

Project Proposals: BIER Plugin

Contents

- [Name](#)
- [Repo Name](#)
- [Description](#)
 - [BIER Technique briefing](#)
 - [BIER architecture](#)
 - [BIER architecture](#)
 - [Use Cases](#)
- [Scope](#)
- [Resources Committed \(developers committed to working\)](#)
- [Initial Committers](#)
- [Vendor Neutral](#)
- [Meets Board Policy \(including IPR\)](#)
- [Presentation](#)

Name

BIER

Repo Name

bier

Description

BIER (Bit Indexed Explicit Replication) introduces a method for multicast flow forwarding, without storing states in every node along the multicast path. BIER provides optimal multicast forwarding through a "BIER domain" without requiring intermediate routers to maintain any multicast related per-flow state. That really simplifies configuration and states of intermediate routers. More information could be found at IETF: <https://datatracker.ietf.org/wg/bier/>.

SDN controller is supposed to support BIER future, to make devices support BIER more easily and BIER network configuration more simply.

BIER Technique briefing

A multicast data packet enters a BIER domain at a "Bit-Forwarding Ingress Router" (BFIR), and leaves the BIER domain at one or more "Bit-Forwarding Egress Routers" (BFERs). The BFIR router adds a BIER header to the packet. The BIER header contains a bit-string in which each bit represents exactly one BFER to forward the packet to. The set of BFERs to which the multicast packet needs to be forwarded is expressed by setting the bits that correspond to those routers in the BIER header.

[blocked URL](#)

The method of BIER is meant to configure a BFR-id to every bordered node in the domain, and flood the BIER configuration with IGP protocol. So the nodes could know all the BFR-id in the domain and how to reach each other. After the topology is synchronized, we could define a multicast flow, include a multicast address, an ingress node and a group of egress node, such as: Address-X/ Ingress-A/ Egress-D(0001),E(0100). Then the flow destination will be translated to a bit string 0101(bit 1 and bit 3 set) and push to the ingress node(node A). Now, the node A will forward the packets which destination is X to node B, with the bit-string 0101 encapsulated on the packet head. Then the packet will be forwarded one by one, and replicated when necessary. Until it reach the destination: node D and node E. This procedure is shown as figure bellow:

[blocked URL](#)

BIER architecture

A BIER domain could be defined as 3 layers like figure below:

[blocked URL](#)

And the layers definition is as table bellow:

BIER Layers	Description
Routing underlay	Flood the routing information, calculate the path on every node separately..
BIER forwarding layer	Receive BIER configuration for controller and flood it, generate BIER forwarding table, forward packets in BIER way.
Multicast flow overlay	Exchange multicast route in domain.
BIER controller	Manage the topology of BIER domain, configure every BIER node, receive multicast flow information and send to devices.

The BIER domain is established by 5 steps:

- step1: BFRs enable IGP protocol and finish flooding and path calculating.
- step2: BIER controller collect the topology of the domain, and allow to configure the node with BFR-id, prefix and so on.
- step3: BFRs exchange the BIER information with extend IGP protocol and generate BIER forwarding table separately.
- step4: BFRs exchange multicast route with PIM or MLD protocol.
- step5: Controller receive the multicast route, which will be translated to BIER flow information and pushed to ingress BFR(BFIR).

BIER architecture

[blocked URL](#)

Interactions with other ODL modules

- Create Yang modules with datastore and RPCs
- BIER Plugin will Query topology information and register notification of topology change
- BIER Plugin will Receive configuration from REST modules
- BIER Plugin will receive multicast flow information from devices through NETCONF
- BIER Plugin will push configuration and bit-string information to devices through NETCONF
- BIER Plugin will save configuration to ODL datastore

Components description

Components	Description
NB Plugin	Phase the configuration from restconf, implement a configuration query interface, notify app when BIER topology change.
Topo Manage	Manage BFR nodes information and configuration, also the topology of the BIER domain.
Flow Manage	Manage flow information received from multicast flow overlay protocol(BGP/PIM/MLD). These information will be used to calculate BIER flow bit string.
BIER Service	Calculate BIER flow path and bit-string with the information of BIER topology and flow information.
SB Plugin	Push the BIER configuration and flow bit string to BFR device by SB protocol.

Use Cases

Use cases of BIER are described at: <https://tools.ietf.org/html/draft-ietf-bier-use-cases-03>.

Scope

The scope of this project is mostly about implementing a BIER Plugin in ODL, make controller be able to cooperate with BFR device and generate multicast flow path manually or dynamically.

Resources Committed (developers committed to working)

- lu.chunhuai@zte.com.cn Username: lucina
- xiong.quan@zte.com.cn Username: xiongquan
- gu.min1@zte.com.cn Username: gu.min1
- dongzhp_bjy@ctbri.com.cn Username: Zhongping

Initial Committers

- lu.chunhuai@zte.com.cn Username: lucina
- xiong.quan@zte.com.cn Username: xiaogquan
- gu.min1@zte.com.cn Username: gu.min1
- dongzhp_bjy@ctbri.com.cn Username: Zhongping

Vendor Neutral

The project is made from scratch, no vendor code, logos nor is anything included.

Meets Board Policy (including IPR)

New Project. No Inbound Code Review required

Presentation

