

BIER BOF

IETF 91 – Honolulu, Hawaii

draft-wijnands-mpls-encapsulation-01

Presenter:

Jeffrey Zhang

Juniper

Authors:

IJsbrand Wijnands (editor)

Cisco

Eric Rosen (editor)

Juniper

Andrew Dolganow

Alcatel-Lucent

Jeff Tantsura

Ericsson

Sam Aldrin

Huawei

Parameters for BIER Forwarding of Data Packets

- Set of Egress BFRs (BFERs)
 - Encoded as set of 1's in a BitString
 - To interpret BitString, must know:
 - *Length* of BitString
 - *SetId*: which BFR-id is represented by low-order bit
- Which underlay is being used
 - To find *next hops* for each packet
- Entropy
 - For ECMP
- These parameters must be inferable from data packet header

Encaps

dispatch *payload* (part of packet beneath BIER header)

- Allow payload to be MPLS packet

Strategy Behind Design of Encaps

- Since BIER already has IGP-based control plane, needed to advertise BIER Ids, etc.
 - Piggyback on this to allow each BFR to assign an MPLS label to the triple: $\langle \text{SetId}, \text{BitString Length}, \text{underlay} \rangle$
 - Piggyback on this to allow each BFR to assign an MPLS label to the triple: $\langle \text{SetId}, \text{BitString Length}, \text{underlay} \rangle$
 - (Detail: OSPF draft actually assigns *label range* to *sequence of SetIds*, instead of assigning single label to single SetId) , instead of
 - No additional MPLS control protocols are needed
- No additional MPLS control protocols are needed
- MPLS label serves as a (locally significant) lookup key for this triple
- MPLS label serves as a (locally significant) lookup key for this triple
 - Appears as *bottom label* of label stack
 - Label precedes BIER header:
 - Appears as *bottom key* of *Bit Index Forwarding Table*
- Serves as lookup key for proper *Bit Index Forwarding Table*
- Additional labels (as needed for app) may follow BIER header, as part of payload
- Additional labels (as needed for app) may follow BIER header, as part of payload
- Integrates BIER well with MPLS transport and with MPLS-based applications
- Integrates BIER well with MPLS transport and with MPLS-based applications

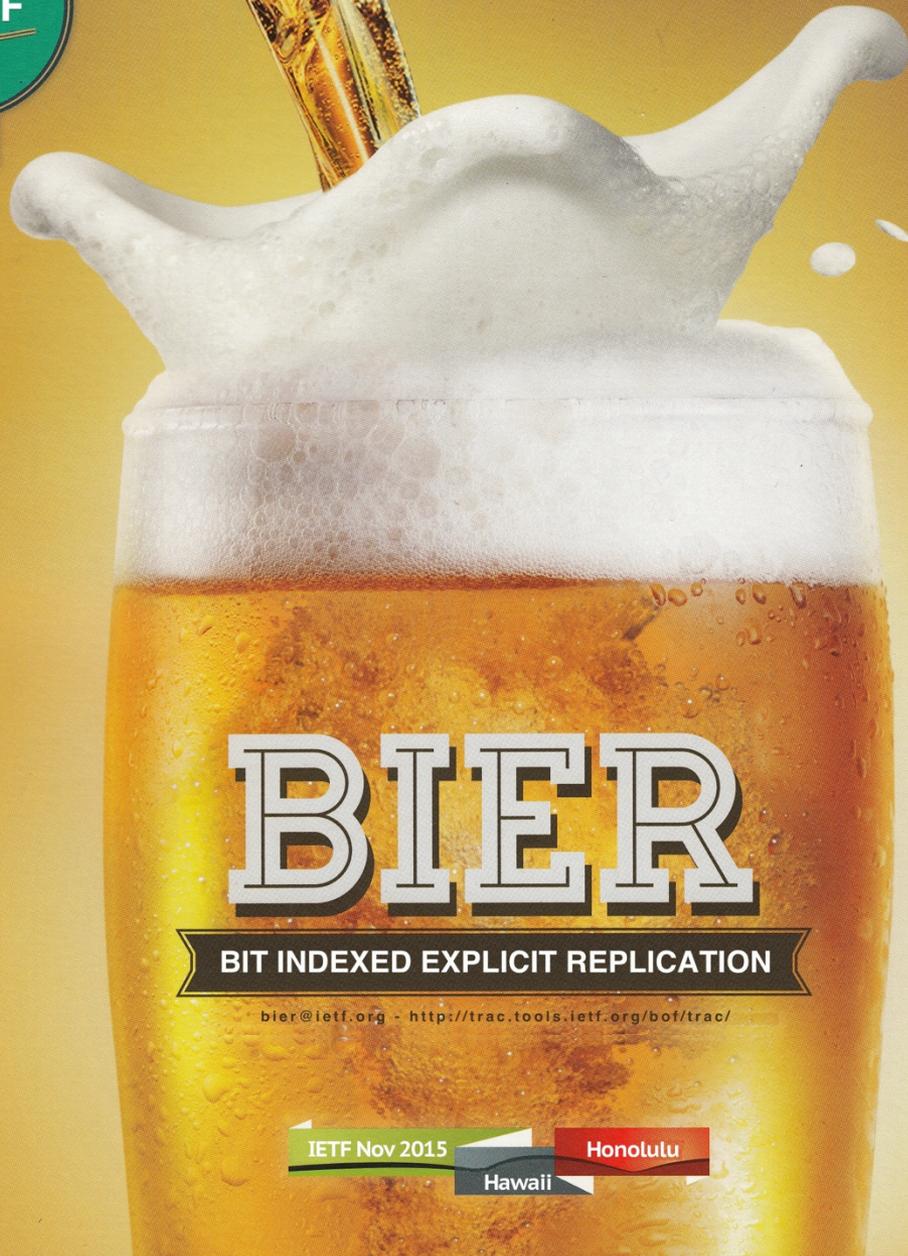
Advantages of Integration with MPLS

- Flexibility to add additional forwarding parameters without changing the encapsulation format
 - Example: maybe the best way to identify an underlay is with a TLV; one could bind a label to a TLV, but one wouldn't want the TLV in the data packet encapsulation header.
- Leverages MPLS forwarding procedures
 - Label maps to *Bit Index Forwarding Table*
 - Reduces need to string together lookup key out of multiple header fields
 - Very simple integration with MPLS protection schemes
- No need for additional layer 2 codepoints
- When MPLS based FRR is used, no need for special label to indicate payload is BIER

Controversies re MPLS Integration

- “I like BIER, but I hate MPLS”
 - Rename the “MPLS Label” to be the “BIER Lookup Key”
- “Having to maintain and distribute locally significant lookup keys requires too much state and protocol.”
 - Compared to the amount of state and protocol required to maintain/distribute the BFR-ids???
- “My hardware can’t swap header fields”
 - Hard to do BIER (or even IP) if you can’t modify the header fields!
- “But isn’t it better if *one size fits all?*”
 - No.
- A longer, non-MPLS, encapsulation could easily be developed
 - may be a viable option in some environments,
 - but that doesn’t eliminate the advantages of using the MPLS encapsulation.

NL



BIER

BIT INDEXED EXPLICIT REPLICATION

bier@ietf.org - <http://trac.tools.ietf.org/bof/trac/>

IETF Nov 2015



Honolulu

Thank you.