Intarea Working Group
Internet-Draft
Intended status: Standards Track
Expires: 25 August 2025

C. Xie
J. Sun
China Telecom
X. Li
C. Bao
CERNET Center/Tsinghua University
M. Smith
21 February 2025

EVN6: Mapping of Ethernet Virtual Network to IPv6 Underlay for Transmission
draft-xls-intarea-evn6-03

Abstract

This document describes the mechanism of mapping of Ethernet Virtual Network to IPv6 Underlay for transmission. Unlike the existing methods, this approach places the Ethernet frames to be transmitted directly in the payload of IPv6 packets, i.e., L2 over IPv6, and uses stateless mapping to generate IPv6 source and destination addresses from the host’s MAC addresses, Ethernet Virtual Network identifier and site prefixes. The IPv6 packets generated in this way carry Ethernet frames and are routed to the destination site across public IPv6 network.

Table of Contents

1. Introduction

Ethernet Virtual Network is network model of Layer-2 built on top of the underlay to provide connectivity between dispersed customer sites across public network. This overlay L2 virtual network is used to carry the Ethernet data from the individual hosts in an encapsulated format over a logical tunnel, as if they were connected using the same LAN. Ethernet Virtual Network can serve scenarios such as campus networks, enterprise branch interconnections, data center networks, wide area IP bearer networks, and SD-WAN. There have been multiple solutions, they may differ in the types of underlying networks or encapsulation methods, besides, they usually serve different scenarios.

VXLAN [RFC7348] is a network virtualization technology which has been used mainly in data centers. VXLAN uses MAC-in-UDP encapsulation for packets, specifically, it encapsulates original Ethernet frames into

UDP packets. It then encapsulates the UDP packets with the IP header and Ethernet header of the physical network as outer headers, enabling these packets to be routed across the network like ordinary IP packets.

VPLS [RFC4762] make use of MPLS and VPN protocols to provide a virtual LAN between multiple locations. It is basically a way to provide Ethernet-based multi-point to multi-point communication over MPLS networks. VPLS operates by creating a virtual 'switch' at the customer's edge (CE) and the provider's edge (PE) of their respective networks.

The new approach, namely EVN6, proposed in this document aims to efficiently carry Ethernet Virtual Networks in IPv6 networks. It provides a methodology for dynamically creating a tunnel on the IPv6 network to transparently forward Ethernet frame when communication is required between a source and destination node in a Ethernet Virtual Network. In this scheme, Ethernet frame to be transmitted is directly placed in the payload field of IPv6 packet without adding additional payload, the MAC address of the hosts that needs to communicate, the identification of the Ethernet Virtual Network and the IPv6 prefix of the site can be used to generate outer IPv6 addresses. With EVN6 implementation, any two host can communicate, regardless of the underlying IPv6 network structure and other details. This document specifies EVN6's overall architecture, typical workflow, Layer-2 multicast and broadcast processing, etc.

1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

2. Terminology

The following terms are defined and used in this document,

EVN6: Multi-site Ethernet Virtual Network built on IPv6 network

E-ADPT: Ethernet Adaptor

IID: Interface Identifier(Section 2.5.1 of [RFC4291])

VEI: Virtual Ethernet Identification, VEI is used to identify and distinguish different Ethernet Virtual Network instances across the entire network, the length of VEI is 32 bits

MAC-VRF: A Virtual Routing and Forwarding table for Media Access Control (MAC) addresses on a PE（Section 3 of [RFC8365]）, it stores IPv6 site prefix, VEI of the Ethernet Virtual Network and other information of each MAC address

PE: Provider Edge Router

Pref6: Site prefix, Pref6 is a 64 bits subnet prefix (Section 2.5 of [RFC4291]) to identify one site of a given EVN6 instance

3. Overall Architecture

As a common underlay infrastructure, IPv6 network should simultaneously support multiple Ethernet Virtual Networks. To distinguish different Ethernet Virtual Network instances, VEI with a length of 32-bits is used to globally identify them, and it can identify up to 4.29 billion Ethernet Virtual Networks.

Generally, Ethernet Virtual Network consists of multiple sites distributed in different geographically locations, and each site is connected to the IPv6 network through local PE at the edge of the IPv6 network. The PE device supports Ethernet Virtual Network services by introducing E-ADPT functional subsystem. E-ADPT directly encapsulates the Ethernet data frames to be transmitted by the customer site into IPv6 packets and sends them to the IPv6 network. For the received IPv6 packets destined to one of this local sites, E-ADPT removes their packet header and restores the original Ethernet frames.

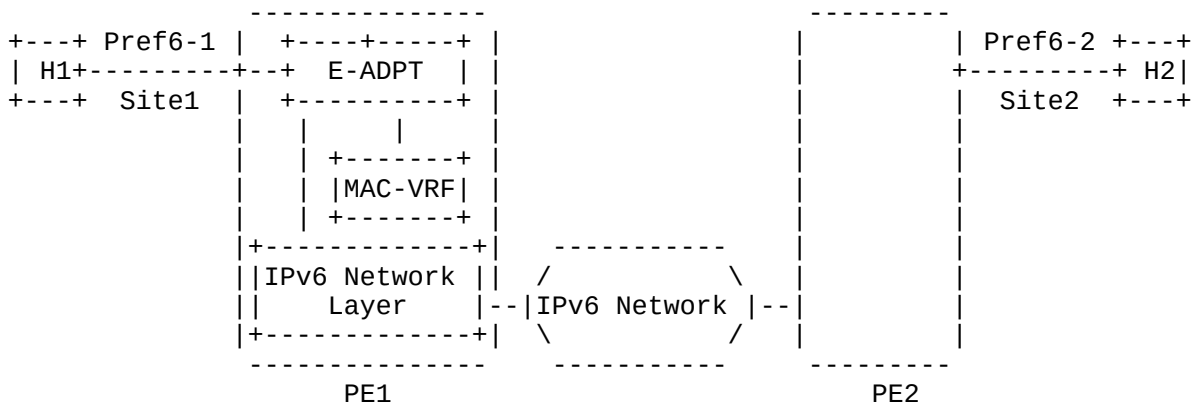

Figure 1: EVN6 System Architecture

For a given Ethernet Virtual Network, E-ADPT uses the IPv6 site prefix, i.e., Pref6, to identify different sites, so the Pref6 of different sites within a given Ethernet Virtual Network is also different. There are no special requirements for the type of address block used for Pref6, as long as it belongs to the global unicast address type and is reachable in global routing system. The Pref6 for each site can be allocated from the IPv6 address space owned by the operator. It should be noted that the length of Pref6 can be flexibly selected, it can be equal to or less than 64 bits. For Ethernet Virtual Network which has multiple sites, there is a 1: N relationship between the VEI and the site prefix of its sites.

In order to send Ethernet frames to the correct destination site through the IPv6 network, MAC-VRF table in PE is used to store the MAC addresses of all hosts in the Ethernet Virtual Network, the corresponding VEI of the Ethernet Virtual Network and Pref6 of the sites they belong to. The format of each record in MAC-VRF is shown in figure 2, it contains MAC address, VEI, prefix of the corresponding site, Group Policy ID and Host ID.

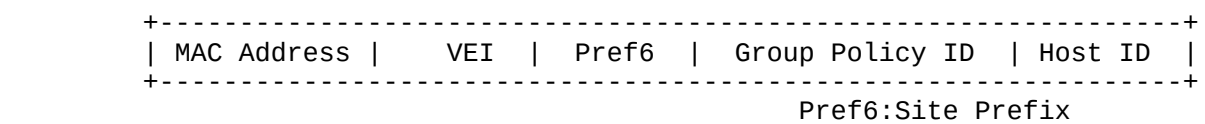

Figure 2: Structure of the Record in MAC-VRF

Group Policy ID: 24-bits identifier that indicates the source TSI Group membership being encapsulated by EVN6. The allocation of Group Policy ID values is outside the scope of this document.

Host ID: 20-bits identifier that indicates the membership of each host within the site it belongs to, the Host ID of each host is allocated from the Host ID Pool based on the host’s MAC address while ensuring its local uniqueness.

For E-ADPT, MAC-VRF provides a data foundation for encapsulating Ethernet frames into corresponding IPv6 packets, the data in it should be available before sending Ethernet frame to other sites, so the mechanism requires the sites to pre-send host MAC/Pref6 Mapping Advertisement to other sites. After receiving mapping relationship data of a host sent by other PEs, the PE stores the mapping data in the local MAC-VRF. The exchange of MAC/Pref6 can be carried out through the control layer, such as extending EVPN[RFC7432], however, this has been out of the scope of this document and will be discussed in other documents. When receiving Ethernet frame data sent by the host within the site, PE uses the destination MAC address as an index to search for the local MAC-VRF table. If a corresponding entry is

found, PE extracts its site prefix Pref6 and VEI value, then uses the process in section 4.2 to encapsulate the Ethernet frame in an IPv6 packet. Afterwards, the IPv6 data packet is transmitted to the IPv6 network.

4. Operation

In this section, the Ethernet Virtual Network in figure 3 is used as an example to illustrate the workflow, its profile includes: the VEI value is N1, it has three branch sites connected to PE1, PE2, and PE3, with site prefixes Pref6-1, Pref6-2, and Pref6-3, respectively. Hosts H1, H2, H3 and H4 are located at sites 1, site 2 and site 3, respectively.

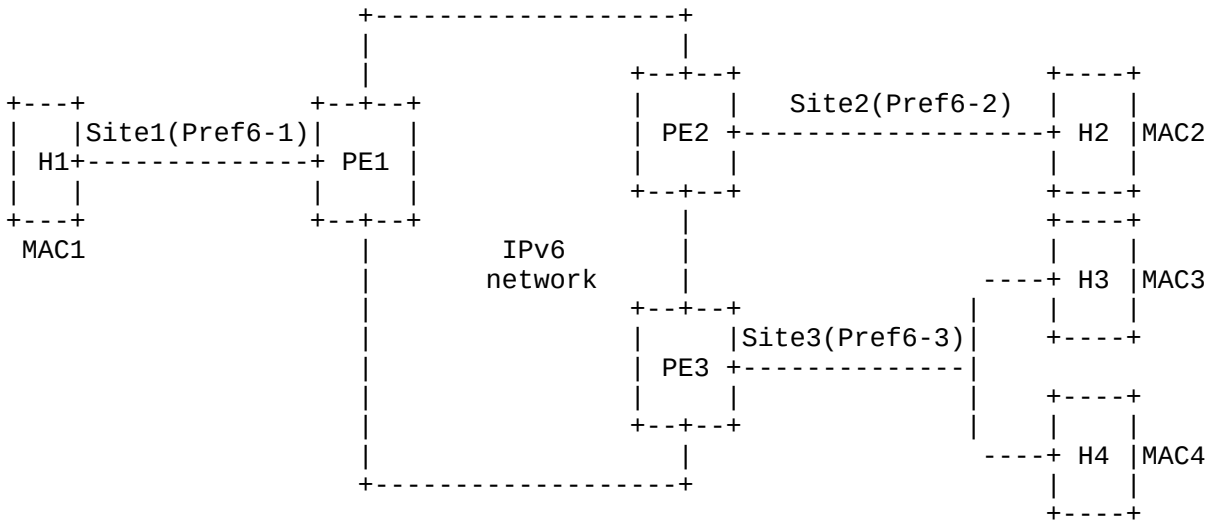

Figure 3: Diagram of Typical EVN6 Instance

The workflow of EVN6 is illustrated as follows:

4.1. Network Creation Procedures

Step 1: EVN6 network creation on each PE device

When creating an Ethernet Virtual Network instance on an IPv6 network, it should firstly enable the EVN6 function is in PE1, PE2, and PE3, then configure the relevant information of the Ethernet Virtual Network on this site, configure the Ethernet Virtual Network identifier VEI as N1, and set the site prefix Pref6 on the interface of the PE that the site accesses, indicating that the VEI of the

Ethernet Virtual Network to which the site belongs to is N1, and its local site prefix is Pref6. This process can be configured manually or using specific systems such as network management. When the EVN6 instance is running, the sites within it exchange host MAC/Pref6 Mapping through the connected PEs, as described in Section 3.

4.2. Data Transmission Procedures

Step 2: Host information searching

Host H2 in site 2 sends an Ethernet frame with the destination being Host H1 in site 1. Its frame header contains the MAC source address and MAC destination address, which are the MAC addresses of hosts H2 and H1, respectively. In this case, PE2 is the source PE and PE1 is the destination PE. After receiving the Ethernet frame in Site 2, PE2 uses the destination MAC address as an index to search for the local MAC-VRF table. If a corresponding entry is found, the Pref6 of remote site (i.e. Pref6-1) and VEI information are extracted; If not found, do not encapsulate and forward.

Step 3: Address mapping and frame encapsulation

In EVN6, each Ethernet frame needs to be associated with the VEI of the Ethernet Virtual Network to which the frame originates. Upon receiving the Ethernet frame, PE2 can determine its VEI value based on local configuration. Then the VEI obtained is divided into two sub-segments: VEI-S1 and VEI-S2, the first 16 bits are VEI-S1, and the last 16 bits are VEI-S2. VEI-S1 and VEI-S2 will be respectively put into IPv6 source and destination address of the new IPv6 packet.

IPv6 source and destination addresses are generated statelessly and their formats are shown in figure 4.

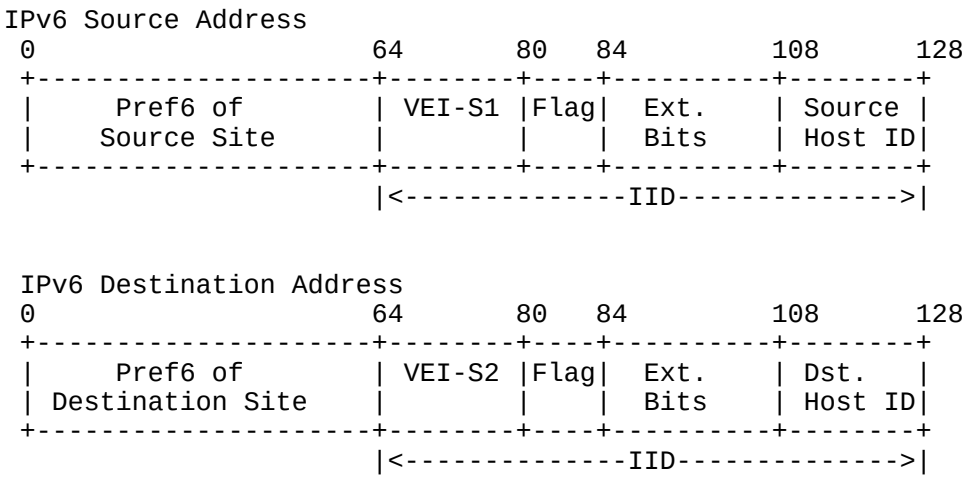

Figure 4: Formats of IPv6 Source and Destination Addresses

Each field of the IPv6 source address is illustrated as follows,

-Pref6 of Source Site: 64-bits in length with the value of Pref6-2, it is for identifing the site that the source host belongs to.

-VEI-S1: 16-bits in length, it starts from the 64th bit of source IPv6 address.

-Flag: 4-bits in length, it is a identification field in the format of GRRR. The first bit, i.e., G bit, is the Group Policy flag, and the last three digits are reserved bits, with a default value of 0. If the G bit is 1, the 24-bits Extensible Bits field is assigned to the Group Policy ID; If the G flag is 0, then all the 24 bits are set to zero.

-Extensible Bits: 24-bits in length, its value is related to the Flag field as mentioned above.

-Source Host ID: 20-bits in length, it is for identifying the site to which the source host belongs. The local site PE allocates the host ID from the Host ID Pool based on the host’s MAC address. Host ID is stored at the MAC-VRF table of local site.

Each field of IPv6 destination address is illustrated as follows,

-Pref6 of Destination Site: 64-bits in length with the value of Pref6-1, it is for identifing the site that the destination host belongs to.

-VEI-S2: 16-bits in length, it starts from the 64th bit of destination IPv6 address.

-Flag: 4-bits in length, it is a identification field in the format of GRRR. The first bit, i.e., G bit, is the Group Policy flag, and the last three digits are reserved bits, with a default value of 0. If the G bit is 1, the 24-bits Extensible Bits field is assigned to the Group Policy ID; If the G flag is 0, then all the 24 bits are set to zero.

-Extensible Bits: 24-bits in length, its value is related to the Flag field as mentioned above.

-Destination Host ID: 20-bits in length, it is for identifying the site to which the destination host belongs. If the frame is an unicast message, the Host ID is allocated from the host ID Pool of the destination site, and stored in the MAC-VRF of destination site; If it is a BUM message, the Host ID in the destination address is set to FFFFF.

Moreover, the Ethernet frame is put into the payload of IPv6 packet and the value of the "Next header" in the header is set to 143, indicating that the payload of the IPv6 packet is an Ethernet frame.

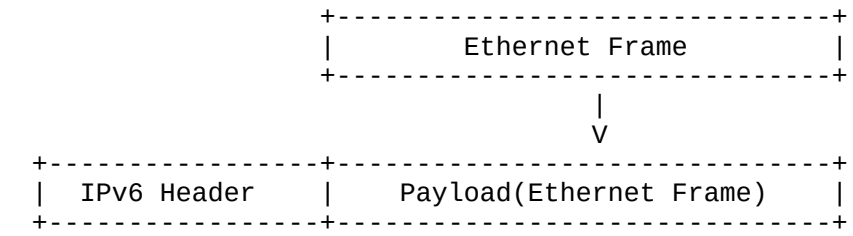

Figure 5: Encapsulation of Ethernet Frame into IPv6 Packet

After the IPv6 packet is generated, it is sent to the IPv6 network via the underlying IPv6 network layer.

Step 4: Packet forwarding in IPv6 network

When receiving an IPv6 packet, routers in an IPv6 network use the destination address in the packet to look up the routing table and forward it. Since the IPv6 destination address contains the site

prefix of the destination site, i.e. Pref6-1 in this case, which provides the egress PE of the packet, routers can forward the IPv6 packet carrying the Ethernet frame to the destination PE, i.e. PE1 in this case. It should be noted that this process does not require additional functionality for non-PE routers in the network, nor does it require extra IPv6 routing information to be added to the IPv6 network.

4.3. Data Receiving Procedures

Step 5: Packet de-capsulation and Ethernet frame restoration

As shown in figure 6, when receiving a IPv6 packet, the receiving PE, i.e., PE1, checks whether the destination address prefix matches the site prefix Pref6-1 on PE1? If yes, it extracts VEI-S1 and VEI-S2 from the IIDs of the source and destination IPv6 addresses, and concatenates them into VEI, then check if the VEI value is equal to N1? If yes, it then checks if the "Next header" value in the IPv6 header is 143? If yes, it then discards the IPv6 header, takes out the Ethernet frame, and sends the Ethernet frame to H1 within Site 1 based on its destination MAC address. Otherwise, the packet is discarded due to abnormal situation.

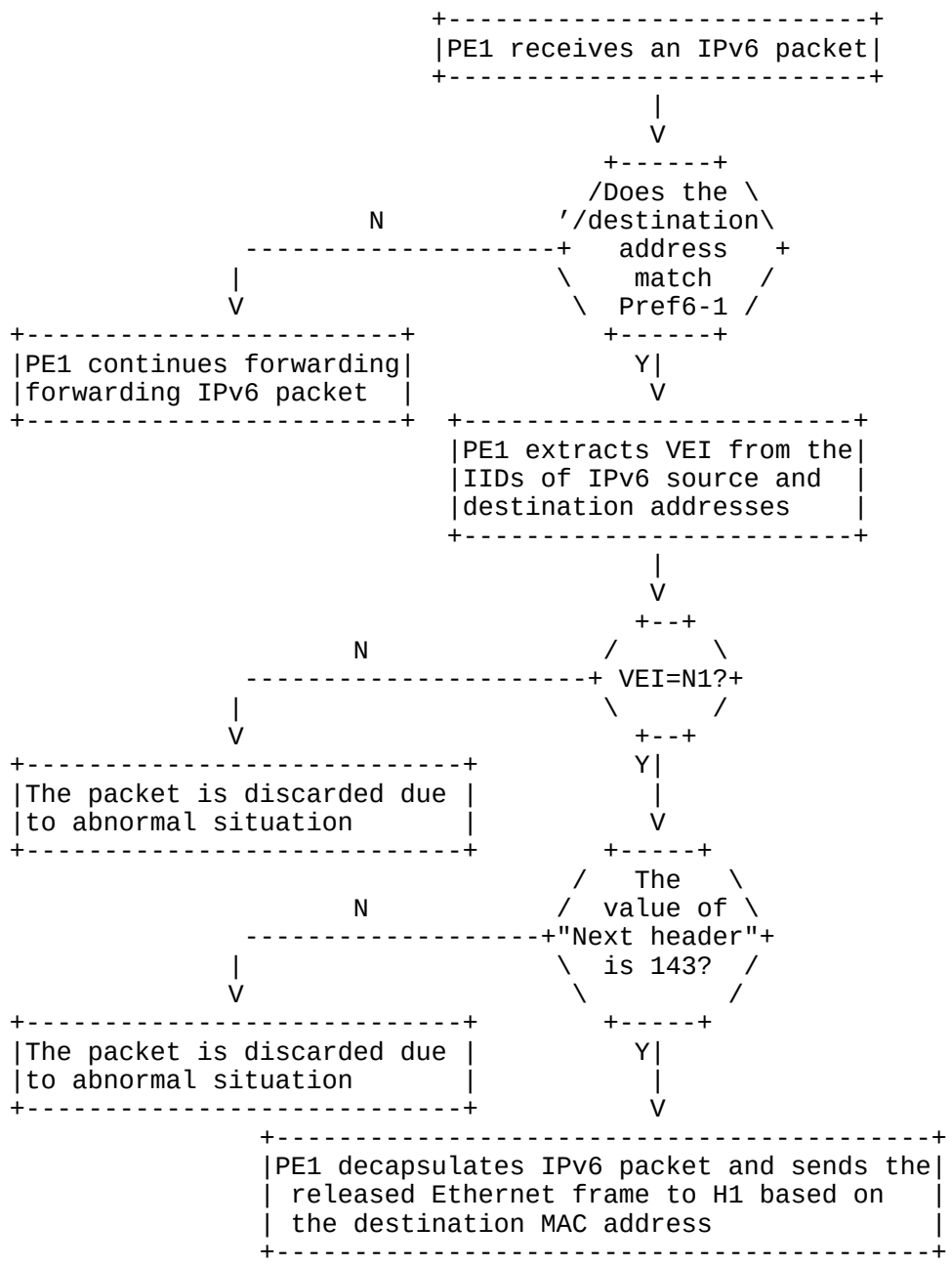

Figure 6: Process of Ethernet Frame Restoration in PE

5. Multicast and Broadcast

5.1. Multicast

Link layer multicast is used to send Ethernet frames to multiple members of a group, these members are distributed on different sites of the Ethernet virtual network. The case here is used to illustrate multicast process: VN1 is an instance of Ethernet virtual network over IPv6 underlay, it consists of N1 sites, contains M1 hosts distributed on different sites, and its VEI is vn1. Of all the sites, site-0 has the site prefix Pref6-0 and is connected to the local PE1.

G1 is a multicast group in instance VN1, it contains m1 members, and m1 is less or equal than M1. The members of G1 are distributed on n1 sites of instance VN1: site-1, site-2, ⋯, Site-n1, since these sites are partial of the total site set, n1 is less or equal than N1. The site prefixes of each site are Pref6-1, Pref6-2, ⋯ , Pref6-n1. For multicast, MAC-VRF also maintains related entries, with MAC addresses being multicast addresses. Per IEEE 802.1Q, multicast addresses of Ethernet have the least significant bit in the first octet set to 1. Due to the presence of multiple destination sites for a given group, there are multiple site prefixes in each entryas follows:

Multicast MAC Address	VEI	Length of Pref6	Pref6-1,Pref6-2,...,Pref6-n1

Figure 7: Record of Multicast in MAC-VRF

Host H1 in site-0 sends a multicast frame to group G1. When the E-ADPT in PE1 receives the frame, it will lookup its MAC-VRF with the destination multicast address of the frame as the key. When a matching entry is found and its VEI field is vn1, the list of site prefix is extracted from the entry. Each item in the list is the site prefix of the remote site where the members of group G1 are located on. Then, for each remote site, the following operations are performed recurrently by the E-ADPT of PE1,

{

- Generate the source IPv6 address with the source MAC address, vn1 and Pref6-0 using the method in section 4.2.

- Generate the source IPv6 address with destination multicast MAC address, vn1 and Pref6-k using the method in section 4.2.

- Generate the IPv6 header using the IPv6 source and destination addresses created above, encapsulate the frame into IPv6 packet, then send the new IPv6 packet into the IPv6-only network.

}

Through the above n1 cycles the multicast frame is encapsulated into IPv6 packet and send the data out.

When the packet traverses the IPv6-only network and reaches an egress PE, for instance, PE2. The E-ADPT of PE2 extracts its destination site prefix, i.e. Pref6-k, and VEI with value of vn1 from its IPv6 destination and source addresses. E-ADPT uses Pref6-k as the key to query the local MAC-VRF. If the corresponding entry is found, and the value of the VEI field is vn1, this indicates that one site attached to PE2 hosts the members of group G1, then, E-ADPT removes the IPv6 header and sends the released frame with the original multicast MAC address into this site.

5.2. Broadcast

Link layer broadcast is used to send Ethernet frames to any other hosts of the virtual Ethernet instance. Per 802.1Q, the destination address of the broadcast frame is FF-FF-FF-FF-FF-FF, and all hosts in the same Layer-2 network will receive the broadcast frame. EVN6 framework needs to support link layer broadcast as well. Herein, the case of instance VN1 in section 5.1 is used to illustrate the process of broadcast.

Host H1 located in site-0 sends a broadcast Ethernet frame. After receiving the frame and detecting that the destination MAC address is a broadcast address, PE1 needs to transmit it to each remote site of instance VN1. The E-ADPT in PE1 queries the MAC-VRF with the value of vn1 as the key, retrieves all the site prefixes of instance VN1: Pref6-1, Pref6-2, … , Pref6-N1. Then, for each remote site, the following operations are performed recurrently by the E-ADPT of PE1,

{

- Generate the source IPv6 address with the source MAC address, vn1 and Pref6-0 of site-0 using the method in section 4.2.

- Generate the source IPv6 address with the broadcast MAC address(ff:ff:ff:ff:ff:ff), Pref6-k and vn1 using the method in section 4.2.

- Generate the IPv6 header using the IPv6 source and destination addresses created above, encapsulate the frame into IPv6 packet, then send the new IPv6 packet into the IPv6-only network.

}

Through the above N1-1 cycles the broadcast frame is encapsulated into IPv6 packets and send the data out.

When the packet traverses the IPv6-only network and reaches an egress PE, for instance, PE2. The E-ADPT of PE2 extracts its Pref6-k and VEI with the value of vn1 from the destination IPv6 address and source IPv6 address of the packet. E-ADPT uses Pref6-k as the key to query the local MAC-VRF. If the corresponding entry is found and its VEI value is also vn1, this indicates that the site on PE2 is one site of instance VN1, then, E-ADPT remove the IPv6 header sends the broadcast frame to this site.

6. Security Considerations

In the EVN6 framework, PE devices located at the edge of the network encapsulate Ethernet frames in IPv6 packets and support transmission between different sites. When generating the outer IPv6 header, the PE device maps information such as the IPv6 address prefix of the site, the Mac address of the host, and the identity of the virtual network to the IPv6 address of the outer encapsulation header, which applies to both the source and destination addresses. In this way, the outer IPv6 address is dynamically generated based on information such as MAC address. For any host to host communication, even if the source and destination hosts are in the same virtual private network, when their source and destination address pairs are different, the generated outer encapsulated IP address is also different. The outer IPv6 address varies with the MAC address of the Ethernet frame, this is different from the traditional encapsulation scheme of pre-configuring tunnel IP addresses, as statically configured tunnel endpoint addresses are likely to become the target of DDOS attacks. In EVN6, tunnel encapsulation adopts dynamically generated tunnel endpoint IPv6 addresses, which avoids the occurrence of DDOS attacks caused by statically pre-configured tunnel addresses. From this perspective, this solution improves the security of Ethernet virtual networks.

7. Benefits and Advantages Analysis

Compared with existing overlay approaches, EVN6 is perceived to have the following major advantages,

1)Improved forwarding efficiency

EVN6 encapsulates Ethernet frame into IPv6 packet without extra encapsulation headers, compared with existing approaches, such as VxLAN, encapsulation and processing cost can be reduced.

2) High delivery flexibility

EVN6 service can be provisioned to customer site as long as its access to lPv6 Internet is available. There is no specific requirement for the interworking between ISPs, so it can be easily deployed in multi-operator environment.

3)Enhanced anti-DDoS capability

With EVN6 tunnel endpoint addresses are generated dynamically and there is no pre-configured static tunnel endpoint address, so the risk of DDoS attack to pre-configured static address can be avoided.

4)Source address based Traffic load-balancing

Since the outer IPv6 source and destination addresses are generated by mapping source and destination host MAC address, VEI and site prefixes, different hosts within the same site have different outer IPv6 addresses, so traffic load balancing can be implemented based on the source IPv6 addresses.

8. IANA Considerations

There are no other special IANA considerations.

9. Acknowledgment

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/info/rfc4291>.

[RFC7432] Sajassi, A., Ed., Aggarwal, R., Bitar, N., Isaac, A., Uttaro, J., Drake, J., and W. Henderickx, "BGP MPLS-Based Ethernet VPN", RFC 7432, DOI 10.17487/RFC7432, February 2015, <https://www.rfc-editor.org/info/rfc7432>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

10.2. Informative References

[IEEE.802.1D.2004]
"IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", March 2012, <http://standards.ieee.org/getieee802/download/802.1D-2004.pdf>.

[IEEE.802.1Q.2014]
"IEEE Standard for Information technology— Telecommunications and information exchange between systems— Local and metropolitan area networks— Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications", March 2012, <http://standards.ieee.org/getieee802/download/802.1Q-2014.pdf>.

[RFC4762] Lasserre, M., Ed. and V. Kompella, Ed., "Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signaling", RFC 4762, DOI 10.17487/RFC4762, January 2007, <https://www.rfc-editor.org/info/rfc4762>.

[RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman, "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861, DOI 10.17487/RFC4861, September 2007, <https://www.rfc-editor.org/info/rfc4861>.

[RFC7348] Mahalingam, M., Dutt, D., Duda, K., Agarwal, P., Kreeger, L., Sridhar, T., Bursell, M., and C. Wright, "Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks", RFC 7348, DOI 10.17487/RFC7348, August 2014, <https://www.rfc-editor.org/info/rfc7348>.

[RFC7364] Narten, T., Ed., Gray, E., Ed., Black, D., Fang, L., Kreeger, L., and M. Napierala, "Problem Statement: Overlays for Network Virtualization", RFC 7364, DOI 10.17487/RFC7364, October 2014, <https://www.rfc-editor.org/info/rfc7364>.

[RFC7365] Lasserre, M., Balus, F., Morin, T., Bitar, N., and Y. Rekhter, "Framework for Data Center (DC) Network Virtualization", RFC 7365, DOI 10.17487/RFC7365, October 2014, <https://www.rfc-editor.org/info/rfc7365>.

[RFC8365] Sajassi, A., Ed., Drake, J., Ed., Bitar, N., Shekhar, R., Uttaro, J., and W. Henderickx, "A Network Virtualization Overlay Solution Using Ethernet VPN (EVPN)", RFC 8365, DOI 10.17487/RFC8365, March 2018, <https://www.rfc-editor.org/info/rfc8365>.

[RFC8926] Gross, J., Ed., Ganga, I., Ed., and T. Sridhar, Ed., "Geneve: Generic Network Virtualization Encapsulation", RFC 8926, DOI 10.17487/RFC8926, November 2020, <https://www.rfc-editor.org/info/rfc8926>.

Authors' Addresses

Chongfeng Xie
China Telecom
Beiqijia Town, Changping District
Beijing
102209
China
Email: xiechf@chinatelecom.cn

Jibin Sun
China Telecom
Beiqijia Town, Changping District
Beijing
102209
China
Email: sunjb@chinatelecom.cn

Xing Li
CERNET Center/Tsinghua University
Shuangqing Road No.30, Haidian District
Beijing
100084
China

Email: xing@cernet.edu.cn

Congxiao Bao
CERNET Center/Tsinghua University
Shuangqing Road No.30, Haidian District
Beijing
100084
China
Email: congxiao@cernet.edu.cn

Mark Smith
PO BOX 521
Heidelberg 3084
Australia
Email: markzzzsmith@gmail.com