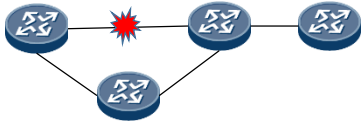


Encapsulation for BIER in Non-MPLS IPv6 Networks

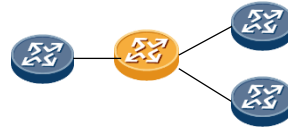
draft-xie-bier-6man-encapsulation-01

Jingrong Xie @Huawei
Liang Geng @China Mobile
Lei Wang @China Mobile
Mike McBride @Huawei
Gang Yan @Huawei
Yang Xia@Huawei

Problem Statement



Case 1: Fast reroute by using a bypass tunnel

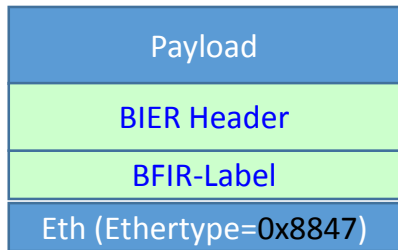


Case2: Non-incapable routers by using a bypass tunnel

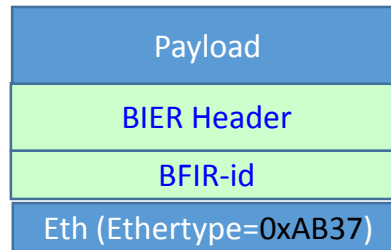
- To quote BIER-Arch RFC (RFC8279):
- In the event that unicast traffic to the BFR-NBR is being sent via a "bypass tunnel" of some sort, the BIER-encapsulated multicast traffic sent to the BFR-NBR SHOULD also be sent via that tunnel. This allows any existing "fast reroute" schemes to be applied to multicast traffic as well as to unicast traffic. ----> The above Case 1;
- unicast tunnels are used to bypass non-BFRs ----> The above Case 2;

Problem Statement (cont.)

- The Key of 'bypassing' is the capability of the MPLS Label Stacking.
- BIER MPLS Encapsulation can easily stacking on any MPLS Label (e.g. Bypass tunnel Label).
- While BIER Non-MPLS Encapsulation (or Eth Encapsulation) **can't** run over a bypass tunnel !
- Even more, BIER Eth Encapsulation **can't** run on links except Ethernet link !



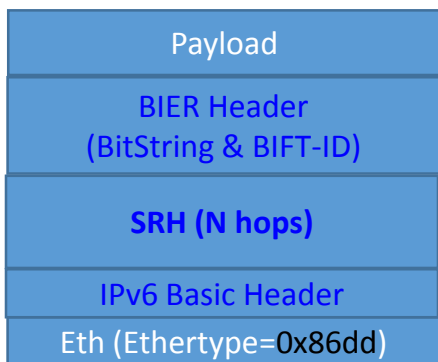
- Bypass tunnel Label can stack here.
- Can run on any link that support MPLS.



- Bypass tunnel Labels Can't Stack here !
- Can't run on links except Ethernet link !

Requirements & Considerations

- How can BIER IPv6 encapsulation behaves like MPLS ?
 - BIER-MPLS encapsulation over Various Links / Unicast MPLS bypass tunnel.
 - BIER-IPv6 encapsulation over Various Links / Unicast SRH bypass tunnel.



Where to put the BIER Header ?

- Opt-1: Re-invent a new IPv6 Extension Header.
- Opt-2: Re-use the existing IPv6 Extension Header.

How can BIER run over SRH ?

- Bypass tunnel uses SRH in IPv6 Non-MPLS Networks.
- SRH behaves like a Label Stack.

Checking existing IPv6 Options

RFC8200: When more than one extension header is used in the same packet, it is recommended that those headers appear in the following order:

IPv6 header

Hop-by-Hop Options header [Not use]

Destination Options header [Not use]----by every destination along the SRH.

Routing header [[SRH is here](#)]----consider BIER over SRH as a requirement.

Fragment header [Not use]

Authentication header [Not use]

Encapsulating Security Payload header [Not use]

Destination Options header [[BIER option TLV](#)]----by the final destination only.

Upper-Layer header [[VpnLabel+IPv4/IPv6 Packet](#)]----VpnLabel is not a Label.

Indicating the Payload

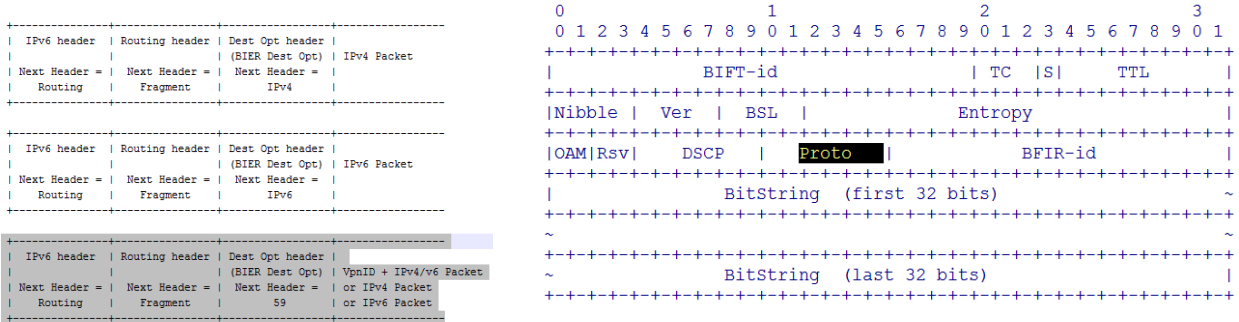
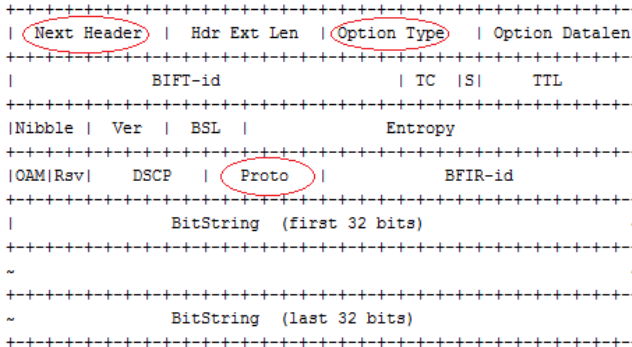


Figure 1: BIER Header

- When the BIER Destination Option Header is included in a Destination Option Header, then it SHOULD be the only Destination header.
- And the Payload format can be further identified by the 'Proto' field of the BIER Header, so the Destination Option Header is carried with a Next Header = 59 (The 3rd of the left 3 pictures).

BIER IPv6: Hop-by-hop



- **IPv6 Header**(DA=Multicast Addr) + **Dest Opt Hdr**(TLV<T=bier, L, V=BIER Hdr>)
- The IPv6 BIER Option is the only one TLV in the Destination Options, with a Option Type **BIER** [IANA].
- The Destination Options that carry a BIER Header, has a recommended Next Header Value of 59.
- Only be checked when the Destination Address is a Multicast Address indicating BIER.
- [IANA] is expected to specify a Multicast Address for **BIER**.
- [RFC82000]: Each extension header is an integer multiple of 8 octets long. **OK**.

BIER IPv6: Over SRH tunnel (optional)

Next Header	Hdr Ext Len	Routing Type	Segments Left
Last Entry	Flags	Tag	
Segment List[0] (128 bits IPv6 address)			
...			
Segment List[n] (128 bits IPv6 address)			
//			
// Optional Type Length Value objects (variable)			
//			

Next Header	Hdr Ext Len	Option Type	Option DataLen
BIFT-id		TC	TS
TTL			
Nibble	Ver	BSL	Entropy
OAM Rev	DSCP	Proto	BFIR-id
BitString (first 32 bits)			~
~			~
BitString (last 32 bits)			

Routing Header (Optional)

Next Header = 60 (Dest Opt Hdr)
 Routing Type = 4 (SRH)
 Last SID (SL[0]) = **Multicast Address**
 Penultimate SID (SL[1]) = **Bier SID**
 Optional TLV Objects = null

Destination Option Header

Next Header = 59 (No Next Header)
 Option Type = BIER (**IANA**)
 Option Value = BIER Header (RFC8296)
 Proto = 1/2/3/4/5/6 (RFC8296)

- **IPv6 Header**(DA=Unicast Addr) + **SRH** + **Dest Opt Hdr**(TLV<T=bier, L, V=BIER Hdr>)

Why use Dest Opt Header and so on

- [RFC8200] Defining [new IPv6 extension headers is not recommended](#), unless there are no existing IPv6 extension headers that can be used by specifying a new option for that IPv6 extension header.
- [RFC8200] it is [recommended that the Destination Options header is used](#) to carry optional information that must be examined only by a packet's destination node(s), because they provide better handling and backward compatibility.
- [RFC8200] Extension headers are not processed until the packet reaches the node (or [each of the set of nodes, in the case of multicast](#)) identified in the Destination Address field of the IPv6 header.
 - **The reason why we select Using [Multicast Addr + Dest Opt Hdr] to pilot the hop-by-hop replication.**
- [RFC6744] As of this writing, IPv6 Destination Options headers, and the options carried by such headers, [are extremely uncommon in the deployed Internet](#). So, it is expected that this Nonce Option commonly would be the only IPv6 Destination Option present in a given IPv6 packet.
 - **The reason why we suggest BIER Option TLV being the only IPv6 Destination Option.**
- The very rare Destination Options can be referred to: RFC8200, 2473,6275, 6744.

Thank you !