HMAC-SHA-2 Authentication Protocols in USM for SNMPv3

draft-ietf-opsawg-hmac-sha-2-usm-snmp-06

Abstract

This memo specifies new HMAC-SHA-2 authentication protocols for the User-based Security Model (USM) for SNMPv3 defined in RFC 3414.

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 http://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 October 22, 2015.

Copyright Notice

Copyright (c) 2015 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 (http://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 Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.
1. Introduction

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols. In particular, it defines additional authentication protocols for the User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3) specified in RFC 3414 [RFC3414].

In RFC 3414, two different authentication protocols, HMAC-MD5-96 and HMAC-SHA-96, are defined based on the hash functions MD5 and SHA-1, respectively. This memo specifies new HMAC-SHA-2 authentication protocols for USM using an HMAC based on the SHA-2 family of hash functions [SHA] and truncated to 128 bits for SHA-224, to 192 bits for SHA-256, to 256 bits for SHA-384, and to 384 bits for SHA-512. These protocols are straightforward adaptations of the authentication protocols HMAC-MD5-96 and HMAC-SHA-96 to the SHA-2 based HMAC.
2. The Internet-Standard Management Framework

For a detailed overview of the documents that describe the current Internet-Standard Management Framework, please refer to section 7 of RFC 3410 [RFC3410].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. MIB objects are generally accessed through the Simple Network Management Protocol (SNMP). Objects in the MIB are defined using the mechanisms defined in the Structure of Management Information (SMI). This memo specifies a MIB module that is compliant to the SMIV2, which is described in STD 58, RFC 2578 [RFC2578], STD 58, RFC 2579 [RFC2579] and STD 58, RFC 2580 [RFC2580].

3. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [RFC2119].

4. The HMAC-SHA-2 Authentication Protocols

This section describes the HMAC-SHA-2 authentication protocols. They use the SHA-2 hash functions, which are described in FIPS PUB 180-4 [SHA] and RFC 6234 [RFC6234], in HMAC mode described in RFC 2104 [RFC2104] and RFC 6234, truncating the output to 128 bits for SHA-224, 192 bits for SHA-256, 256 bits for SHA-384, and 384 bits for SHA-512. RFC 6234 also provides source code for all the SHA-2 algorithms and HMAC (without truncation). It also includes test harness and standard test vectors for all the defined hash functions and HMAC examples.

The following protocols are defined:

usmHMAC128SHA224AuthProtocol: uses SHA-224 and truncates the output to 128 bits (16 octets);

usmHMAC192SHA256AuthProtocol: uses SHA-256 and truncates the output to 192 bits (24 octets);

usmHMAC256SHA384AuthProtocol: uses SHA-384 and truncates the output to 256 bits (32 octets);

usmHMAC384SHA512AuthProtocol: uses SHA-512 and truncates the output to 384 bits (48 octets).
Implementations conforming to this specification MUST support usmHMAC192SHA256AuthProtocol and SHOULD support usmHMAC384SHA512AuthProtocol. The protocols usmHMAC128SHA224AuthProtocol and usmHMAC256SHA384AuthProtocol are OPTIONAL.

4.1. Deviations from the HMAC-SHA-96 Authentication Protocol

All the HMAC-SHA-2 authentication protocols are straightforward adaptations of the HMAC-MD5-96 and HMAC-SHA-96 authentication protocols. Precisely, they differ from the HMAC-MD5-96 and HMAC-SHA-96 authentication protocols in the following aspects:

- The SHA-2 hash function is used to compute the message digest in the HMAC computation according to RFC 2104 and RFC 6234, as opposed to the MD5 hash function [RFC1321] and SHA-1 hash function [SHA] used in HMAC-MD5-96 and HMAC-SHA-96, respectively. Consequently, the length of the message digest prior to truncation is 224 bits for SHA-224 based protocol, 256 bits for SHA-256 based protocol, 384 bits for SHA-384 based protocol, and 512 bits for SHA-512 based protocol.

- The resulting message digest (output of HMAC) is truncated to
  * 16 octets for usmHMAC128SHA224AuthProtocol
  * 24 octets for usmHMAC192SHA256AuthProtocol
  * 32 octets for usmHMAC256SHA384AuthProtocol
  * 48 octets for usmHMAC384SHA512AuthProtocol

  as opposed to the truncation to 12 octets in HMAC-MD5-96 and HMAC-SHA-96.

- The user’s secret key to be used when calculating a digest MUST be:
  * 28 octets long and derived with SHA-224 for the SHA-224 based protocol usmHMAC128SHA224AuthProtocol
  * 32 octets long and derived with SHA-256 for the SHA-256 based protocol usmHMAC192SHA256AuthProtocol
  * 48 octets long and derived with SHA-384 for the SHA-384 based protocol usmHMAC256SHA384AuthProtocol
4.2. Processing

This section describes the procedures for the HMAC-SHA-2 authentication protocols. The descriptions are based on the definition of services and data elements defined for HMAC-SHA-96 in RFC 3414 with the deviations listed in Section 4.1.

4.2.1. Processing an Outgoing Message

Values of constants M (the length of the secret key in octets) and N (the length of the MAC output in octets) used below, are:

usmHMAC128SHA224AuthProtocol: M=28, N=16;
usmHMAC192SHA256AuthProtocol: M=32, N=24;
usmHMAC256SHA384AuthProtocol: M=48, N=32;
usmHMAC384SHA512AuthProtocol: M=64, N=48.

This section describes the procedure followed by an SNMP engine whenever it must authenticate an outgoing message using one of the authentication protocols defined above.

1. The msgAuthenticationParameters field is set to serialization, according to the rules in RFC 3417 [RFC3417], of an OCTET STRING containing N zero octets.

2. From the secret authKey of M octets, calculate the HMAC-SHA-2 digest over it according to RFC 6234. Take the first N octets of the final digest - this is the Message Authentication Code (MAC).

3. Replace the msgAuthenticationParameters field with the MAC obtained in the previous step.

4. The authenticatedWholeMsg is then returned to the caller together with statusInformation indicating success.
4.2.2. Processing an Incoming Message

Values of the constants M and N are the same as in Section 4.2.1, and are selected based on which authentication protocol is configured for the given USM usmUser Table entry.

This section describes the procedure followed by an SNMP engine whenever it must authenticate an incoming message using one of the HMAC-SHA-2 authentication protocols.

1. If the digest received in the msgAuthenticationParameters field is not N octets long, then an failure and an errorIndication (authenticationError) is returned to the calling module.

2. The MAC received in the msgAuthenticationParameters field is saved.

3. The digest in the msgAuthenticationParameters field is replaced by the N zero octets.

4. Using the secret authKey, the HMAC is calculated over the wholeMsg.

5. N first octets of the above HMAC are taken as the computed MAC value.

6. The msgAuthenticationParameters field is replaced with the MAC value that was saved in step 2.

7. The newly calculated MAC is compared with the MAC saved in step 2. If they do not match, then a failure and an errorIndication (authenticationFailure) are returned to the calling module.

8. The authenticatedWholeMsg and statusInformation indicating success are then returned to the caller.

5. Key Localization and Key Change

For any of the protocols defined in Section 4, key localization and key change SHALL be performed according to RFC 3414 using the SHA-2 hash function applied in the respective protocol.

6. Structure of the MIB Module

The MIB module specified in this memo does not define any managed objects, subtrees, notifications or tables, but only object identities (for authentication protocols) under a subtree of an existing MIB.
7. Relationship to Other MIB Modules

7.1. Relationship to SNMP-USER-BASED-SM-MIB

RFC 3414 specifies the MIB module for the User-based Security Model (USM) for SNMPv3 (SNMP-USER-BASED-SM-MIB), which defines authentication protocols for USM based on the hash functions MD5 and SHA-1, respectively. The following MIB module defines new HMAC-SHA2 authentication protocols for USM based on the SHA-2 hash functions [SHA]. The use of the HMAC-SHA2 authentication protocols requires the usage of the objects defined in the SNMP-USER-BASED-SM-MIB.

7.2. Relationship to SNMP-FRAMEWORK-MIB

RFC 3411 [RFC3411] specifies the SNMP-FRAMEWORK-MIB, which defines a subtree snmpAuthProtocols for SNMP authentication protocols. The following MIB module defines new authentication protocols in the snmpAuthProtocols subtree.

7.3. MIB modules required for IMPORTS

The following MIB module IMPORTS definitions from SNMPv2-SMI [RFC2578] and SNMP-FRAMEWORK-MIB [RFC3411].

8. Definitions

SNMP-USM-HMAC-SHA2-MIB DEFINITIONS ::= BEGIN
IMPORTS
    MODULE-IDENTITY, OBJECT-IDENTITY,
    snmpModules             FROM SNMPv2-SMI          -- [RFC2578]
    snmpAuthProtocols       FROM SNMP-FRAMEWORK-MIB; -- [RFC3411]

snmpUsmHmacSha2MIB MODULE-IDENTITY
    LAST-UPDATED    "201503090000Z"       -- 9th Mar 2015, midnight
    -- RFC Ed.: replace with publication date & remove this line
    ORGANIZATION    "SNMPv3 Working Group"
    CONTACT-INFO    "WG email: OPSAWG@ietf.org
    Subscribe:
        https://www.ietf.org/mailman/listinfo/opsawg
    Editor:    Johannes Merkle
                secunet Security Networks
    postal:    Mergenthaler Allee 77
                D-65760 Eschborn
                Germany
    phone:     +49 20154543091
    email:     johannes.merkle@secunet.com

Merkle & Lochter  Expires October 22, 2015 [Page 7]
Co-Editor: Manfred Lochter
Bundesamt fuer Sicherheit in der
Informationstechnik (BSI)

postal:    Postfach 200363
D-53133 Bonn
Germany

phone:     +49 228 9582 5643
email:     manfred.lochter@bsi.bund.de

DESCRIPTION "Definitions of Object Identities needed
for the use of HMAC-SHA2 by SNMP’s User-based
Security Model.

Copyright (c) 2014 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 Simplified
BSD License set forth in Section 4.c of the IETF Trust’s
Legal Provisions Relating to IETF Documents
(http://trustee.ietf.org/license-info).

REVISION "201503090000Z"       -- 9th Mar 2015, midnight
-- RFC Ed.: replace with publication date & remove this line
DESCRIPTION "Initial version, published as RFC TBD"
-- RFC Ed.: replace TBD with actual RFC number & remove this line

::= { snmpModules nn }        -- nn to be assigned by IANA
-- RFC Ed.: replace nn with actual number assigned by IANA & remove
-- this comment

usmHMAC128SHA224AuthProtocol OBJECT-IDENTITY
STATUS      current
DESCRIPTION "The Authentication Protocol
usmHMAC128SHA224AuthProtocol uses HMAC-SHA-224 and
truncates output to 128 bits."
REFERENCE   "- Krawczyk, H., Bellare, M., and R. Canetti, HMAC:
Keyed-Hashing for Message Authentication, RFC 2104.
- National Institute of Standards and Technology,
Secure Hash Standard (SHS), FIPS PUB 180-4, 2012."
::= { snmpAuthProtocols aa } -- aa to be assigned by IANA
-- RFC Ed.: replace aa with actual number assigned by IANA & remove
-- this comment

usmHMAC192SHA256AuthProtocol OBJECT-IDENTITY
STATUS      current
DESCRIPTION "The Authentication Protocol
usmHMAC192SHA256AuthProtocol uses HMAC-SHA-256 and truncates output to 192 bits.

REFERENCE

::= { snmpAuthProtocols bb }  -- bb to be assigned by IANA
-- RFC Ed.: replace bb with actual number assigned by IANA & remove
-- this comment

usmHMAC256SHA384AuthProtocol OBJECT-IDENTITY
STATUS  current
DESCRIPTION "The Authentication Protocol
usmHMAC256SHA384AuthProtocol uses HMAC-SHA-384 and truncates output to 256 bits."

REFERENCE

::= { snmpAuthProtocols cc }  -- cc to be assigned by IANA
-- RFC Ed.: replace cc with actual number assigned by IANA & remove
-- this comment

usmHMAC384SHA512AuthProtocol OBJECT-IDENTITY
STATUS  current
DESCRIPTION "The Authentication Protocol
usmHMAC384SHA512AuthProtocol uses HMAC-SHA-512 and truncates output to 384 bits."

REFERENCE

::= { snmpAuthProtocols dd }  -- dd to be assigned by IANA
-- RFC Ed.: replace dd with actual number assigned by IANA & remove
-- this comment

END

9. Security Considerations
9.1. Use of the HMAC-SHA-2 authentication protocols in USM

The security considerations of RFC 3414 also apply to the HMAC-SHA-2 authentication protocols defined in this document.

9.2. Cryptographic strength of the authentication protocols

At the time of publication of this document, all of the HMAC-SHA-2 authentication protocols provide a very high level of security. The security of each HMAC-SHA-2 authentication protocol depends on the parameters used in the corresponding HMAC computation, which are the length of the key (if the key has maximum entropy), the size of the hash function’s internal state, and the length of the truncated MAC. For the HMAC-SHA-2 authentication protocols these values are as follows (values are given in bits).

<table>
<thead>
<tr>
<th>Protocol</th>
<th>Key length</th>
<th>Size of internal state</th>
<th>MAC length</th>
</tr>
</thead>
<tbody>
<tr>
<td>usmHMAC128SHA224AuthProtocol</td>
<td>224</td>
<td>256</td>
<td>128</td>
</tr>
<tr>
<td>usmHMAC192SHA256AuthProtocol</td>
<td>256</td>
<td>256</td>
<td>192</td>
</tr>
<tr>
<td>usmHMAC256SHA384AuthProtocol</td>
<td>384</td>
<td>512</td>
<td>256</td>
</tr>
<tr>
<td>usmHMAC384SHA512AuthProtocol</td>
<td>512</td>
<td>512</td>
<td>384</td>
</tr>
</tbody>
</table>

Table 1: HMAC parameters of the HMAC-SHA-2 authentication protocols

The security of the HMAC scales with both the key length and the size of the internal state: longer keys render key guessing attacks more difficult, and a larger internal state decreases the success probability of MAC forgeries based on internal collisions of the hash function.

The role of the truncated output length is more complicated: according to [BCK], there is a trade-off in that "by outputting less bits the attacker has less bits to predict in a MAC forgery but, on the other hand, the attacker also learns less about the output of the compression function from seeing the authentication tags computed by legitimate parties"; thus, truncation weakens the HMAC against forgery by guessing, but at the same time strengthens it against chosen message attacks aiming at MAC forgery based on internal collisions or at key guessing. RFC 2104 and [BCK] allow truncation to any length that is not less than half the size of the internal state.

Further discussion of the security of the HMAC construction is given in RFC 2104.
9.3. Derivation of keys from passwords

If secret keys to be used for HMAC-SHA-2 authentication protocols are derived from passwords, the derivation SHOULD be performed using the password-to-key algorithm from Appendix A.1 of RFC 3414 with MD5 being replaced by the SHA-2 hash function H used in the HMAC-SHA-2 authentication protocol. Specifically, the password is converted into the required secret key by the following steps:

1. forming a string of length 1,048,576 octets by repeating the value of the password as often as necessary, truncating accordingly, and using the resulting string as the input to the hash function H. The resulting digest, termed "digest1", is used in the next step.
2. a second string is formed by concatenating digest1, the SNMP engine’s snmpEngineID value, and digest1. This string is used as input to the hash function H.

9.4. Access to the SNMP-USM-HMAC-SHA2-MIB

The SNMP-USM-HMAC-SHA2-MIB module defines OBJECT IDENTIFIER values for use in other MIB modules. It does not define any objects that can be accessed. As such, the SNMP-USM-HMAC-SHA2-MIB does not, by itself, have any effect on the security of the Internet.

The values defined in this module are expected to be used with the usmUserTable defined in the SNMP-USER-BASED-SM-MIB [RFC3414]. The considerations in Section 11.5 of RFC 3414 should be taken into account.

10. IANA Considerations

IANA is requested to assign an OID for

<table>
<thead>
<tr>
<th>Descriptor</th>
<th>OBJECT IDENTIFIER value</th>
</tr>
</thead>
<tbody>
<tr>
<td>snmpUsmHmacSha2MIB</td>
<td>{ snmpModules nn }</td>
</tr>
</tbody>
</table>

Table 2: OID of MIB

with nn appearing in the MIB module definition in Section 8.

Furthermore, IANA is requested to assign a value in the SnmpAuthProtocols registry for each of the following protocols.
Table 3: Code points assigned to HMAC-SHA-2 authentication protocols

<table>
<thead>
<tr>
<th>Description</th>
<th>Value</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>usmHMAC128SHA224AuthProtocol</td>
<td>aa</td>
<td>RFC YYYY</td>
</tr>
<tr>
<td>usmHMAC192SHA256AuthProtocol</td>
<td>bb</td>
<td>RFC YYYY</td>
</tr>
<tr>
<td>usmHMAC256SHA384AuthProtocol</td>
<td>cc</td>
<td>RFC YYYY</td>
</tr>
<tr>
<td>usmHMAC384SHA512AuthProtocol</td>
<td>dd</td>
<td>RFC YYYY</td>
</tr>
</tbody>
</table>

-- RFC Ed.: replace YYYY with actual RFC number and remove this line

with aa, bb, cc, etc. appearing in the MIB module definition in Section 8.

11. References

11.1. Normative References


11.2. Informative References


Authors’ Addresses

Johannes Merkle (editor)
Secunet Security Networks
Mergenthaler Allee 77
65760 Eschborn
Germany
Phone: +49 201 5454 3091
EMail: johannes.merkle@secunet.com

Manfred Lochter
BSI
Postfach 200363
53133 Bonn
Germany
Phone: +49 228 9582 5643
EMail: manfred.lochter@bsi.bund.de