Stream: Internet Engineering Task Force (IETF)
RFC: 9985
Category: Experimental
Published: June 2026
ISSN: 2070-1721
Authors:
M. Jethanandani *Arrcus*
A. Mishra *Aalyria Technologies*
J. Haas *HPE*
A. Saxena *Ciena Corporation*
M. Bhatia *Google*

RFC 9985
Optimizing Bidirectional Forwarding Detection (BFD) Authentication

Abstract

This document describes an experimental optimization to Bidirectional Forwarding Detection (BFD) Authentication. This optimization enables BFD to scale better when there is a desire to use authentication where applying the same authentication mechanism to every BFD Control Packet may adversely impact performance. This optimization partitions BFD Authentication into a more computationally intensive (MCI) mechanism that is applied to BFD significant changes and a less computationally intensive (LCI) mechanism that is applied to the majority of BFD Control Packets.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

BFD [RFC5880] authentication procedures, when enabled, authenticate each control packet using the same authentication mechanism. Devices implementing BFD are often resource-constrained and authentication may adversely impact the performance of BFD, thus discouraging the deployment of authentication.

When implemented in software, BFD Authentication mechanisms compete with other necessary work done by the systems implementing the protocol. When implemented using hardware acceleration, these mechanisms may scale better situationally, but they still impose a cost on the implementation. BFD's value is tied to its ability to scale in terms of numbers of sessions and a Detection Time that relies on sending its control packets at a high rate. Implementers and operators are forced to evaluate trade-offs of the benefits of authentication vs. its impact on BFD performance.

The authentication mechanisms documented in [RFC5880], MD5 Message-Digest Algorithm [RFC1321], and Secure Hash Algorithm (SHA-1) [RFC3174] are not particularly strong in a cryptographic sense. However, they may still not appropriately scale situationally in a given implementation. In the future, there may be a desire to use stronger authentication mechanisms than those already specified, and those mechanisms are likely to use even more resources.

The BFD protocol can broadly be described as the set of procedures that handle its state machine changes to reach the Up state, and once BFD is in the Up state, it will send those Up packets at the negotiated high rate. The number of BFD Control Packets needed to signal state changes (called significant changes) is very small, while the majority of the Control Packets validate that the session remains in the Up state.

This document describes an experimental optimization to BFD Authentication. This optimization partitions BFD Authentication into a more computationally intensive (MCI) mechanism used to authenticate significant changes, and a less computationally intensive (LCI) mechanism applied to the majority of the BFD Control Packets that don't signal such significant changes.

The details of the motivation for experimental status are given in Appendix B.

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 used in this document have been defined in BFD [RFC5880].

- Auth Type
- Detect Multiplier
- Detection Time

The following terms are introduced in this document.

significant change: A state change, demand mode change (to D bit), or poll sequence change (P or F bit). Changes to BFD Control Packets that do not require a poll sequence, such as bfd.DetectMult, are also considered a significant change.

More Computationally Intensive (MCI) authentication: The authentication mechanism applied to BFD Control Packets that are significant changes.

Less Computationally Intensive (LCI) authentication: The authentication mechanism applied to BFD Control Packets that are NOT significant changes.

configured MCI reauthentication interval: Interval at which BFD Control Packets are retried using MCI authentication.

The authentication mechanisms described in this optimization are paired as MCI and LCI. While it will be generally the case that the relationship between these mechanisms will be "stronger" and "less strong", this document doesn't use the term "strong" to avoid conflation with either mechanism's relative cryptographic strength. The relative criteria for each mechanism is the impact on the implementation.

3. BFD Control Packets That Require MCI Authentication

The intention of these optimized procedures is to permit more computationally intensive authentication for BFD state changes and utilize the less computationally intensive authentication mechanisms to provide protection for the session in the Up state while performing less work overall. Such procedures are intended to aid BFD session scaling without compromising BFD session security.

All BFD Control Packets with the state AdminDown, Down, and Init **MUST** use MCI authentication.

Once the BFD state machine has reached the Up state, it will continue to send BFD Control Packets with MCI authentication in the Up state for a period as discussed in Section 7.2. If optimized authentication mechanisms are in use, as defined in Section 6, the session **MAY** switch to the LCI mode.

The contents of an Up packet must not change aside from the Authentication Section unless MCI authentication is in use.

3.1. Protecting BFD Significant Changes with MCI Authentication

This document proposes that BFD Control Packets that signal a state change, a change in demand mode (D bit), or a poll sequence (P or F bit change) be categorized as a "significant change". Control packets that do not require a poll sequence, such as bfd.DetectMult, are also considered a significant change.

Such significant changes are intended to be protected by more computationally intensive authentication.

4. Using LCI Auth Types

The majority of packets exchanged in a BFD session in the Up state are not significant changes. This document proposes a new optimized authentication mode where packets that are not significant changes may use an LCI authentication mechanism.

Once the session has reached the Up state, the session can use an LCI Auth Type derived from the format in Section 7. Currently, this includes:

- Meticulous Keyed ISAAC Authentication as described in [RFC9986]. This authentication type protects the BFD session when BFD Up packets do not change, because only the paired devices know the shared secret, key, and sequence number to select the ISAAC result.

Other mechanisms may be defined in the future.

5. Periodic MCI Reauthentication

When using the LCI authentication mechanism, BFD should periodically test the session using the MCI authentication mechanism. MCI authentication is tested using a Poll sequence. To test MCI authentication, a Poll sequence **SHOULD** be initiated by the sender using the MCI authentication mode rather than the LCI mechanism. If a control packet with the Final (F) bit is not received using MCI authentication within twice the Detect Interval as would be calculated by the receiving system, the session has been compromised, and it **MUST** be brought down.

The value "twice the Detect interval as would be calculated by the receiving system" is, roughly, twice the number of packets the local system would transmit to the receiving system within its own Detect Interval. This accommodates for possible packet loss from the sending system during the Poll sequence to the receiving system, plus time for the receiving system to transmit a control packet with the Final (F) bit set to the local system.

This "MCI reauthentication interval" for performing such periodic tests using the MCI authentication mechanism can be configured depending on the capability of the system.

Most packets transmitted in a BFD session are BFD Up packets. MCI authenticating a limited subset of these packets with a Poll sequence as described above, e.g., every one minute, significantly reduces the computational demand for the system while maintaining security of the session across the configured MCI reauthentication interval.

6. Optimized Authentication Modes

The cryptographic authentication mechanisms specified in Section 6.7 of BFD [RFC5880] describe enabling and disabling of authentication as a one-time operation. The following is stated in Section 6.7.1 of [RFC5880]:

> ... implementations using this method **SHOULD** only allow the authentication state to be changed at most once without some form of intervention (so that authentication cannot be turned on and off repeatedly simply based on the receipt of BFD Control Packets from remote systems).

Once enabled, every packet must have the Authentication Present (A) bit set and the associated Authentication Type appended (Section 4.1 of [RFC5880]). In addition, Section 6.7.1 of [RFC5880] states that an implementation **SHOULD NOT** allow the authentication state to be changed based on the receipt of a BFD Control Packet.

This document proposes that an "optimized" authentication mode that permits both an MCI authentication mode and an LCI mode be used within the same BFD session. This pairing of an MCI and an LCI mode of authentication is carried in new BFD Authentication types representing a given optimized authentication type pairing.

This document defines which BFD Control Packets require MCI authentication in Section 3.1. A BFD Control Packet that fails authentication, or a BFD Control Packet that was supposed to be MCI-authenticated but was not (e.g., a significant change packet), is discarded. However, there is no change to the state machine for BFD, as the decision of a significant change is still decided by how many valid consecutive packets were received.

In this specification, the contents of an Up packet **MUST NOT** change aside from the Authentication Section without MCI authentication. The full procedure is documented in the following sections.

7. Signaling Optimized Authentication

When the Authentication Present (A) bit is set and the Auth Type ([RFC5880], Section 4.1) is a type supporting Optimized BFD Authentication, the Auth Type signals a pairing of an MCI authentication type and an LCI authentication type. This pairing is advertised in a single Auth Type value in order to permit implementations to be aware that:

- Optimized BFD procedures will be in use.
- The pairing of the MCI and LCI authentication mechanisms will be used for that session.
- There is a requirement to carry a Sequence Number.
- The current MCI or LCI mode will be carried as described below.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Auth Type   |   Auth Len    |  Auth Key ID  |   Opt. Mode   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                   Authentication Specific Data                ~
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Common Optimized BFD Authentication Section

The values of Auth Type and Auth Len are defined in their respective Optimized BFD Authentication procedural documents.

The values of the Optimized Authentication Mode field are:

1. The MCI authentication type for Optimized BFD Auth Types.
2. The LCI authentication type for Optimized BFD Auth Types.

Authentication Specific Data: When using the more computationally intensive authentication type, the remainder of the Authentication Section carries that type's data.

7.1. Transmitting and Receiving Using Optimized Authentication

The procedures for authenticating BFD Control Packets using Optimized Authentication are similar to the existing procedures covered in Section 6.7 of [RFC5880]. Optimized Authentication modes have common procedural requirements for authentication regardless of which more or less computationally intensive authentication modes are used.

The required value of the Auth Len field for a given Optimized Authentication mode is defined in the respective specifications for their respective MCI and LCI modes.

The following common procedures apply to authenticating BFD Control packets utilizing Optimized Authentication:

- If the received BFD Control Packet does not contain an Authentication Section ([RFC5880], Section 4.1), or the Auth Type is not a supported Optimized Authentication Auth Type, then the received packet **MUST** be discarded.
- If the received BFD Control Packet contains an optimized authentication type using these procedures and the Optimized Authentication Mode field is not 1 or 2, then the received packet **MUST** be discarded.
- If bfd.SessionState is AdminDown, Down, or Init and the Optimized Authentication Mode field is not 1, then the received packet **MUST** be discarded.
- If bfd.SessionState is Up and there is a significant change as defined in Section 3.1, and the Optimized Authentication Mode field is not 1, then the received packet **MUST** be discarded.
- If the Auth Len field is not equal to a value appropriate for the Optimized Authentication Mode field, the packet **MUST** be discarded.
- If bfd.AuthSeqKnown is 1, examine the Sequence Number field. If the sequence number lies outside of the inclusive range of bfd.RcvAuthSeq+1 to bfd.RcvAuthSeq+(3*Detect Mult) when treated as an unsigned 32-bit circular number space, the received packet **MUST** be discarded.

Otherwise (bfd.AuthSeqKnown is 0), bfd.AuthSeqKnown **MUST** be set to 1, bfd.RcvAuthSeq **MUST** be set to the value of the received Sequence Number field, and the received packet **MUST** be accepted.

For the specified Auth Type and Optimized Authentication Mode, perform the appropriate authentication procedures. If authentication succeeds, the received packet **MUST** be accepted. Otherwise, the received packet **MUST** be discarded.

7.2. Optimized Authentication Operations

As noted in Section 3.1, when using Optimized BFD procedures, MCI authentication is used in the BFD state machine to bring a BFD session to the Up state or to make any change of the BFD parameters as carried in the BFD Control Packet when in the Up state.

Once the BFD session has reached the Up state, the BFD Up state **MUST** be signaled to the remote BFD system using the MCI authentication mode for an interval that is at least the Detection Time before switching to the LCI authentication mode. This is to permit mechanisms such as Meticulous Keyed ISAAC for BFD Optimized Authentication [RFC9986] or other approved, less intensive authentication mechanisms to be bootstrapped before switching to the LCI mode.

It is **RECOMMENDED** that when using optimized authentication that implementations switch from MCI authentication to LCI authentication mode after an interval that is at least the Detection Time. In the circumstances where a BFD session successfully reaches the Up state with MCI authentication, but there are problems with the LCI authentication, this will permit the remote system to tear down the session as quickly as possible.

BFD sessions using optimized authentication that succeed in reaching the Up state using MCI authentication and fail using LCI authentication **SHOULD** bring the issue to the attention of the operator. Furthermore, implementations **MAY** wish to throttle session restarts.

It is further **RECOMMENDED** that BFD implementations using optimized authentication defer notifying their client that the session has reached the Up state until it has transitioned to using the LCI authentication mode. In the event where LCI authentication is failing in the protocol, this avoids propagating the failed transitions to the LCI mode to their clients.

8. Optimizing Authentication YANG Data Model

8.1. Data Model Overview

The YANG 1.1 [RFC7950] data model defined in this document augments the "ietf-bfd" module to add data nodes relevant to the management of the feature defined in this document. It adds an interval value that specifies how often the BFD session should be reauthenticated using more computationally intensive authentication once it is in the Up state.

8.2. Tree Diagram

The tree diagram for the YANG modules defined in this document uses annotations defined in YANG Tree Diagrams [RFC8340].

```
module: ietf-bfd-opt-auth

  augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/bfd:bfd/bfd-ip-sh:ip-sh
            /bfd-ip-sh:sessions/bfd-ip-sh:session
            /bfd-ip-sh:authentication:
    +--rw reauth-interval?   uint32
  augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/bfd:bfd/bfd-ip-mh:ip-mh
            /bfd-ip-mh:session-groups/bfd-ip-mh:session-group
            /bfd-ip-mh:authentication:
    +--rw reauth-interval?   uint32
  augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/bfd:bfd/bfd-lag:lag
            /bfd-lag:sessions/bfd-lag:session/bfd-lag:authentication:
    +--rw reauth-interval?   uint32
  augment /rt:routing/rt:control-plane-protocols
            /rt:control-plane-protocol/bfd:bfd/bfd-mpls:mpls
            /bfd-mpls:session-groups/bfd-mpls:session-group
            /bfd-mpls:authentication:
    +--rw reauth-interval?   uint32
```

8.3. The YANG Data Model

This YANG module imports modules defined in "A YANG Data Model for Routing Management (NMDA Version)" [RFC8349] and "YANG Data Model for Bidirectional Forwarding Detection (BFD)" [RFC9314].

Implementations supporting the optimization procedures defined in this document enable optimization by using one of the newly defined key-chain crypto-algorithms in the ietf-bfd-met-keyed-isaac YANG module in [RFC9986].

```
<CODE BEGINS> file "ietf-bfd-opt-auth@2026-06-19.yang"

module ietf-bfd-opt-auth {
  yang-version 1.1;
  namespace "urn:ietf:params:xml:ns:yang:ietf-bfd-opt-auth";
  prefix bfd-oa;

  import ietf-routing {
    prefix rt;
    reference
      "RFC 8349: A YANG Data Model for Routing Management
       (NMDA version).";
  }

  import ietf-bfd {
    prefix bfd;
    reference
      "RFC 9314: YANG Data Model for Bidirectional
       Forwarding Detection (BFD).";
  }

  import ietf-bfd-ip-sh {
    prefix bfd-ip-sh;
    reference
      "RFC 9314: YANG Data Model for Bidirectional
       Forwarding Detection (BFD).";
  }

  import ietf-bfd-ip-mh {
    prefix bfd-ip-mh;
    reference
      "RFC 9314: YANG Data Model for Bidirectional
       Forwarding Detection (BFD).";
  }

  import ietf-bfd-lag {
    prefix bfd-lag;
    reference
      "RFC 9314: YANG Data Model for Bidirectional
       Forwarding Detection (BFD).";
  }

  import ietf-bfd-mpls {
    prefix bfd-mpls;
    reference
      "RFC 9314: YANG Data Model for Bidirectional
       Forwarding Detection (BFD).";
  }

  organization
    "IETF Bidirectional Forwarding Detection (BFD) Working Group";

  contact
```

```
    "WG Web:   <http://tools.ietf.org/wg/bfd>
     WG List:  <rtg-bfd@ietf.org>

     Authors: Mahesh Jethanandani (mjethanandani@gmail.com)
              Ashesh Mishra (ashesh@aalyria.com)
              Ankur Saxena (ankurpsaxena@gmail.com)
              Manav Bhatia (mnvbhatia@google.com)
              Jeffrey Haas (jeffrey.haas@hpe.com).";

  description
    "This YANG module augments the base BFD YANG module to add
     attributes related to the experimental BFD Optimized
     Authentication.

     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) 2026 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 9985
     (https://www.rfc-editor.org/info/rfc9985); see the RFC itself
     for full legal notices.";

  revision "2026-06-19" {
    description
      "Initial Version.";
    reference
      "RFC 9985: Optimizing BFD Authentication.";
  }

  feature optimized-auth {
    description
      "Indicates that the implementation supports optimized
       authentication.";
    reference
      "RFC 9985: Optimizing BFD Authentication.";
  }

  grouping bfd-opt-auth-config {
    description
      "Grouping for BFD Optimized Authentication Parameters.";
    leaf reauth-interval {
      type uint32;
      units "seconds";
      default "60";
      description
        "Interval of time after which more computationally intensive
```

```
          authentication should be utilized to prevent an
          on-path-attacker attack.

          A value of zero means that we do not do periodic
          reauthentication using the more computationally intensive
          authentication method.

          This value SHOULD have jitter applied to it to avoid
          self-synchronization during expensive authentication
          operations.";
    }
  }

  augment "/rt:routing/rt:control-plane-protocols"
        + "/rt:control-plane-protocol/bfd:bfd/bfd-ip-sh:ip-sh"
        + "/bfd-ip-sh:sessions/bfd-ip-sh:session"
        + "/bfd-ip-sh:authentication" {
    uses bfd-opt-auth-config;

    description
      "Augment the 'authentication' container for single-hop BFD
       module to add attributes related to BFD Optimized
       Authentication.";
  }

  augment "/rt:routing/rt:control-plane-protocols"
        + "/rt:control-plane-protocol/bfd:bfd/bfd-ip-mh:ip-mh"
        + "/bfd-ip-mh:session-groups/bfd-ip-mh:session-group"
        + "/bfd-ip-mh:authentication" {
    uses bfd-opt-auth-config;

    description
      "Augment the 'authentication' container for multi-hop BFD
       module to add attributes related to BFD Optimized
       Authentication.";
  }

  augment "/rt:routing/rt:control-plane-protocols"
        + "/rt:control-plane-protocol/bfd:bfd/bfd-lag:lag"
        + "/bfd-lag:sessions/bfd-lag:session"
        + "/bfd-lag:authentication" {
    uses bfd-opt-auth-config;

    description
      "Augment the 'authentication' container for BFD over LAG
       module to add attributes related to BFD Optimized
       Authentication.";
  }

  augment "/rt:routing/rt:control-plane-protocols"
        + "/rt:control-plane-protocol/bfd:bfd/bfd-mpls:mpls"
        + "/bfd-mpls:session-groups/bfd-mpls:session-group"
        + "/bfd-mpls:authentication" {
    uses bfd-opt-auth-config;

    description
      "Augment the 'authentication' container for BFD over MPLS
       module to add attributes related to BFD Optimized
```

```
        Authentication.";
  }
}
<CODE ENDS>
```

9. IANA Considerations

IANA has assigned one URI and one YANG module as described in this section.

9.1. IETF XML Registry

IANA has registered the following URI in the "ns" registry within the "IETF XML Registry" group [RFC3688]:

URI: urn:ietf:params:xml:ns:yang:ietf-bfd-opt-auth
Registrant Contact: The IESG
XML: N/A; the requested URI is an XML namespace.

9.2. The YANG Module Names Registry

IANA has registered the following YANG module in the "YANG Module Names" registry [RFC6020] within the "YANG Parameters" registry group:

Name: ietf-bfd-opt-auth
Maintained by IANA: N
Namespace: urn:ietf:params:xml:ns:yang:ietf-bfd-opt-auth
Prefix: bfd-oa
Reference: RFC 9985

10. Security Considerations

10.1. Protocol Security Considerations

Devices implementing BFD are often resource-constrained, whether in a single session or a multidimensional set of scaled sessions. Desired detection intervals for the BFD sessions, and their number, are common scaling considerations for BFD implementations. Security mechanisms also impact the performance of implementations, whether in software or hardware, due to the use of additional computational resources these mechanisms use.

The optimized procedures in this document provide a different level of resistance to attack than methods using a single authentication mechanism:

- The MCI authentication mechanisms used for optimized authentication are expected to have similar cryptographic strength acceptable for BFD for authenticating the entire session, as described in [RFC5880].

- When the BFD state machine is attempting to move from the Down state to the Up state, the MCI authentication mechanism is intended to protect vs. attempt to inappropriately start BFD sessions.
- When the BFD state machine is in the Up state, the MCI authentication mechanism is intended to protect vs. attempt to change BFD session parameters or to reset the BFD session.
- When the BFD state machine is in the Up state, the LCI authentication mechanism is intended to provide resistance to keeping a BFD session in the Up state inappropriately. Since the procedures for changing BFD state require utilizing the MCI mechanism, and the LCI mechanism requires that the contents of the Control Packet in the Up state remain unchanged, the only thing that successfully spoofing such packets can do is keep the session Up.
- The periodic, MCI reauthentication procedure provides protection against long-term successful spoofing of the LCI authentication mechanism.

In other words, the intention of Optimized BFD procedures is to make it difficult to reset or inappropriately start BFD sessions. However, protecting against keeping the session Up is seen as a less interesting attack and can receive less protection.

The recent escalating series of attacks on MD5 and SHA-1 described in Finding Collisions in the Full SHA-1 [SHA-1-attack1] and New Collision Search for SHA-1 [SHA-1-attack2] raise concerns about their remaining useful lifetime as outlined in Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithm [RFC6151] and Security Considerations for the SHA-0 and SHA-1 Message-Digest Algorithm [RFC6194]. If replaced by stronger algorithms, the computational overhead will make the task of authenticating every packet even more difficult to achieve.

The procedures described in this document provide a mechanism that could enable implementations to leverage stronger security to address the concerns above when strong authentication is required. However, this requires operators to evaluate the trade-offs of the less computationally intensive mechanisms to adequately address their desired security stance.

Keys generated and distributed out of band for the purposes described in this specification are generally limited in the security they can provide. It is essential that these keys are selected well and protected when stored.

10.2. YANG Security Considerations

This section is modeled after the template described in Section 3.7.1 of [RFC9907].

The "ietf-bfd-opt-auth" YANG module defines a data model that is designed to be accessed via YANG-based management protocols, such as the Network Configuration Protocol (NETCONF) [RFC6241] and RESTCONF [RFC8040]. These YANG-based management protocols (1) have to use a secure transport layer (e.g., Secure Shell (SSH) [RFC4252], TLS [RFC8446], and QUIC [RFC9000]) and (2) have to use mutual authentication.

The Network Configuration Access Control Model (NACM) [RFC8341] provides the means to restrict access for particular NETCONF or RESTCONF users to a preconfigured subset of all available NETCONF or RESTCONF protocol operations and content.

There are a number of data nodes defined in this YANG module that are writable/creatable/deletable (i.e., "config true", which is the default). All writable data nodes are likely to be sensitive or vulnerable in some network environments. Write operations (e.g., edit-config) and delete operations to these data nodes without proper protection or authentication can have a negative effect on network operations. The following subtrees and data nodes have particular sensitivities/vulnerabilities:

- 'reauth-interval' specifies the interval in Up state, after which MCI authentication **SHOULD** be performed to prevent a Person-in-the-Middle (PITM) attack. If this interval is set very low, the utility of these optimization procedures is lessened. If this interval is set very high, attacks detected by the MCI authentication mechanisms may happen overly late.

There are no particularly sensitive readable data nodes.

There are no RPC operations defined in this model.

11. References

11.1. 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>.

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

[RFC5880] Katz, D. and D. Ward, "Bidirectional Forwarding Detection (BFD)", RFC 5880, DOI 10.17487/RFC5880, June 2010, <https://www.rfc-editor.org/info/rfc5880>.

[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>.

[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>.

[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>.

[RFC8341] Bierman, A. and M. Bjorklund, "Network Configuration Access Control Model", STD 91, RFC 8341, DOI 10.17487/RFC8341, March 2018, <https://www.rfc-editor.org/info/rfc8341>.

[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>.

[RFC9314] Jethanandani, M., Ed., Rahman, R., Ed., Zheng, L., Ed., Pallagatti, S., and G. Mirsky, "YANG Data Model for Bidirectional Forwarding Detection (BFD)", RFC 9314, DOI 10.17487/RFC9314, September 2022, <https://www.rfc-editor.org/info/rfc9314>.

11.2. Informative References

[RFC1321] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, DOI 10.17487/RFC1321, April 1992, <https://www.rfc-editor.org/info/rfc1321>.

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, DOI 10.17487/RFC2026, October 1996, <https://www.rfc-editor.org/info/rfc2026>.

[RFC3174] Eastlake 3rd, D. and P. Jones, "US Secure Hash Algorithm 1 (SHA1)", RFC 3174, DOI 10.17487/RFC3174, September 2001, <https://www.rfc-editor.org/info/rfc3174>.

[RFC4252] Ylonen, T. and C. Lonvick, Ed., "The Secure Shell (SSH) Authentication Protocol", RFC 4252, DOI 10.17487/RFC4252, January 2006, <https://www.rfc-editor.org/info/rfc4252>.

[RFC6151] Turner, S. and L. Chen, "Updated Security Considerations for the MD5 Message-Digest and the HMAC-MD5 Algorithms", RFC 6151, DOI 10.17487/RFC6151, March 2011, <https://www.rfc-editor.org/info/rfc6151>.

[RFC6194] Polk, T., Chen, L., Turner, S., and P. Hoffman, "Security Considerations for the SHA-0 and SHA-1 Message-Digest Algorithms", RFC 6194, DOI 10.17487/RFC6194, March 2011, <https://www.rfc-editor.org/info/rfc6194>.

[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>.

[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>.

[RFC8340] Bjorklund, M. and L. Berger, Ed., "YANG Tree Diagrams", BCP 215, RFC 8340, DOI 10.17487/RFC8340, March 2018, <https://www.rfc-editor.org/info/rfc8340>.

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

[RFC8792] Watsen, K., Auerswald, E., Farrel, A., and Q. Wu, "Handling Long Lines in Content of Internet-Drafts and RFCs", RFC 8792, DOI 10.17487/RFC8792, June 2020, <https://www.rfc-editor.org/info/rfc8792>.

[RFC9000] Iyengar, J., Ed. and M. Thomson, Ed., "QUIC: A UDP-Based Multiplexed and Secure Transport", RFC 9000, DOI 10.17487/RFC9000, May 2021, <https://www.rfc-editor.org/info/rfc9000>.

[RFC9907] Bierman, A., Boucadair, M., Ed., and Q. Wu, "Guidelines for Authors and Reviewers of Documents Containing YANG Data Models", BCP 216, RFC 9907, DOI 10.17487/RFC9907, March 2026, <https://www.rfc-editor.org/info/rfc9907>.

[RFC9986] DeKok, A., Jethanandani, M., Agarwal, S., Mishra, A., and J. Haas, "Meticulous Keyed ISAAC for Bidirectional Forwarding Detection (BFD) Optimized Authentication", RFC 9986, June 2026, <https://www.rfc-editor.org/info/rfc9986>.

[SHA-1-attack1] Wang, X., Yin, Y., and H. Yu, "Finding Collisions in the Full SHA-1", Advances in Cryptology - CRYPTO 2005, Lecture Notes in Computer Science, vol. 3621, pp. 17-36, DOI 10.1007/11535218_2, 2005, <https://doi.org/10.1007/11535218_2>.

[SHA-1-attack2] Wang, X., Yao, A., and F. Yao, "Cryptanalysis on SHA-1", 2005, <https://csrc.nist.gov/csrc/media/events/first-cryptographic-hash-workshop/documents/wang_sha1-new-result.pdf>.

Appendix A. Examples

This section tries to show some examples in how the model can be configured.

A.1. Single-Hop BFD Configuration

This example demonstrates how a single-hop BFD session can be configured for optimized authentication. Note that line wrapping is used per [RFC8792].

```
=============== NOTE: '\' line wrapping per RFC 8792 ===============

<?xml version="1.0" encoding="UTF-8"?>
<key-chains
    xmlns="urn:ietf:params:xml:ns:yang:ietf-key-chain"
    xmlns:opt-auth="urn:ietf:params:xml:ns:yang:ietf-bfd-opt-auth"
    xmlns:bfd-mki="urn:ietf:params:xml:ns:yang:ietf-bfd-met-keyed-i\
saac">
  <key-chain>
    <name>bfd-auth-config</name>
    <description>"An example for BFD Optimized Auth configuration."\
</description>
    <key>
      <key-id>55</key-id>
      <lifetime>
        <send-lifetime>
          <start-date-time>2017-01-01T00:00:00Z</start-date-time>
          <end-date-time>2017-02-01T00:00:00Z</end-date-time>
        </send-lifetime>
        <accept-lifetime>
          <start-date-time>2016-12-31T23:59:55Z</start-date-time>
          <end-date-time>2017-02-01T00:00:05Z</end-date-time>
        </accept-lifetime>
```

```
      </lifetime>
      <crypto-algorithm>bfd-mki:optimized-sha1-meticulous-keyed-isa\
ac</crypto-algorithm>
      <key-string>
        <keystring>testvector</keystring>
      </key-string>
    </key>
  </key-chain>
</key-chains>
<interfaces
    xmlns="urn:ietf:params:xml:ns:yang:ietf-interfaces"
    xmlns:if-type="urn:ietf:params:xml:ns:yang:iana-if-type">
  <interface>
    <name>eth0</name>
    <type>if-type:ethernetCsmacd</type>
  </interface>
</interfaces>
<routing
    xmlns="urn:ietf:params:xml:ns:yang:ietf-routing"
    xmlns:bfd-types="urn:ietf:params:xml:ns:yang:ietf-bfd-types"
    xmlns:iana-bfd-types="urn:ietf:params:xml:ns:yang:iana-bfd-type\
s"
    xmlns:opt-auth="urn:ietf:params:xml:ns:yang:ietf-bfd-opt-auth"
    xmlns:bfd-mki="urn:ietf:params:xml:ns:yang:ietf-bfd-met-keyed-i\
saac">
  <control-plane-protocols>
    <control-plane-protocol>
      <type>bfd-types:bfdv1</type>
      <name>name:BFD</name>
      <bfd xmlns="urn:ietf:params:xml:ns:yang:ietf-bfd">
        <ip-sh xmlns="urn:ietf:params:xml:ns:yang:ietf-bfd-ip-sh">
          <sessions>
            <session>
              <interface>eth0</interface>
              <dest-addr>2001:db8:0:113::101</dest-addr>
              <desired-min-tx-interval>10000</desired-min-tx-interv\
al>
              <required-min-rx-interval>
                10000
              </required-min-rx-interval>
              <authentication>
                <key-chain>bfd-auth-config</key-chain>
                <opt-auth:reauth-interval>30</opt-auth:reauth-inter\
val>
              </authentication>
            </session>
          </sessions>
        </ip-sh>
      </bfd>
    </control-plane-protocol>
  </control-plane-protocols>
</routing>
```

Appendix B. Experimental Status

This document describes an experiment that presents a candidate solution to update BFD Authentication that is currently specified in [RFC5880]. This experiment is intended to provide additional insights into what happens when the optimized authentication mechanism defined in this document is used. Here are the reasons why this document is on the Experimental track:

- In the initial stages of the document, there were significant participation and reviews from the working group. Since then, there have been considerable changes to the document, such as the use of ISAAC, the allowance for ISAAC bootstrapping when a BFD session comes up, and the use of a single Auth Type to indicate optimized authentication. These changes did not get significant review from the working group and therefore do not meet the bar set in Section 4.1.1 of [RFC2026].
- There are no known implementations at this time.
- The work in this document could become very valuable in the future, especially if the need for deploying BFD Authentication at scale becomes a reality.

This document is classified as Experimental and is not part of the IETF Standards Track. Implementations based on this document should not be considered as compliant with BFD [RFC5880].

Acknowledgments

The authors would like to thank Qiufang Ma, Stephen Farrell, and Acee Lindem for providing directorate reviews of this document.

Contributors

The authors of this document would like to acknowledge Reshad Rahman as a contributor to this document.

Authors' Addresses

Mahesh Jethanandani
Arrcus
United States of America
Email: mjethanandani@gmail.com

Ashesh Mishra
Aalyria Technologies
Email: ashesh@aalyria.com

Jeffrey Haas
HPE
Email: jeffrey.haas@hpe.com

Ankur Saxena
Ciena Corporation
3939 N 1st Street
San Jose, CA 95134
United States of America
Email: ankurpsaxena@gmail.com

Manav Bhatia
Google
Bagmane Capital Park
Bengaluru 560048
Karnataka
India
Email: mnvbhatia@google.com