A method of forwarding a multicast packet, comprising:

after receiving a first multicast packet from an underlay network, determining, by a virtual switch (vSwitch), a first virtual machine (VM) corresponding to a first underlay network multicast group identity (uMGID) and a first overlay network multicast group identity (oMGID) in the first multicast packet by using a category-1 entry (C1 entry), performing VXLAN decapsulation on the first multicast packet and sending the decapsulated first multicast packet to the first VM;

after receiving a second multicast packet from a second VM, determining a second uMGID corresponding to a tenant ID of a tenant to which the second VM belongs and a second oMGID in the second multicast packet by using a category-2 entry (C2 entry), performing VXLAN encapsulation on the second multicast packet using the second uMGID, and forwarding the encapsulated second multicast packet to the underlay network.

2. The method of claim 1, wherein the C1 entry comprises: a tenant ID, an oMGID, an uMGID and an egress port;

wherein determining the first VM by looking up the C1 entry comprises: searching for a C1 entry which matches the first uMGID and the first oMGID, and determining the first VM using the egress port in the C1 entry.

3. The method of claim 1, wherein the C2 entry comprises: a tenant ID, an oMGID, an uMGID and an egress port;

wherein determining the second uMGID by using the C2 entry comprises: searching for a C2 entry which matches the tenant ID of the tenant to which the second VM belongs and the second oMGID, determining the uMGID in the C2 entry as the second uMGID, performing VXLAN encapsulation on the second multicast packet, and forwarding the encapsulated second multicast packet to the underlay networking via the egress port in the C2 entry.

4. The method of claim 2, further comprising: obtaining the C1 entry; wherein obtaining the C1 entry comprises:

receiving a first multicast group join packet from the first VM, and providing the first multicast group join packet to an SDN controller which searches for the first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and the first oMGID in the first multicast group join packet and generates a C1 entry; and

receiving and storing the C1 entry sent by the SDN controller.

5. The method of claim 3, further comprising: obtaining the C2 entry; wherein obtaining the C2 entry comprises:

receiving the C2 entry sent by the SDN controller;

wherein the C2 entry is created by the SDN controller in response to a multicast group join packet sent by another vSwitch.

6. The method of claim 2, further comprising: deleting the C1 entry; wherein deleting the C1 entry comprises:

receiving a first multicast group leave packet from the first VM, and providing the first multicast group leave packet to an SDN controller which searches for the first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and the first oMGID in the first multicast group leave packet and generates a C1 entry for deletion; and

receiving the C1 entry for deletion sent by the SDN controller; and

searching in stored C1 entries for a C1 entry identical to the C1 entry for deletion, and deleting the C1 entry found.

7. The method of claim 3, further comprising: deleting the C2 entry; wherein deleting the C2 entry comprises:

receiving a C2 entry for deletion sent by the SDN controller;

searching in stored C2 entries for a C2 entry identical to the C2 entry for deletion, and deleting the C2 entry found;

wherein the C2 entry for deletion is created by the SDN controller in response to a multicast group leave packet sent by another vSwitch.

8. A device, comprising a processor and a non-transitory storage medium which stores logic instructions for forwarding a multicast packet, wherein the logic instructions are executable by the processor to:

after receiving a first multicast packet from an underlay network, determine a first virtual machine (VM) corresponding to a first underlay network multicast group identity (uMGID) and a first overlay network multicast group identity (oMGID) in the first multicast packet by using a category-1 entry (C1 entry), perform VXLAN decapsulation on the first multicast packet and send the decapsulated first multicast packet to the first VM;

after receiving a second multicast packet from a second VM, determine a second uMGID corresponding to a tenant ID of a tenant to which the second VM belongs and a second oMGID in the second multicast packet by using a category-2 entry (C2 entry), perform VXLAN encapsulation on the second multicast packet using the second uMGID, and forward the encapsulated second multicast packet to the underlay network.

9. The device of claim 8, wherein

the C1 entry comprises: a tenant ID, an oMGID, an uMGID, and an egress port;

the logic instructions are executable by the processor to:

search for a C1 entry which matches the first uMGID and the first oMGID, and determine the first VM using the egress port in the C1 entry found.

10. The device of claim 8, wherein

the C2 entry comprises: a tenant ID, an oMGID, an uMGID, and an egress port;

the logic instructions are executable by the processor to:

search for a C2 entry which matches the tenant ID of the tenant of the second VM and the second oMGID, determine the second uMGID to be the uMGID in the C2 entry found, and forward the second multicast packet which was processed through VXLAN encapsulation to the underlay network through the egress port in the C2 entry.

11. The device of claim 9, wherein the logic instructions are executable by the processor to obtain the C1 entry by:

receiving a first multicast group join packet from the first VM, and providing the first multicast group join packet to an SDN controller which searches for the first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and the first oMGID in the first multicast group join packet and generates a C1 entry; and

receiving and storing the C1 entry sent by the SDN controller.

12. The device of claim 10, wherein the logic instructions are executable by the processor to obtain the C2 entry by:

receiving the C2 entry sent by the SDN controller;

wherein the C2 entry is created by the SDN controller in response to a multicast group join packet sent by a second vSwitch.

13. The device of claim 9, wherein the logic instructions are executable by the processor to delete the C1 entry by:

receiving a first multicast group leave packet from the first VM, and providing the first multicast group leave packet to an SDN controller which searches for the first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and the first oMGID in the first multicast group leave packet and generates a C1 entry for deletion;

receiving the first C1 entry for deletion sent by the SDN controller; and

searching in stored C1 entries for a C1 entry identical to the first C1 entry for deletion, and deleting the C1 entry found.

14. The device of claim 10, wherein the logic instructions are executable by the processor to delete the C2 entry by:

receiving a C2 entry for deletion sent by the SDN controller;

wherein the C2 entry for deletion is created by the SDN controller in response to a multicast group leave packet sent by a third vSwitch.

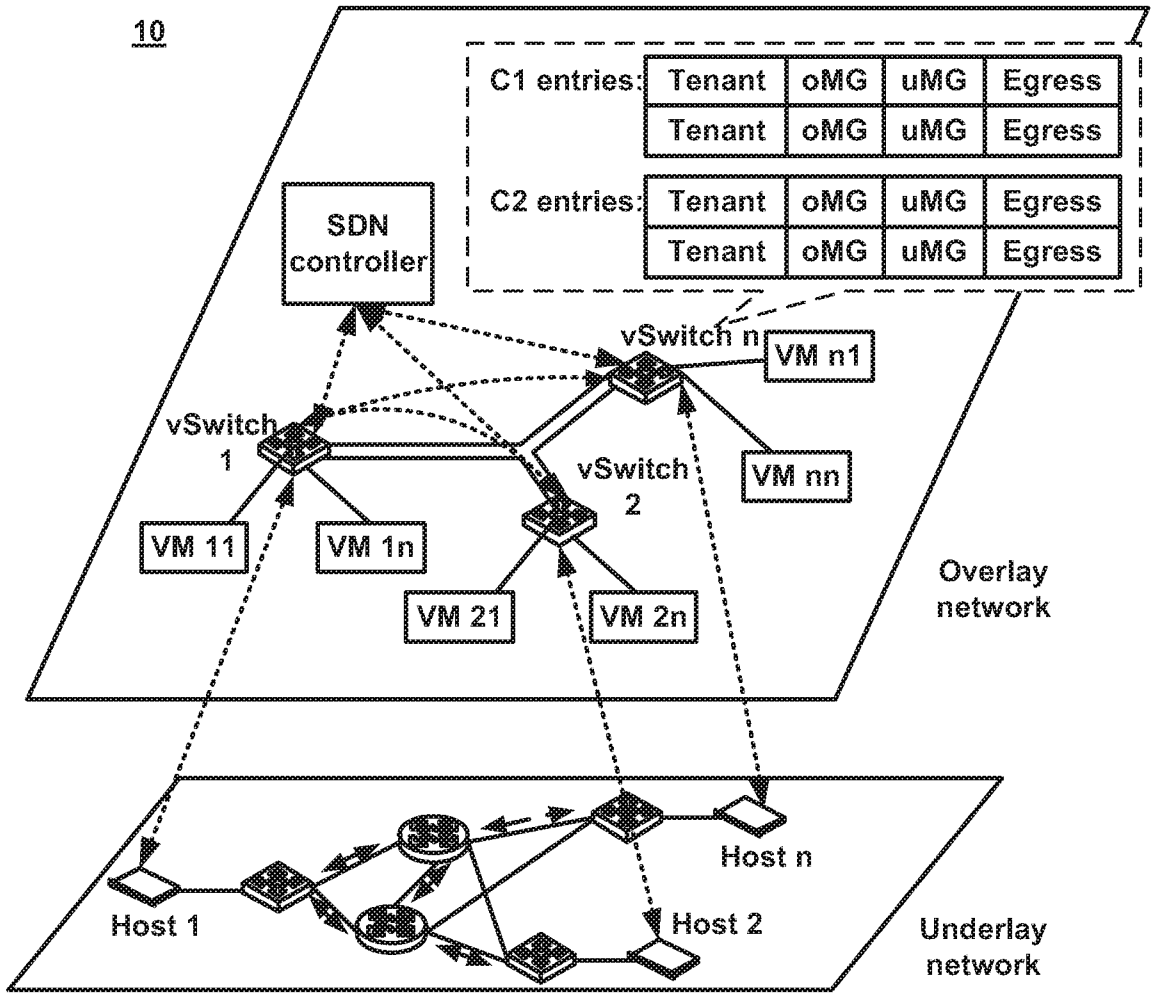

10
C1 entries:
Tenant
oMG
uMG
Egress
Tenant
oMG
uMG
Egress
C2 entries:
Tenant
oMG
uMG
Egress
Tenant
oMG
uMG
Egress
SDN controller
vSwitch n
VM n1
vSwitch 1
vSwitch 2
VM nn
VM 11
VM 1n
VM 21
VM 2n
Overlay network
Host n
Host 1
Host 2
Underlay network

FIG. 1

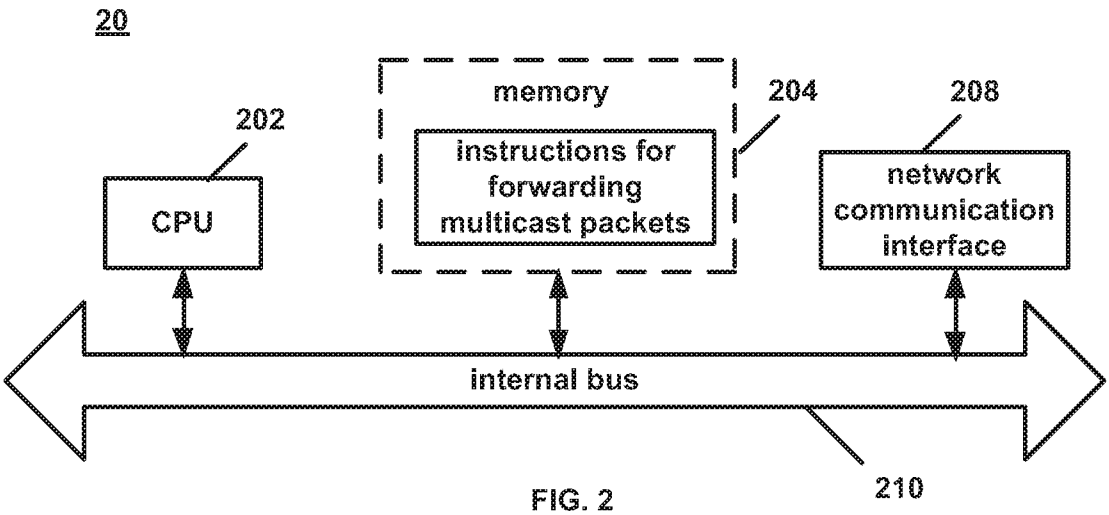

20
memory
204
208
202
instructions for forwarding multicast packets
network communication interface
CPU
internal bus
210

FIG. 2

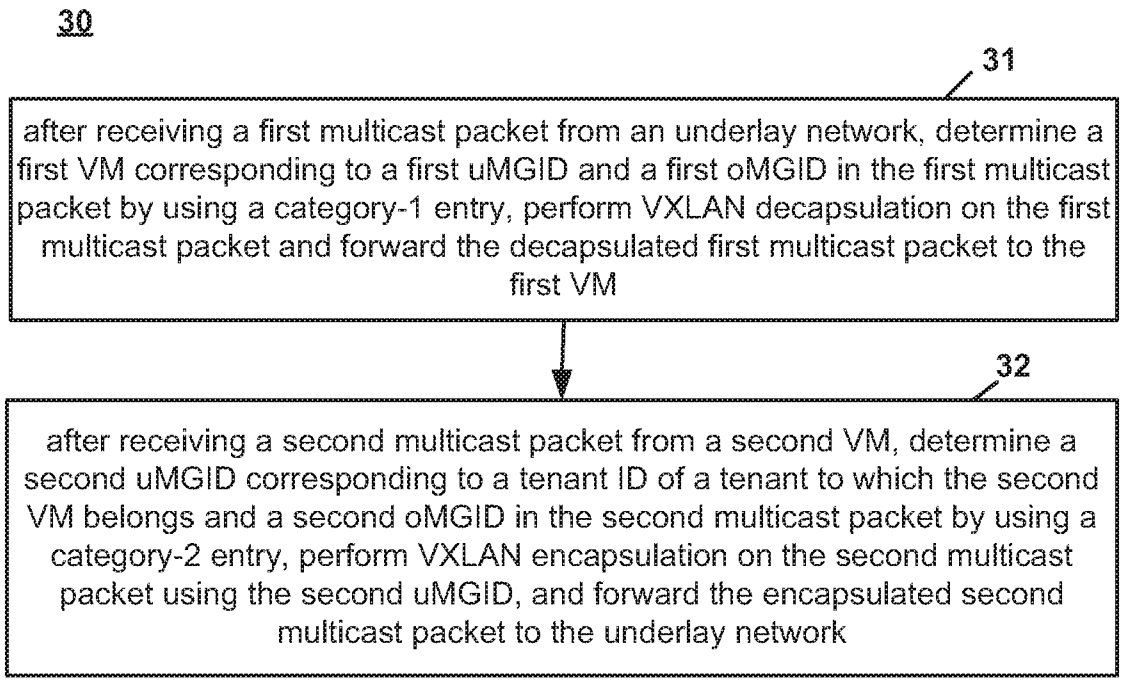

30
31
after receiving a first multicast packet from an underlay network, determine a first VM corresponding to a first uMGID and a first oMGID in the first multicast packet by using a category-1 entry, perform VXLAN decapsulation on the first multicast packet and forward the decapsulated first multicast packet to the first VM
32
after receiving a second multicast packet from a second VM, determine a second uMGID corresponding to a tenant ID of a tenant to which the second VM belongs and a second oMGID in the second multicast packet by using a category-2 entry, perform VXLAN encapsulation on the second multicast packet using the second uMGID, and forward the encapsulated second multicast packet to the underlay network

FIG. 3

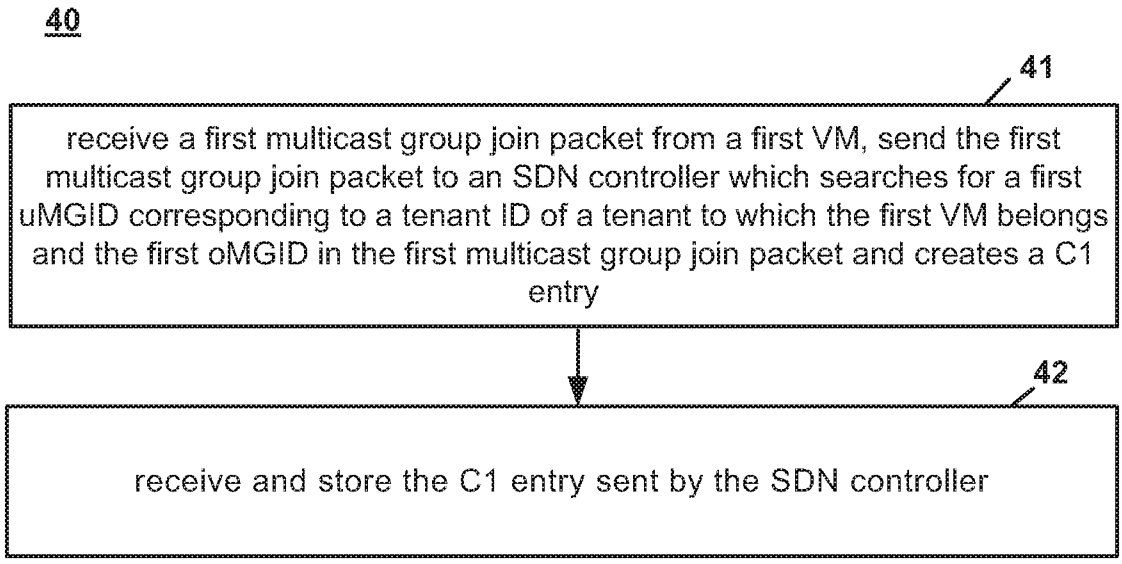

40
41
receive a first multicast group join packet from a first VM, send the first multicast group join packet to an SDN controller which searches for a first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and the first oMGID in the first multicast group join packet and creates a C1 entry
42
receive and store the C1 entry sent by the SDN controller

FIG. 4

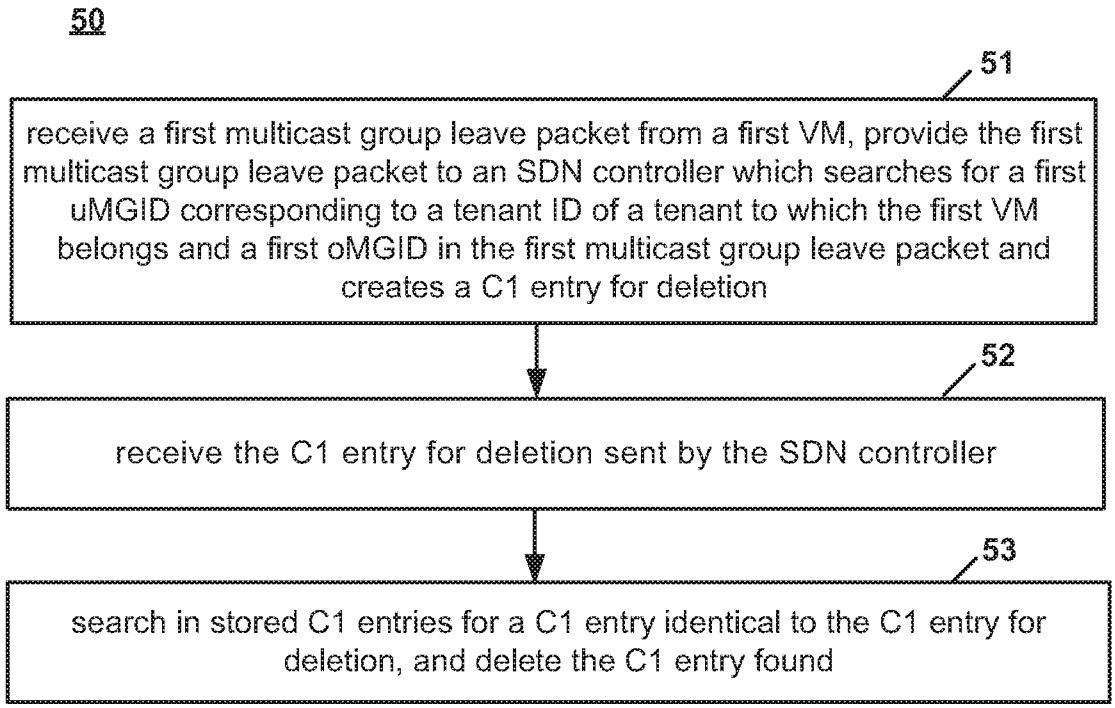

50
51
receive a first multicast group leave packet from a first VM, provide the first multicast group leave packet to an SDN controller which searches for a first uMGID corresponding to a tenant ID of a tenant to which the first VM belongs and a first oMGID in the first multicast group leave packet and creates a C1 entry for deletion
52
receive the C1 entry for deletion sent by the SDN controller
53
search in stored C1 entries for a C1 entry identical to the C1 entry for deletion, and delete the C1 entry found

FIG. 5

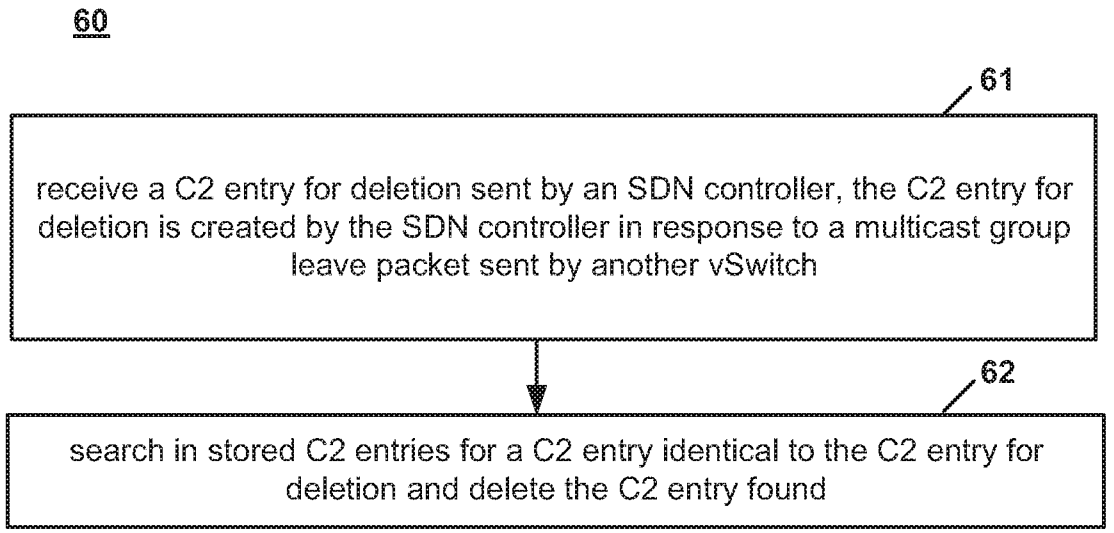

60
61
receive a C2 entry for deletion sent by an SDN controller, the C2 entry for deletion is created by the SDN controller in response to a multicast group leave packet sent by another vSwitch
62
search in stored C2 entries for a C2 entry identical to the C2 entry for deletion and delete the C2 entry found

FIG. 6

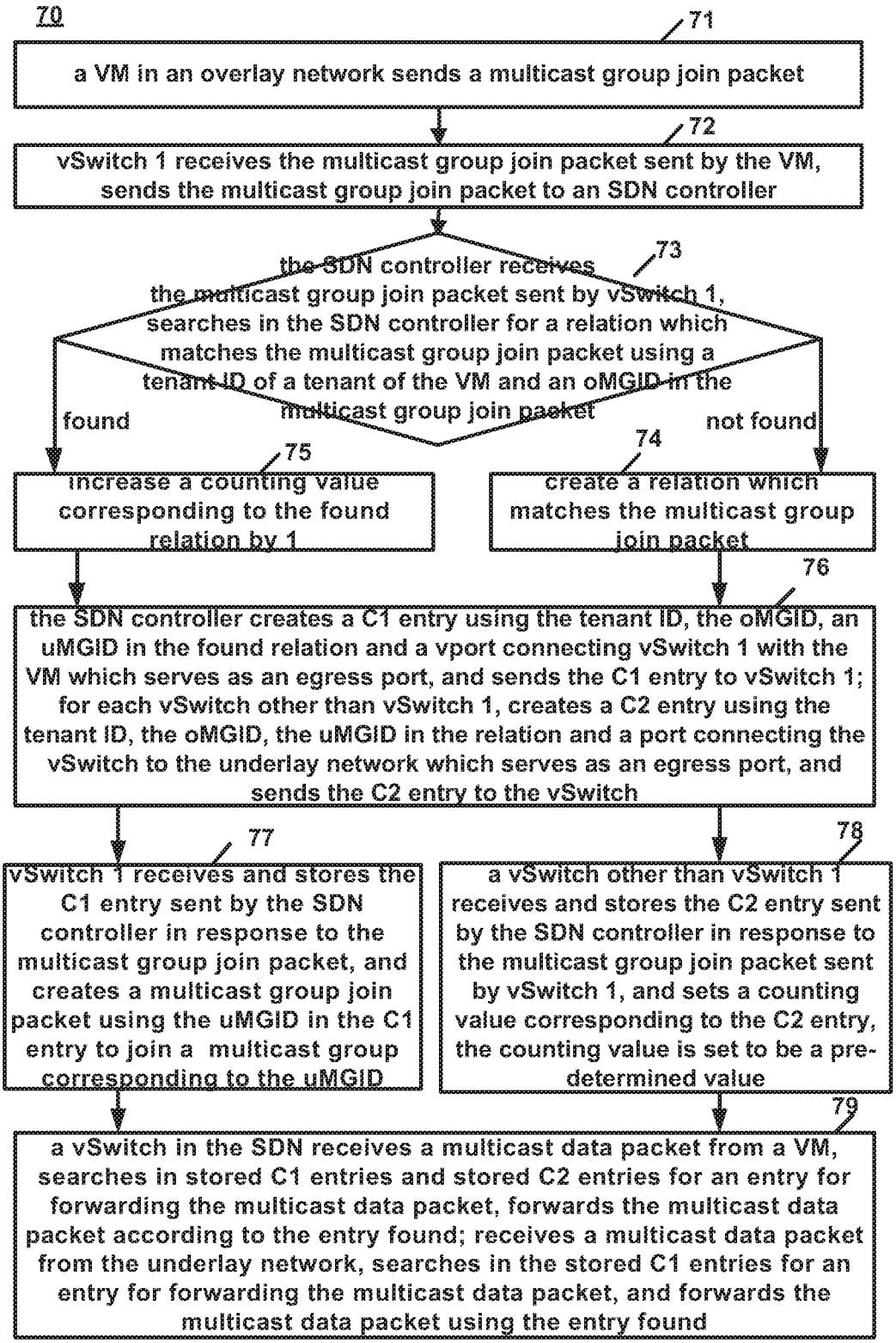

70
71
a VM in an overlay network sends a multicast group join packet
72
vSwitch 1 receives the multicast group join packet sent by the VM, sends the multicast group join packet to an SDN controller
73
the SDN controller receives the multicast group join packet sent by vSwitch 1, searches in the SDN controller for a relation which matches the multicast group join packet using a tenant ID of a tenant of the VM and an oMGID in the multicast group join packet
found
not found
75
increase a counting value corresponding to the found relation by 1
74
create a relation which matches the multicast group join packet
76
the SDN controller creates a C1 entry using the tenant ID, the oMGID, an uMGID in the found relation and a vport connecting vSwitch 1 with the VM which serves as an egress port, and sends the C1 entry to vSwitch 1; for each vSwitch other than vSwitch 1, creates a C2 entry using the tenant ID, the oMGID, the uMGID in the relation and a port connecting the vSwitch to the underlay network which serves as an egress port, and sends the C2 entry to the vSwitch
77
vSwitch 1 receives and stores the C1 entry sent by the SDN controller in response to the multicast group join packet, and creates a multicast group join packet using the uMGID in the C1 entry to join a multicast group corresponding to the uMGID
78
a vSwitch other than vSwitch 1 receives and stores the C2 entry sent by the SDN controller in response to the multicast group join packet sent by vSwitch 1, and sets a counting value corresponding to the C2 entry, the counting value is set to be a pre-determined value
79
a vSwitch in the SDN receives a multicast data packet from a VM, searches in stored C1 entries and stored C2 entries for an entry for forwarding the multicast data packet, forwards the multicast data packet according to the entry found; receives a multicast data packet from the underlay network, searches in the stored C1 entries for an entry for forwarding the multicast data packet, and forwards the multicast data packet using the entry found

FIG. 7

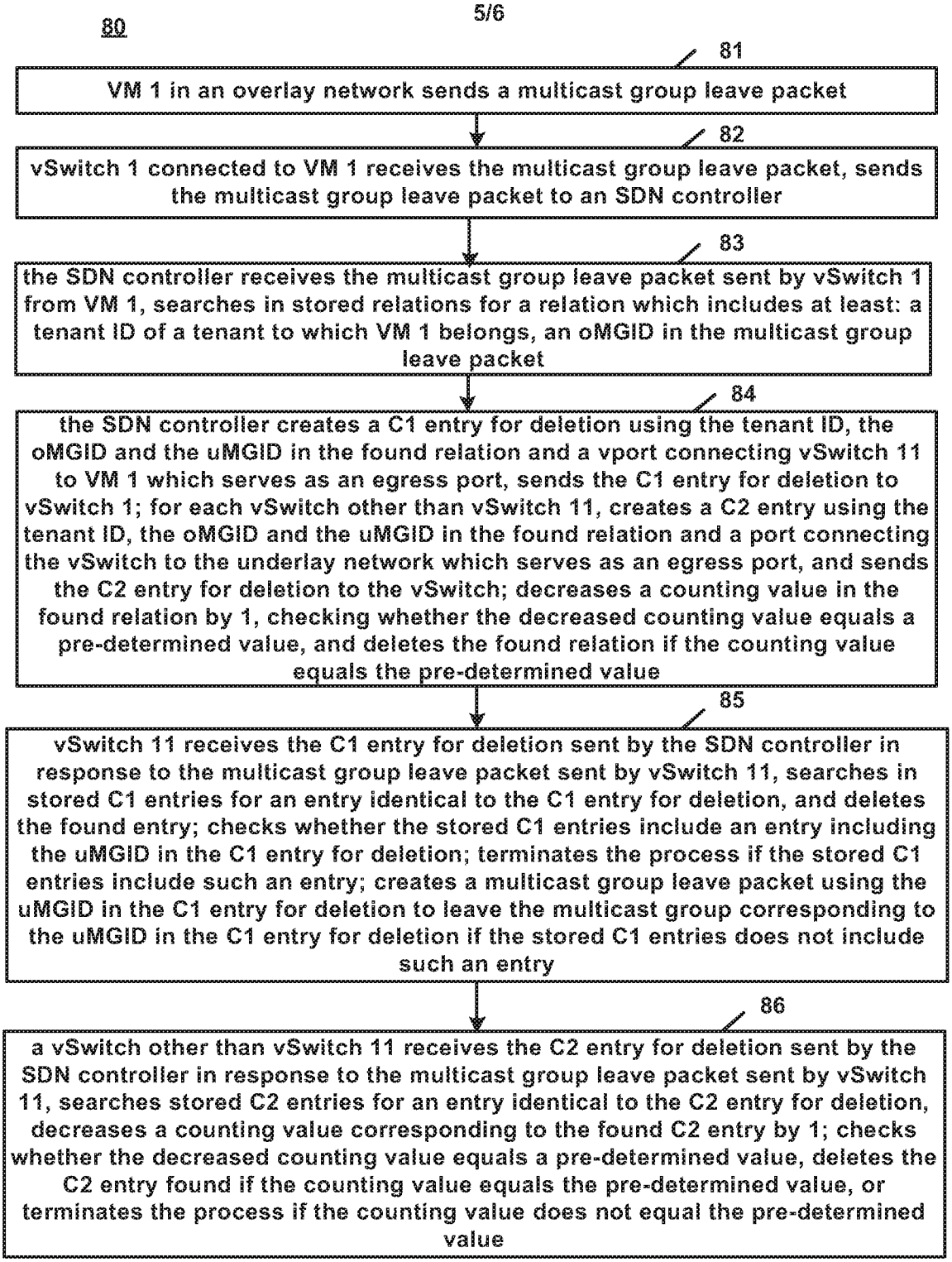

80
81
VM 1 in an overlay network sends a multicast group leave packet
82
vSwitch 1 connected to VM 1 receives the multicast group leave packet, sends the multicast group leave packet to an SDN controller
83
the SDN controller receives the multicast group leave packet sent by vSwitch 1 from VM 1, searches in stored relations for a relation which includes at least: a tenant ID of a tenant to which VM 1 belongs, an oMGID in the multicast group leave packet
84
the SDN controller creates a C1 entry for deletion using the tenant ID, the oMGID and the uMGID in the found relation and a vport connecting vSwitch 11 to VM 1 which serves as an egress port, sends the C1 entry for deletion to vSwitch 1; for each vSwitch other than vSwitch 11, creates a C2 entry using the tenant ID, the oMGID and the uMGID in the found relation and a port connecting the vSwitch to the underlay network which serves as an egress port, and sends the C2 entry for deletion to the vSwitch; decreases a counting value in the found relation by 1, checking whether the decreased counting value equals a pre-determined value, and deletes the found relation if the counting value equals the pre-determined value
85
vSwitch 11 receives the C1 entry for deletion sent by the SDN controller in response to the multicast group leave packet sent by vSwitch 11, searches in stored C1 entries for an entry identical to the C1 entry for deletion, and deletes the found entry; checks whether the stored C1 entries include an entry including the uMGID in the C1 entry for deletion; terminates the process if the stored C1 entries include such an entry; creates a multicast group leave packet using the uMGID in the C1 entry for deletion to leave the multicast group corresponding to the uMGID in the C1 entry for deletion if the stored C1 entries does not include such an entry
86
a vSwitch other than vSwitch 11 receives the C2 entry for deletion sent by the SDN controller in response to the multicast group leave packet sent by vSwitch 11, searches stored C2 entries for an entry identical to the C2 entry for deletion, decreases a counting value corresponding to the found C2 entry by 1; checks whether the decreased counting value equals a pre-determined value, deletes the C2 entry found if the counting value equals the pre-determined value, or terminates the process if the counting value does not equal the pre-determined value

FIG. 8

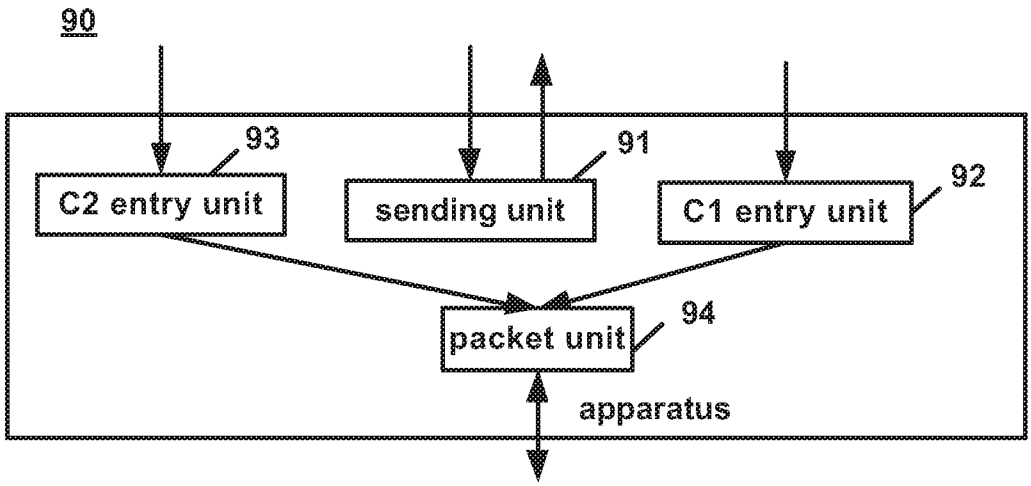

90
93
C2 entry unit
91
sending unit
92
C1 entry unit
94
packet unit
apparatus

FIG. 9

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2016/077480

A. CLASSIFICATION OF SUBJECT MATTER

H04L 12/761(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; 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)

CNTXT, CPRSABS, CNKI, VEN, IEEE: multicast, packet, group, overlay, underlay, network, virtual, switch, identity, encapsulat +, send, receive, message

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2014092907 A1 (VMWARE INC.ET AL.) 03 April 2014 (2014-04-03) the whole document	1-14
A	US 2014016501 A1 (KAMATH DAYAVANTI G.KAMATH ET AL.) 16 January 2014 (2014-01-16) the whole document	1-14
A	CN 104350714 A (HUAWEI TECH CO., LTD.) 11 February 2015 (2015-02-11) the whole document	1-14
A	CN 103684966 A (HUAWEI TECH CO., LTD.) 26 March 2014 (2014-03-26) the whole document	1-14

☐ 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
20 May 2016	**27 May 2016**
Name and mailing address of the ISA/CN **STATE INTELLECTUAL PROPERTY OFFICE OF THE P.R.CHINA** **6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China** Facsimile No. **(86-10)62019451**	Authorized officer **CHENG,Dong** Telephone No. **(86-10)62411276**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2016/077480

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
US	2014092907	A1	03 April 2014	JP	2015531212	A	29 October 2015
				EP	2885898	A1	24 June 2015
				AU	2013303200	B2	01 October 2015
				US	9210079	B2	08 December 2015
				WO	2014028094	A1	20 February 2014
				CN	104704778	A	10 June 2015
				AU	2013303200	A1	02 April 2015
US	2014016501	A1	16 January 2014	US	9094459	B2	28 July 2015
				CN	103546451	A	29 January 2014
				US	2015295819	A1	15 October 2015
CN	104350714	A	11 February 2015	EP	2966815	A1	13 January 2016
				EP	2966815	A4	13 April 2016
				WO	2015180084	A1	03 December 2015
				US	2015358232	A1	10 December 2015
CN	103684966	A	26 March 2014	WO	2015085788	A1	18 June 2015