BIER Working Group
Internet-Draft
Intended status: Standards Track
Expires: 21 March 2024

R. Chen
ZTE Corporation
Fangwei. Hu
Individual
Zheng. Zhang
Xianxia. Dai
ZTE Corporation
Mahesh. Sivakumar
Juniper networks
18 September 2023

YANG Data Model for BIER Protocol
draft-ietf-bier-bier-yang-08

Abstract

This document defines a YANG data model that can be used to configure and manage devices supporting Bit Index Explicit Replication"(BIER). The YANG module in this document conforms to the Network Management Datastore Architecture (NMDA).

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 21 March 2024.

Copyright Notice

Copyright (c) 2023 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (https://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights

and restrictions with respect to this document. Code Components extracted from this document must include Revised BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Revised BSD License.

Table of Contents

1. Introduction 2
  1.1. Requirements Language 3
  1.2. Terminology 3
  1.3. Tree Diagrams 3
  1.4. Prefixes in Data Node Names 3
2. Design of Data Model 4
3. Module Structure 4
4. Configuration 5
5. IGP Control-Plane Configuration 6
6. Notifications 6
7. YANG Module for BIER 7
8. IANA Considerations 16
9. Security Considerations 17
10. Acknowledgements 17
11. Normative References 17
Authors' Addresses 18

1. Introduction

[RFC8279] describes a new architecture for the forwarding of multicast data packets. Known as "Bit Index Explicit Replication"(BIER), that architecture provides optimal forwarding of multicast data packets through a "multicast domain".

YANG [RFC7950] is a data modeling language that was introduced to model the configuration and operational state of a device managed using network management protocols such as the Network Configuration Protocol (NETCONF) [RFC6241] or RESTCONF [RFC8040]. YANG is now also being used as a component of other management interfaces, such as command-line interfaces (CLIs).

This document defines a YANG data model that can be used to configure and manage devices supporting Bit Index Explicit Replication"(BIER). The YANG module in this document conforms to the Network Management Datastore Architecture (NMDA).

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.

1.2. Terminology

The terminology for describing YANG data models is found in [RFC7950].

The following abbreviations are used in this document and the defined model:

BIER: Bit Index Explicit Replication [RFC8279]

BFR: Bit Forwarding Router [RFC8279]

BIFT: Bit Index Forwarding Table [RFC8279]

BSL: BitStringLength [RFC8279]

>SI: Set Identifier [RFC8279]

IS-IS: Intermediate System-to-Intermediate System

OSPF: Open Shortest Path First

1.3. Tree Diagrams

Tree diagrams used in this document follow the notation defined in [RFC8340].

1.4. Prefixes in Data Node Names

In this document, names of data nodes, actions, and other data model objects are often used without a prefix, as long as the context clearly indicates the YANG module in which each name is defined. Otherwise, names are prefixed using the standard prefix associated with the corresponding YANG module, as shown in Table 1.

Prefix	YANG Module	Reference
yang	ietf-yang-types	[RFC6991]
inet	ietf-inet-types	[RFC6991]
if	ietf-interfaces	[RFC8343]
rt	ietf-routing	[RFC8349]

Table 1: Prefixes and Corresponding YANG Modules

2. Design of Data Model

The data model can be used to configure and manage BIER protocol [RFC8279] features. The operational state data and statistics can be retrieved by this model. The protocol-specific notifications are also defined in the model.

This model is used to consistently provision the BIER parameters for one or more BIER subdomains across all BFIR/BFR/BFER. This configuration will also be read by BIER enabled IGPs, such as IS-IS [RFC8401] OSPFv2 [RFC8444] and OSPFv3[I-D.ietf-bier-ospfv3-extensions] to learn the BIER parameters. In scenarios where the IGP protocol is not used/available, this model defines the writeable BIFT and all BFIR/BFR/BFER can get the BIFT directly from the controller, or by any other ways such as configuration.

3. Module Structure

The ietf-bier YANG module augments the routing container in the ietf-routing model [RFC8349] with a BIER container and defines generic BIER configuration and operational state. This module is augmented by modules supporting different data planes.

```
module: ietf-bier
  augment /rt:routing:
    +--rw bier
       +--rw sub-domain* [sub-domain-id address-family]
       |  +--rw sub-domain-id                   uint8
       |  +--rw address-family                  identityref
       |  +--rw bfr-prefix?                     inet:ip-prefix
       |  +--rw underlay-protocol-type?         underlay-protocol-type
       |  +--rw mt-id?                          uint16
```

```
       |  +--rw bfr-id?                        uint16
       |  +--rw bsl?                           bsl
       |  +--rw igp-algorithm?                 uint8
       |  +--rw bier-algorithm?                uint8
       |  +--rw load-balance-num?              uint8
       |  +--rw encapsulation* [bsl encapsulation-type]
       |     +--rw bsl                         uint16
       |     +--rw encapsulation-type          identityref
       |     +--rw max-si?                     uint8
       |     +--rw in-bift-id?
       |        +--rw:(in-bift-id-base)
       |        |  +--rw in-bift-id-base?      uint32
       |        +--rw (in-bift-id-encoding)
       |           +--rw in-bift-id-encoding   boolean
       +--rw bift* [bfr-id]
            +--rw bfr-id                       bsl
            +--rw birt-bitstringlength* [bsl]
               +--rw bsl                       bsl
               +--rw bfr-nbr* [bfr-nbr]
                  +--rw bfr-nbr                inet:ip-prefix
                  +--rw encapsulation-type?    identityref
                  +--rw our-bift-id?
                     +--rw:(out-bift-id)
                     |  +--rw out-bift-id?       uint32
                     +--rw:(out-bift-id-encoding)
                        +--rw out-bift-id-encoding    boolean

notifications:
    +---n bfr-id-collision
    |   +--ro bfr-id-collisions*[]
    |      +--ro received-bfr-id?         uint16
    +---n bfr-id-out-of-range
    |   +--ro received-bfr-id?            uint16
    +---n bfr-zero
    |   +--ro ipv4-bfr-prefix?            inet:ipv4-prefix
    |   +--ro ipv6-bfr-prefix?            inet:ipv6-prefix
    +---n sub-domain-id-collision
        +--ro received-sub-domain-id?     uint16
        +--ro received-mt-id?             uint16
```

4. Configuration

The ietf-bier YANG module augments the routing container in the ietf-routing model [RFC8349] with a BIER container and defines generic BIER configuration. It includes:

sub-domain:Defines the relevant BIER information of the BIER subdomain, such as configuring the BIER domain identifier, configuring the BFR prefix in the BIER subdomain, configuring the Underlay protocol for advertising BIER information, etc. It contains two in-bift-id generation methods, one is direct configuration, the other is calculated based on <BSL,SD,SI>. The leaf "in-bift-id" is the first BIFT-id of the BIFT-id range. The "BIFT-id range" is the set of 20-bit values beginning with the in-bift-id and ending with (in-bift-id + (Max SI)).The leaf "in-bift-id-encoding" is defined as a boolean, used to enable or disable whether to calculate in-bift-id based on < BSL,SD,SI>.

bift: Defines the Bit Index Forwarding Table. The grouping "bfr-nbr" is to define the BFR neighbor, and it contains two bift-id generation methods, one is direct configuration, the other is calculated based on < BSL,SD,SI>. The leaf "out-bift-id-encoding" is defined as a boolean, used to enable or disable whether to calculate out-bift-id based on < BSL,SD,SI>.

5. IGP Control-Plane Configuration

Support of BIER extensions for a particular IGP control plane is achieved by augmenting routing-protocol configuration with BIER extensions. This augmentation SHOULD be part of the routing-protocol YANG modules as not to create any dependency for implementations to support BIER extensions for all routing protocols.

This module defines groupings that SHOULD be used by IGP BIER modules.

The "enabled" leaf enables BIER extensions for the routing-protocol instance.

The "sub-domain" container controls the routing-protocol instance's advertisement of the relevant BIER information of the BIER subdomain and the processing of received the relevant BIER information of the BIER subdomain.

6. Notifications

The model defines the following notifications for BIER.

This module defines groupings that SHOULD be used by IGP BIER modules.

bfr-id-collision: Raised when control-plane-advertised BFR ID have conflicts.

bfr-id-out-of-range: Raised when control-plane-advertised BFR ID is larger than locally configured (bsl * max-si).

bfr-zero: Raised when invalid value associated with prefix.

sub-domain-id-collision: Raised when control-plane-advertised Sub domain ID have conflicts.

7. YANG Module for BIER

```
<CODE BEGINS> file "ietf-bier@2023-09-16.yang"
module ietf-bier {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-bier";
  prefix "bier";

  import ietf-routing{
    prefix "rt";
        reference
          "RFC 8349: A YANG Data Model for Routing Management (NMDA
      Version)";
   }
  import ietf-interfaces{
    prefix "if";
    reference
          "RFC 8343: A YANG Data Model for Interface Management";
  }
  import ietf-inet-types{
    prefix "inet";
        reference
      "RFC 6991: Common YANG Data Types";

  }
  import ietf-isis{
    prefix "isis";
        reference "RFC 9130: YANG Data Model for the IS-IS Protocol";
  }

  import ietf-ospf{
    prefix "ospf";
        reference "RFC 9129: YANG Data Model for the OSPF Protocol";
  }

  organization
    "IETF BIER(Bit Indexed Explicit Replication) Working Group";
  contact
        "WG Web:   &lt;https://datatracker.ietf.org/wg/bier/&gt;
     WG List:  &lt;mailto:bier@ietf.org&gt;
```

```
     WG Chair: Tony Przygienda
                   &lt;mailto:tonysietf@gmail.com&gt;

     WG Chair: Greg Shepherd
                   &lt;mailto:gjshep@gmail.com&gt;


     Editor:   Ran Chen
                   &lt;mailto:chen.ran@zte.com.cn&gt;
     Editor:   Fangwei Hu
                   &lt;mailto:hu.fangwei@zte.com.cn&gt;
     Editor:   Zheng Zhang
                   &lt;mailto:zhang.zheng@zte.com.cn&gt;
     Editor:   Xianxian Dai
                   &lt;mailto:dai.xianxian@zte.com.cn&gt;
     Editor:   Mahesh Sivakumar
                   &lt;mailto:masivaku@cisco.com&gt;
        ";
  description
    "The YANG module defines a generic configuration model
         for BIER.;

    This YANG module conforms to the Network Management
    Datastore Architecture (NMDA), as described in RFC 8242.

    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 (RFC 2119) (RFC 8174) when, and only when,
    they appear in all capitals, as shown here.

    Copyright (c) 2022 IETF Trust and the persons identified as
    authors of the code.  All rights reserved.

    Redistribution and use in source and binary forms, with or
    without modification, is permitted pursuant to, and subject to
    the license terms contained in, the Revised BSD License set
    forth in Section 4.c of the IETF Trust's Legal Provisions
    Relating to IETF Documents
    (https://trustee.ietf.org/license-info).

        This version of this YANG module is part of RFC XXXX
    (https://www.rfc-editor.org/info/rfcXXXX); see the RFC itself
    for full legal notices.";

        reference
    "RFC XXXX: YANG Data Model for BIER";
```

```
    revision 2023-09-12 {
      description
        "initial version.";
      reference
        "RFC XXXX: YANG Data Model for BIER ";
    }

 /* Identities */
  identity bier-encapsulation{
    description
    "Base identity for BIER encapsulation.";
  }
  identity bier-encapsulation-mpls {
    base bier-encapsulation;
    description
      "This identity represents MPLS encapsulation for bier.";
  }
  identity bier-encapsulation-ipv6 {
    base bier-encapsulation;
    description
      "This identity represents ipv6 encapsulation for bier.";
  }
  identity bier-encapsulation-ethernet {
    base bier-encapsulation;
      description
        "This identity represents ethernet encapsulation for bier.";
  }


  identity address-family {
    description
      "Base identity from which identities describing address
      families are derived.";
  }
  identity ipv4 {
    base address-family;
    description
      "This identity represents an IPv4 address family.";
  }
  identity ipv6 {
    base address-family;
      description
        "This identity represents an IPv6 address family.";
  }



  /* typedef */
```

```
  typedef underlay-protocol-type {
    type enumeration {
      enum IS-IS {
        description
           "This BIER subdomains configuration can be read and
                   advertise by BIER enabled IS-IS.";
      }
      enum OSPF {
        description
          "This BIER subdomains configuration can be read and
                  advertise by BIER enabled OSPF.";
      }
      enum BGP {
        description
          "This BIER subdomains configuration can be read and
                  advertise by BIER enabled BGP.";
      }
    }
    description
      "List of the underlay protocol to be supported.";
  }


typedef bsl {
    type enumeration {
      enum IS-IS {
        description
           "This BIER subdomains configuration can be read and
                   advertise by BIER enabled IS-IS.";
      }
      enum OSPF {
        description
          "This BIER subdomains configuration can be read and
                  advertise by BIER enabled OSPF.";
      }
      enum BGP {
        description
          "This BIER subdomains configuration can be read and
                  advertise by BIER enabled BGP.";
      }
    }
    description
      "list of the underlay protocol to be supported.";
  }

  augment "/rt:routing" {
    description
      "This augments routing-instance configuration with bier.";
```

```
  container bier{
    description
      "bier subdomain configuration.";
    list sub-domain {
      key "sub-domain-id address-family";
          description
                "The parameters of the BIER subdomain. "

          leaf sub-domain-id {
        type uint16;
        description
          "The bier sub-domain-id";
       }

      leaf address-family {
        type identityref {
        base address-family;
        }
        mandatory true;
        description
          "Address family.";
       }

      leaf bfr-prefix {
        type inet:ip-prefix;
        description
          "the bfr prefix.";
       }

          leaf underlay-protocol-type {
            type underlay-protocol-type;
        description
          "List of the underlay protocol to be supported..";
       }

      leaf mt-id {
        type uint16;
        description
          "The multi-topology identifier";
       }

      leaf bfr-id {
        type uint16;
        description
          "Configure the unique BFR-id value within the BIER
                  subdomain for the BFIR/BFER device, and BFR doesnot
                  need a BFR-id, but for diagnostics purposes of the IGP,
                  highly recommended to assign one - but beyond max-si*bls.";
```

```
       }

          leaf bsl {
            type bsl;
        description
          "The length of the bitstring in the BIER encapsulation
                   within the BIER subdomain.";
       }

      leaf igp-algorithm {
        type uint8;
            default "0";
          description
            "Calculation type value ranges from 0 to 255 both
                        inclusive from the IGP Algorithm Types registry
                        defined under Interior Gateway Protocol (IGP)
            Parameters IANA registries.If the required calculation
                        type is Shortest Path First, the value 0 SHOULD appear
                        in this field.";
       }

      leaf bier-algorithm {
        type uint8;
        description
          "Calculation type value ranges from 0 to 255 both inclusive
           from the BIER Algorithm registry.Specifies a BIER-specific
                   Algorithm and BIER-specific Constraints used to either
modify,
                   enhance, or replace the calculation of underlay paths to
reach
                   other BFRs as defined by the IPA value as defined in
RFC9272.";
       }

      leaf load-balance-num {
        type uint8;
        description
          "The multicast load balance num.";
       }

      list encapsulation {
        key "bsl encapsulation-type";
            description
          "The BIER encapsulation type.When MPLS is used as the
                  transport, the Bit Indexed Forwarding Table (BIFT) is
                  identified by a MPLS Label. When non-MPLS transport is
                  used, the BIFT is identified by a 20bit value.";
        leaf bsl{
```

```
          type bsl;
          description
            "The length of the bitstring in the BIER encapsulation
```

```
                     within the BIER subdomain.";
        }
        leaf encapsulation-type {
          type identityref {
          base bier-encapsulation;
        }
          description
            "The BIER encapsulation that can be used in either
                     MPLS networks or non-MPLS networks.";
        }
            leaf max-si {
          type uint16;
          description
            "Maximum Set Identifier.The SI value in the subdomain
                    is an integer from 0 to max-si.";
        }
            container in-bift-id {
                  description
                  "In BIFT-ID specification.";
                  choice in-bift-id {
                    default "in-bift-id-base";
                        description
              "Options for specifying in-bift-id";
                        case in-bift-id-base {
                          leaf in-bift-id-base {
                            type uint32;
                 description
                  "The first BIFT ID value, there are maximum SI+1 BIFT
                           IDs in total as define in RFC8401.";
                 }
                        }
                        case in-bift-id-encoding {
                          leaf in-bift-id-encoding {
                 type boolean;
                 default "false";
                        description
                  "setting this attribute to 'true' will enable
                           calculation of in-bift-id based on <BSL, SD, SI>.";
                }
            }
           }
        }
      }
    }
          list bift {
            key "bfr-id";
            description
              "BIER forwarding tabel."
```

```
            leaf bfr-id {
          type uint16;
          description
            "The unique BFR-id value within the BIER
                    subdomain for the BFIR/BFER device.";
        }
            list birt-bitstringlength {
              key "bsl";
                  description
                    "specify BSL's bfr-nbr, encapsulation-type and
                        out-bift-id in the BIER forwarding tabel.";
                  leaf bsl {
                type bsl;
            description
              "Configure the bitstring length in BIFT in the
                          BIER subdomain";
           }
          list bfr-nbr {
            key bfr-nbr;
            description
              "bfr-nbr.";
            leaf bfr-nbr {
              type inet:ip-prefix;
              description
                "bfr-nbr.";
            }
            leaf encapsulation-type {
              type identityref {
              base bier-encapsulation;
            }
              description
                "The BIER encapsulation that can be used in either
                         MPLS networks or non-MPLS networks.";
            }
                    container out-bift-id {
                      description
                        "Out BIFT-ID specification.";
                      choice out-bift-id {
                        default "out-bift-id";
                            description
                  "Options for specifying out-bift-id";
                            case out-bift-id {
                              leaf out-bift-id {
                              type uint32;
                  description
                    "Configure the out-bift-id";
                  }
                            }
```

```
                            case out-bift-id-encoding {
                              leaf out-bift-id-encoding {
                  type boolean;
                  default "false";
                          description
                    "setting this attribute to 'true' will enable
                            calculation of out-bift-id based on <BSL,SD,SI>.";
                    }
                }
              }
            }
          }
                }
          }
     }
        }

  notification bfr-id-collision {
    description
      "This notification is sent when BFR-id received from
      different routers collide.";
    list bfr-id-collision {
      description
        "List of BFR-id that collide.";
           leaf received-bfr-id {
        type uint16;
                description
                  "Value of the BFR-id received.";
       }
        }
   }

   notification bfr-id-out-of-range {
     description
       "This notification is sent when a BFR-id is received
        that is is larger than locally configured (bsl * max-si).
                The notification generation must be throttled with at
                least a 5-second gap between notifications.";
     leaf received-bfr-id {
           type uint16;
           description
                 "Value of the BFR-id received.";
         }
   }

   notification bfr-zero {
     description
           "This notification is sent when an invalid value
```

```
            associated with prefix.";
         leaf ipv4-bfr-prefix {
           type inet:ipv4-prefix;
       description
         "BIER ipv4 bfr prefix";
      }
     leaf ipv6-bfr-prefix{
       type inet:ipv6-prefix;
       description
         "BIER ipv6 bfr prefix";
      }
   }

    notification sub-domain-id-collision {
     description
           "This notification is sent when sub-domain-id received from
        different routers collide.";
         leaf received-sub-domain-id {
           type uint16;
           description
                 "Value of the sub-domain-id received.";
         }
     leaf received-mt-id{
       type uint16;
       description
         "Value of the multi-topology ID received.;
     }
    }
  }
  <code end>
```

8. IANA Considerations

This document registers a URI in the "IETF XML Registry" [RFC3688]. Following the format in [RFC3688], the following registration is requested to be made:

```
    ID:  yang:ietf-bier
    URI:  urn:ietf:params:xml:ns:yang:ietf-bier
    Registrant Contact:  The IESG.
    XML:  N/A, the requested URI is an XML namespace.
```

This document registers YANG modules in the "YANG Module Names"registry [RFC6020].

```
Name:  ietf-bier
Maintained by IANA:  N
Namespace:  urn:ietf:params:xml:ns:yang:ietf-bier
Prefix:  bier
Reference:  RFC ****
```

9. Security Considerations

TBD.

10. Acknowledgements

TBD.

11. Normative References

[I-D.ietf-bier-ospfv3-extensions]
Psenak, P., Nainar, N. K., and I. Wijnands, "OSPFv3 Extensions for BIER", Work in Progress, Internet-Draft, draft-ietf-bier-ospfv3-extensions-07, 1 December 2022, <https://datatracker.ietf.org/doc/html/draft-ietf-bier-ospfv3-extensions-07>.

[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>.

[RFC3688] Mealling, M., "The IETF XML Registry", BCP 81, RFC 3688, DOI 10.17487/RFC3688, January 2004, <https://www.rfc-editor.org/info/rfc3688>.

[RFC6020] Bjorklund, M., Ed., "YANG - A Data Modeling Language for the Network Configuration Protocol (NETCONF)", RFC 6020, DOI 10.17487/RFC6020, October 2010, <https://www.rfc-editor.org/info/rfc6020>.

[RFC6241] Enns, R., Ed., Bjorklund, M., Ed., Schoenwaelder, J., Ed., and A. Bierman, Ed., "Network Configuration Protocol (NETCONF)", RFC 6241, DOI 10.17487/RFC6241, June 2011, <https://www.rfc-editor.org/info/rfc6241>.

[RFC6991] Schoenwaelder, J., Ed., "Common YANG Data Types", RFC 6991, DOI 10.17487/RFC6991, July 2013, <https://www.rfc-editor.org/info/rfc6991>.

[RFC7950] Bjorklund, M., Ed., "The YANG 1.1 Data Modeling Language", RFC 7950, DOI 10.17487/RFC7950, August 2016, <https://www.rfc-editor.org/info/rfc7950>.

[RFC8040] Bierman, A., Bjorklund, M., and K. Watsen, "RESTCONF Protocol", RFC 8040, DOI 10.17487/RFC8040, January 2017, <https://www.rfc-editor.org/info/rfc8040>.

[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>.

[RFC8279] Wijnands, IJ., "Multicast Using Bit Index Explicit Replication (BIER)", RFC 8279, DOI 10.17487/RFC8279, November 2017, <https://www.rfc-editor.org/info/rfc8279>.

[RFC8340] Bjorklund, M., "YANG Tree Diagrams", RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8343] Bjorklund, M., "A YANG Data Model for Interface Management", RFC 8343, DOI 10.17487/RFC8343, March 2018, <https://www.rfc-editor.org/info/rfc8343>.

[RFC8349] Lhotka, L., Lindem, A., and Y. Qu, "A YANG Data Model for Routing Management (NMDA Version)", RFC 8349, DOI 10.17487/RFC8349, March 2018, <https://www.rfc-editor.org/info/rfc8349>.

[RFC8401] Ginsberg, L., Przygienda, T., Aldrin, S., and J. Zhang, "Bit Index Explicit Replication (BIER) Support via IS-IS", RFC 8401, DOI 10.17487/RFC8401, June 2018, <https://www.rfc-editor.org/info/rfc8401>.

[RFC8444] Psenak, P., Wijnands, IJ., Dolganow, A., Przygienda, T., Zhang, J., and S. Aldrin, "OSPFv2 Extensions for Bit Index Explicit Replication (BIER)", RFC 8441, DOI 10.17487/RFC8441, November 2018, <https://www.rfc-editor.org/info/rfc8444>.

Authors' Addresses

Ran Chen
ZTE Corporation
Nanjing
China
Email: chen.ran@zte.com.cn

Fangwei Hu
Individual
Shanghai
China
Email: hufwei@gmail.com

Zheng Zhang
ZTE Corporation
Nanjing
China
Email: zhang.zheng@zte.com.cn

Xianxian Dai
ZTE Corporation
Nanjing
China
Email: Dai.xianxian@zte.com.cn

Mahesh Sivakumar
Juniper networks
Sunnyvale, CALIFORNIA 94089,
United States
Email: sivakumar.mahesh@gmail.com