Network Working Group
Request for Comments: 4822
Obsoletes: 2082
Updates: 2453
Category: Standards Track

R. Atkinson
Extreme Networks
M. Fanto
NIST

February 2007

RIPv2 Cryptographic Authentication

Status of This Memo

Copyright Notice

IESG Note

In the interests of encouraging rapid migration away from Keyed-MD5 and its known weakness, the IESG has approved this document even though it does not meet the guidelines in BCP 107 (RFC 4107). However, the IESG stresses that automated key management should be used to establish session keys and urges that the future work on key management described in Section 5.6 of this document should be performed as soon as possible.

Abstract

This note describes a revision to the RIPv2 Cryptographic Authentication mechanism originally specified in RFC 2082. This document obsoletes RFC 2082 and updates RFC 2453. This document adds details of how the SHA family of hash algorithms can be used with RIPv2 Cryptographic Authentication, whereas the original document only specified the use of Keyed-MD5. Also, this document clarifies a potential issue with an active attack on this mechanism and adds significant text to the Security Considerations section.

1. Introduction

Growth in the Internet has made us aware of the need for improved authentication of routing information. RIPv2 provides for unauthenticated service (as in classical RIP), or password authentication. Both are vulnerable to passive attacks currently widespread in the Internet. Well-understood security issues exist in routing protocols [Bell89]. Cleartext passwords, originally specified for use with RIPv2, are widely understood to be vulnerable to easily deployed passive attacks [HA94].

The original RIPv2 cryptographic authentication specification, RFC 2082 [AB97], used the Keyed-MD5 cryptographic mechanism. While there are no openly published attacks on that mechanism, some reports [Dobb96a, Dobb96b] create concern about the ultimate strength of the MD5 cryptographic hash function. Further, some end users, particularly several different governments, require the use of the SHA hash function family rather than any other such function for policy reasons. Finally, the original specification uses a hashing construction widely believed to be weaker than the HMAC construction used with the algorithms added in this revision of the specification.

This document obsoletes the original specification, RFC 2082 [AB97]. This specification differs from RFC 2082 by adding support for the SHA family of hash algorithms and the HMAC technique, while retaining the original Keyed-MD5 algorithm and mode. As the original RIPv2 Cryptographic Authentication mechanism was algorithm-independent, backwards compatibility is retained. This requirement for backwards compatibility precludes making significant protocol changes. So, this document limits changes to the addition of support for an additional family of cryptographic algorithms. The original specification has been very widely implemented, is known to be widely interoperable, and is also widely deployed.

The authors do NOT believe that this specification is the final answer to RIPv2 authentication and encourage the reader to consult the Security Considerations section of this document for more details.

If RIPv2 authentication is disabled, then only simple misconfigurations are detected. The original RIPv2 authentication mechanism relied upon reused cleartext passwords. Use of cleartext password authentication can protect against accidental misconfigurations if that were the only concern, but is not helpful from a security perspective. By simply capturing information on the wire -- straightforward even in a remote environment -- a hostile

entity can read the cleartext RIPv2 password and use that knowledge to inject false information into the routing system via the RIPv2 routing protocol.

This mechanism is intended to reduce the risk of a successful passive attack upon RIPv2 deployments. That is, deployment of this mechanism greatly reduces the vulnerability of the RIPv2-based routing system from a passive attack. When cryptographic authentication is enabled, we transmit the output of a keyed cryptographic one-way function in the authentication field of the RIPv2 packet, instead of sending a cleartext reusable password in the RIPv2 packet. The RIPv2 Authentication Key is known only to the authorized parties of the RIPv2 session. The RIPv2 Authentication Key is never sent over the network in the clear.

In this way, protection is afforded against forgery or message modification. While it is possible to replay a message until the sequence number changes, a sequence number can be used to reduce replay risks. The mechanism does not provide confidentiality, since messages stay in the clear. Since the objective of a routing protocol is to advertise the routing topology, confidentiality is not normally required for routing protocols.

Other relevant rationales for the approach are that MD5 and SHA-1 are both being used for other purposes and are therefore generally already present in IP routers, as is some form of password management.

1.1. Terminology

In this document, the words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" are to be interpreted as described in [BCP14] and indicate requirement levels for compliant or conformant implementations.

2. Implementation Approach

Implementation requires use of a special packet format, special authentication procedures, and also management controls. Implementers need to remember that the Security Considerations section is an integral part of this specification and contains important parts of this specification.

2.1. RIPv2 PDU Format

The basic RIPv2 message format provides for an 8-octet header with an array of 20-octet records as its data content. When RIPv2 Cryptographic Authentication is enabled, the same header and content are used as with the original RIPv2 specification, but the 16-octet "Authentication" password field of the original RIPv2 specification is reused to contain a packet offset to the Authentication Data, a Key Identifier, the Authentication Data Length, and a non-decreasing sequence number.

AUTHENTICATION TYPE
: The "Authentication Type" is Cryptographic Hash Function, which is indicated by the value 3.

RIPv2 PACKET LENGTH
: An unsigned 16-bit offset from the start of the RIPv2 header to the end of the regular RIPv2 packet (not including the authentication trailer).

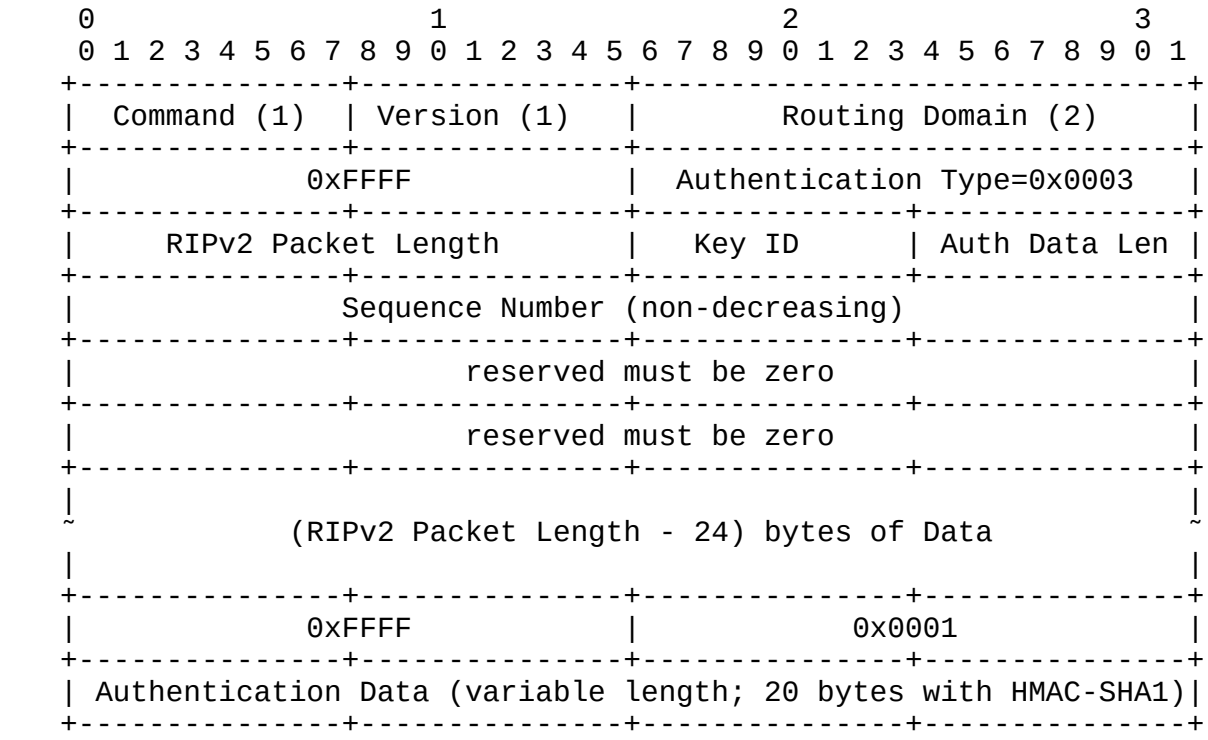
KEY IDENTIFIER
: An unsigned 8-bit field that contains the Key Identifier or Key-ID. This, in combination with the network interface, identifies the RIPv2 Security Association in use for this packet. The RIPv2 Security Association, which is defined in Section 2.2 below, includes the Authentication Key that was used to create the Authentication Data for this RIPv2 message and other parameters. In implementations supporting more than one authentication algorithm, the RIPv2 Security Association also includes information about which authentication algorithm is in use for this message. A RIPv2 Security Association is always associated with an interface, rather than with a router. The actual cryptographic key is part of the RIPv2 Security Association.

AUTHENTICATION DATA LENGTH
: An unsigned 8-bit field that contains the length in octets of the trailing Authentication Data field. The presence of this field helps provide cryptographic algorithm independence.

AUTHENTICATION DATA
: This field contains the cryptographic Authentication Data used to validate this packet. The length of this field is stored in the AUTHENTICATION DATA LENGTH field above.

SEQUENCE NUMBER
An unsigned 32-bit sequence number. The sequence number MUST be non-decreasing for all messages sent from a given source router with a given Key ID value.

The authentication trailer contains the Authentication Data, which is the output of the keyed cryptographic hash function. See later subsections of this section for details on computing this field.



2.2. RIPv2 Security Association

Understanding the RIPv2 Security Association concept is central to understanding this specification. A RIPv2 Security Association contains the set of shared authentication configuration parameters needed by the legitimate sender or any legitimate receiver.

An implementation MUST be able to support at least 2 concurrent RIPv2 Security Associations on each RIP interface. This is a functional requirement for supporting key rollover. Support for key rollover is mandatory.

The RIPv2 Security Association, defined below, is selected by the sender based on the outgoing router interface. Each RIPv2 Security Association has a lifetime and other configuration parameters

associated with it. In normal operation, a RIPv2 Security Association is never used outside its lifetime. Certain abnormal cases are discussed later in this document.

The minimum data items in a RIPv2 Security Association are as follows:

KEY-IDENTIFIER (KEY-ID)
The unsigned 8-bit KEY-ID value is used to identify the RIPv2 Security Association in use for this packet.

The receiver uses the combination of the interface the packet was received upon and the KEY-ID value to uniquely identify the appropriate Security Association.

The sender selects which RIPv2 Security Association to use based on the outbound interface for this RIPv2 packet and then places the correct KEY-ID value into that packet. If multiple valid and active RIPv2 Security Associations exist for a given outbound interface at the time a RIPv2 packet is sent, the sender may use any of those security associations to protect the packet.

AUTHENTICATION ALGORITHM
This specifies the cryptographic algorithm and algorithm mode used with the RIPv2 Security Association. This information is never sent in cleartext over the wire. Because this information is not sent on the wire, the implementer chooses an implementation specific representation for this information. At present, the following values are possible: KEYED-MD5, HMAC-SHA-1, HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512.

AUTHENTICATION KEY
This is the value of the cryptographic authentication key used with the associated Authentication Algorithm. It MUST NOT ever be sent over the network in cleartext via any protocol. The length of this key will depend on the Authentication Algorithm in use. Operators should take care to select unpredictable and strong keys, avoiding any keys known to be weak for the algorithm in use. [ESC05] contains helpful information on both key generation techniques and cryptographic randomness.

SEQUENCE NUMBER
This is an unsigned 32-bit number. For a given KEY-ID value and sender, this number MUST NOT decrease. In normal operation, the operator should rekey the RIPv2 session prior to reaching the maximum value. The initial value used in the sequence number is arbitrary. Receivers SHOULD keep track of the most recent sequence number received from a given sender.

START TIME
This is a local representation of the day and time that this Security Association first becomes valid.

STOP TIME
This is a local representation of the day and time that this Security Association becomes invalid (i.e., when it expires). It is permitted, but not recommended, for an operator to configure this to "never expire". The "never expire" value is not recommended operational practice because it reduces security as compared with periodic rekeying. Normally, a RIPv2 Security Association is deleted at its STOP TIME. However, there are certain pathological cases, which are discussed in Section 5.1.

The authentication trailer consists of the Authentication Data, which is the output of the keyed cryptographic hash function. See later subsections of this section for details on computing this field.

2.3. Basic Authentication Processing

When the authentication type is "Cryptographic Hash Function", message processing is changed in message creation and reception as compared with the original RIPv2 specification in [Mal94].

This section describes the message processing generically. Additional algorithm-dependent processing that is required is described in separate, subsequent sections of this document. As of this writing, there are 2 kinds of algorithm-dependent processing. One covers the "Keyed-MD5" algorithm. The other covers the "HMAC-SHA1" family of algorithms.

2.3.1. Message Generation

The RIPv2 Packet is created as usual, with these exceptions:

(1) The UDP checksum SHOULD be calculated, but MAY be set to zero because any of the cryptographic authentication mechanisms in this specification will provide stronger integrity protection than the standard UDP checksum.

(2) The Authentication Type field indicates Cryptographic Authentication (3).

(3) The Authentication "password" field is reused to store a packet offset to the Authentication Data, a Key Identifier, the Authentication Data Length, and a non-decreasing sequence number.

See also Section 2.2 above on RIPv2 Security Association for other important background information.

When creating the RIPv2 Packet, the following process is followed:

(1) The Packet Length field of the RIPv2 header indicates the size of the main body of the RIPv2 packet.

(2) An appropriate RIPv2 Security Association is selected for use with this packet, based on the outbound interface for the packet. Any valid RIPv2 Security Association for that outbound interface may be used. The Authentication Data Offset, Key Identifier, and Authentication Data Length fields are filled in appropriately.

(3) Algorithm-dependent processing occurs now, either for the "Keyed-MD5" algorithm or for the "HMAC-SHA1" algorithm family. See the respective sub-sections (below) for details of this algorithm-dependent processing.

(4) The resulting Authentication Data value is written into the Authentication Data field. The trailing pad (if any) is not actually transmitted, as it is entirely predictable from the message length and Authentication Algorithm in use.

2.3.2. Message Reception

When the message is received, the process is reversed:

(1) The received Authentication Data is set aside and stored for later use,

(2) The appropriate RIPv2 Security Association is determined from the value of the Key Identifier field and the interface the packet was received on. If there is no valid RIPv2 Security Association for the received Key Identifier on the interface that the packet was received on, then:

(a) all processing of the incoming packet ceases, and

(b) a security event SHOULD be logged by the RIPv2 subsystem of the receiving system. That security event should indicate at

least the day/time that the bad packet was received, the Source IP Address of the received RIPv2 packet, the Key-ID field value, the interface the bad packet arrived upon, and the fact that no valid RIPv2 Security Association was found for that interface and Key-ID combination.

(3) Algorithm-dependent processing is performed, using the algorithm specified by the appropriate RIPv2 Security Association for this packet. This results in calculation of the Authentication Data based on the information in the received RIPv2 packet and information from the appropriate RIPv2 Security Association for that packet.

(4) The calculated Authentication Data result is compared with the received Authentication Data.

(5) If the calculated authentication data result does not match the received Authentication Data field, then:

(a) the message MUST be discarded without being processed, and

(b) a security event SHOULD be logged by the RIPv2 subsystem of the receiving system. That security event SHOULD indicate at least the day/time that the bad packet was received, the Source IP Address of the received RIPv2 packet, the Key-ID field value, the interface the bad packet arrived upon, and the fact that RIPv2 Authentication failed upon receipt of the packet.

(6) If the neighbor has been heard from recently enough to have viable routes in the local routing table, and the received sequence number is less than the last sequence number received, then the message MUST be discarded unprocessed. If the received sequence number is less than the last sequence number received, that fact SHOULD be logged as a security event. This logged security event SHOULD indicate at least the day/time that the bad packet was received, the Source IP Address of the received RIPv2 packet, the Key-ID field value, and the fact that an out-of-order RIPv2 sequence number was received.

When connectivity to the neighbor has been lost, the receiver SHOULD be ready to accept either:

- a message with a sequence number of zero.

- a message with a higher sequence number than the last received sequence number.

(7) Acceptable messages are now truncated to the RIPv2 message itself, minus the authentication trailer, and are processed normally (i.e., in accordance with the RIPv2 base specification in RFC 2453 [Mal98]). The last received sequence number for this RIPv2 Security Association and sender is also updated.

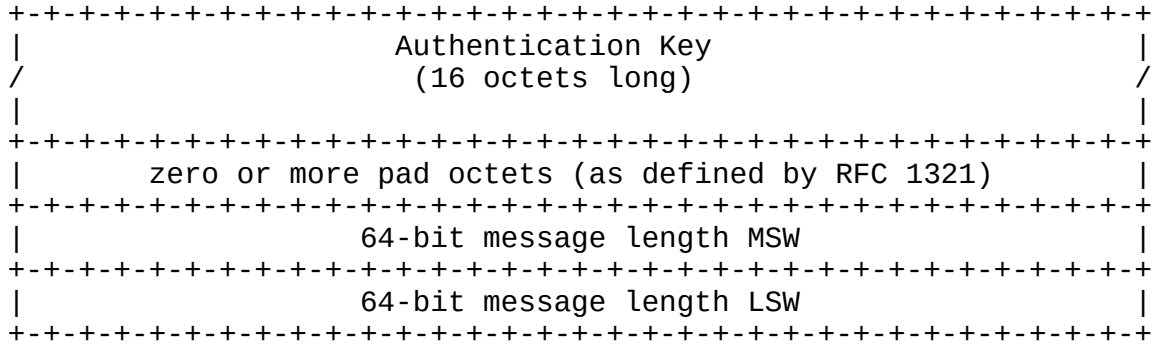
NOTA BENE: A router that has forgotten its current sequence number but remembers its Security Association MUST send its first packet with a sequence number of zero. This leaves a small opening for a replay attack. To reduce the risk of such attacks by precluding the situation where a router has forgotten its current sequence number, implementers SHOULD provide non-volatile storage for all components of a RIPv2 Security Association, and receiving systems SHOULD provide non-volatile storage for the last received sequence number from each sender. See also the Security Considerations section of this document.

2.4. Keyed-MD5 Algorithm-Dependent Processing

This section describes the algorithm-dependent processing steps applicable when the "Keyed-MD5" authentication algorithm is in use. The RIPv2 Authentication Key is always 16 octets when "Keyed-MD5" is in use.

(1) The RIPv2 Authentication Key is appended to the RIPv2 packet in memory.

(2) The Trailing Pad for MD5 and message length fields are added in memory. The diagram below shows how these additions appear when appended in memory:



(3) The Authentication Data is then calculated according to the MD5 algorithm defined by RFC 1321 [Rivest92].

2.5. HMAC-SHA1 Algorithm-Dependent Processing

This section describes the processing steps for HMAC Authentication. While HMAC was originally documented in [KMC97], for this specification, the terminology used in [FIPS-198] is used. While the current specification only provides full details for HMAC Authentication using the National Institute of Standards and Technology (NIST) SHA-1 algorithm (and its direct derivatives), this same basic process could be used with other cryptographic hash functions in the future. Because the RIPv2 packet is only hashed once, the overhead of the double hashing in this process is negligible.

The US NIST Secure Hash Standard (SHS), defined by [FIPS-180-2], includes specifications for SHA-1, SHA-256, SHA-384, and SHA-512. This specification defines processing for each of these.

The output of the cryptographic computations (e.g., HMAC-SHA1) is NOT truncated for RIPv2 Cryptographic Authentication.

The Authentication Data Length is equal to the Message Digest Size for the hash algorithm in use.

Any key value known to be weak with an algorithm defined by the NIST Secure Hash Standard MUST NOT be used with such an algorithm in an implementation of this specification. US NIST is the authoritative source for public information on weak keys for those algorithms.

In the algorithm description below, the following nomenclature, which is consistent with [FIPS-198], is used:

```
H    is the specific hashing algorithm,
     for example, SHA-1 or SHA-256.
Ko   is the cryptographic key used with the hash algorithm.
B    is the block-size of H, measured in octets, not bits.
     Note that B is the internal block size, not the hash size.
     For SHA-1   and SHA-256:  B == 64.
     For SHA-384 and SHA-512:  B == 128
L    is the length of the hash, measured in octets, not bits.
     For example, with SHA-1, L == 20.
XOR  is the exclusive-or operation.
Opad is the hexadecimal value 0x5c repeated B times.
Ipad is the hexadecimal value 0x36 repeated B times.
Apad is the hexadecimal value 0x878FE1F3 repeated (L/4) times.
```

(1) PREPARATION OF KEY
In this application, Ko is always L octets long.

If the Authentication Key is L octets long, then Ko is set equal to the Authentication Key. If the Authentication Key is more than L octets long, then Ko is set to H(Authentication Key). If the Authentication Key is less than L octets long, then Ko is set to the Authentication Key with zeros appended to the end of the Authentication Key such that Ko is L octets long.

(2) FIRST HASH
First, the RIPv2 packet's Authentication Data field is filled with the value Apad.

Then, a first hash, also known as the inner hash, is computed as follows:

First-Hash = H(Ko XOR Ipad || (RIPv2 Packet))

(3) SECOND HASH
Then a second hash, also known as the outer hash, is computed as follows:

Second-Hash = H(Ko XOR Opad || First-Hash)

(4) RESULT
The result Second-Hash becomes the authentication data that is sent in the Authentication Data field of the RIPv2 packet. The length of the Authentication Data field is always identical to the message digest size of the hash function H that is being used.

This also implies that use of hash functions with larger output sizes will also increase the size of the packet as transmitted on the wire.

3. Management Procedures

Key management is an important component of this mechanism and proper implementation is central to providing the intended level of risk reduction.

3.1. Key Management Requirements

It is strongly desirable that a hypothetical security breach in one Internet protocol not automatically compromise other Internet protocols. The Authentication Key of this specification SHOULD NOT be configured or stored using protocols (e.g., RADIUS) or cryptographic algorithms that have known flaws.

Implementations MUST support the storage of more than one key at the same time, although it is recognized that only one key will normally be active on an interface. Implementations MUST associate a specific Security Association lifetime (i.e., date/time first valid and date/time no longer valid) and a key identifier with each key. Implementations also MUST support manual key distribution. An example of manual key distribution is having the privileged user typing in the key, key lifetime, and key identifier on the router console. An operator may configure the Security Association lifetime to infinite, which means that the session is never rekeyed. However, instead, it is strongly recommended that operators rekey regularly, using a moderately short Security Association lifetime (e.g., 24 hours).

This specification requires support for at least two authentication algorithms, so the implementation MUST require that the authentication algorithm be specified for each key when the other key information is entered. Manual deletion of active Security Associations MUST be supported.

It is likely that the IETF will define a standard key management protocol for use with routing protocols. It is strongly desirable to use an IETF standards-track key management protocol to distribute RIPv2 Authentication Keys among communicating RIPv2 implementations. Such a protocol would provide scalability and significantly reduce the human administrative burden. The Key-ID field can be used as a hook between RIPv2 and such a future protocol.

Key management protocols have a long history of subtle flaws that are often discovered long after the protocol was first described in public. To avoid having to change all RIPv2 implementations should such a flaw be discovered, integrated key management protocol techniques were deliberately omitted from this specification.

3.2. Key Management Procedures

As with all security methods using keys, it is necessary to change the RIPv2 Authentication Key on a regular basis. To maintain routing stability during such changes, implementations MUST be able to store and use more than one RIPv2 Authentication Key on a given interface at the same time.

Each key will have its own Key Identifier (KEY-ID), which is stored locally. The combination of the Key Identifier and the interface associated with the message uniquely identifies the Authentication Algorithm and RIPv2 Authentication Key in use.

As noted above in Section 2.3.1, the party creating the RIPv2 message will select a valid RIPv2 Security Association from the set of valid RIPv2 Security Associations for that interface. The receiver MUST use the Key Identifier and receiving interface to determine which RIPv2 Security Association to use for authentication of the received message. More than one RIPv2 Security Association MAY be associated with an interface at the same time. The receiver MUST NOT simply try all RIPv2 Security Associations (i.e., keys) that might be configured for RIPv2 on the receiving interface, as that creates an easily exploited denial-of-service attack on the RIP subsystem of the receiver. (At least one widely used implementation of the previous version of this specification violates these requirements as of the publication date of this document and has consequent security vulnerabilities.)

Hence, it is possible to have fairly smooth RIPv2 Security Association (i.e., key) rollovers, without losing legitimate RIPv2 messages due to an invalid shared key and without requiring people to change all the keys at once. To ensure a smooth rollover, each communicating RIPv2 system must be updated with the new RIPv2 Security Association (including the new key) several minutes before the current RIPv2 Security Association will expire and several minutes before the new RIPv2 Security Association lifetime begins. Also, the new RIPv2 Security Association should have a lifetime that starts several minutes before the old RIPv2 Security Association expires. This gives time for each system to learn of the new security association before that security association will be used. It also ensures that the new security association will begin use and the current security association will go out of use before the current security association’s lifetime expires. For the duration of the overlap in security association lifetimes, a system may receive messages corresponding to either security association and successfully authenticate the message. The Key-ID in the received message is used to select the appropriate security association (i.e., key) to be used for authentication.

4. Conformance Requirements

For this specification, the term "conformance" has identical meaning to the phrase "full compliance".

The Keyed MD5 authentication algorithm and the HMAC-SHA1 algorithm MUST be implemented by all conforming implementations. In addition, the HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 algorithms SHOULD be implemented. MD5 is defined in [Rivest92]. SHA-1, SHA-256, SHA-384, and SHA-512 have been defined by the US NIST in [FIPS-180-2].

A conforming implementation MAY also support additional authentication algorithms, provided those additional algorithms are publicly and openly specified.

Manual key distribution as described above MUST be supported by all conforming implementations. All implementations MUST support the smooth key rollover described under "Key Management Procedures". This also means that implementations MUST support at least 2 concurrent RIPv2 Security Associations.

The user documentation provided with the implementation ought to contain clear instructions on how to configure the implementation such that smooth key rollover occurs successfully.

Implementations SHOULD support a standard key management protocol for secure distribution of RIPv2 Authentication Keys once such a key management protocol is standardized by the IETF.

The Security Considerations section of this document is an integral part of the specification, not just a discussion of the protocol.

5. Security Considerations

This entire memo describes and specifies an authentication mechanism for the RIPv2 routing protocol that is believed to be secure against passive attacks. The term "passive attack" is defined in RFC 1704 [HA94]. The analysis contained in RFC 1704 motivated this work. Passive attacks are clearly widespread in the Internet at present [HA94].

Protection against active attacks is incomplete in this current specification. The main issue relative to active attacks lies in the need to support the case where another router has recently rebooted and that router lacks the non-volatile storage needed to remember the RIPv2 Security Association(s) and last received RIPv2 sequence number(s) across that reboot.

5.1. Known Pathological Cases

Two known pathological cases exist that MUST be handled by implementations. Both of these are failures of the network manager. Each of these should be exceedingly rare in normal operation.

(1) During key rollover, devices might exist that have not yet been successfully configured with the new key. Therefore, routers SHOULD implement an algorithm that detects the set of RIPv2 Security Associations being used by its neighbors, and transmit its messages using both the new and old RIPv2 Security

Associations (i.e., keys) until all of the neighbors are using the new security association or the lifetime of the old security association expires. Under normal circumstances, this elevated transmission rate will exist for a single RIP update interval.

(2) In the event that the last RIPv2 Security Association of an interface expires, it is unacceptable to revert to an unauthenticated condition, and not advisable to disrupt routing. Therefore, the router MUST send a "last RIPv2 Security Association expiration" notification to the network manager (e.g., via SYSLOG, SNMP, and/or other means) and SHOULD treat that last Security Association as having an infinite lifetime until the lifetime is extended, the Security Association is deleted by network management, or a new security association is configured.

In some circumstances, the practice described in (2) can leave an opening to an active attack on the RIPv2 routing subsystem. Therefore, any actual occurrence of a RIPv2 Security Association expiration MUST cause a security event to be logged by the implementation. This log item MUST include at least a note that the RIPv2 Authentication Key expired, the RIP routing protocol instance(s) affected, the routing interfaces affected, the Key-ID that is affected, and the current date/time. Operators are encouraged to check such logs as an operational security practice to help detect active attacks on the RIPv2 routing subsystem. Further, implementations SHOULD provide a configuration knob ("fail secure") to let a network operator prefer to have the RIPv2 routing fail when the last key expires, rather than continue using RIPv2 in an insecure manner.

5.2 Network Management Considerations

Also, the use of SNMP, even SNMPv3 with cryptographic authentication and cryptographic confidentiality enabled, to modify or configure the RIPv2 Security Associations, or any component of the security association (for example, the cryptographic key), is NOT RECOMMENDED. This practice would create a potential for a cascading vulnerability, whereby a compromise in the SNMP security implementation would necessarily lead to a compromise not only of the local routing table (which could be accessed via SNMP) but also of all other routers that receive RIPv2 packets (directly or indirectly) from the compromised router.

Similarly, the use of protocols not designed and evaluated for use in key management (e.g., RADIUS, Diameter) to configure the security association is also NOT RECOMMENDED. Reading the Security Associations via SNMP is allowed, but the information is to be treated as security-sensitive and protected by using the priv mode.

Also, the use of SNMP to configure which form of RIPv2 authentication is in use is also NOT RECOMMENDED because of a similar cascading failure issue. Any future revision of the RIPv2 Management Information Base (MIB) [MB94] should consider making the rip2IfConfAuthType object read-only. Further, this object would need a new enum value to accommodate the RIPv2 cryptographic authentication type. In addition, the compliance statement for this MIB does not have a MIN-ACCESS for this object. At a minimum, if the MIB is updated, a new compliance statement SHOULD be written for this object that allows this object to be implemented as read-only. For the rip2ifConfAuthKey object, since this object always returns ''H when read, the object's MIN-ACCESS in any revised compliance statement SHOULD be not-accessible if the MIB is updated.

Further, for similar reasons, any future revisions to the RIPv2 Management Information Base (MIB) SHOULD deprecate or omit any objects that would permit the writing of any RIPv2 Security Association or RIPv2 Security Association component (e.g., the cryptographic key).

Also, it is RECOMMENDED that any future revisions to the RIPv2 Management Information Base (MIB) consider adding MIB objects to hold information about any RIPv2 security events that might have occurred, and MIB objects that could be used to read the set of security events that have been logged by the RIPv2 subsystem. For each security event mentioned in this document, it is also RECOMMENDED that appropriate notifications be included, with a MAX-ACCESS of Accessible-for-notify, in any future versions of the RIPv2 MIB module.

5.3. Key Management Considerations

For the past several years, manual configuration (e.g., via a console) has been commonly used to create and modify RIPv2 Security Associations. There are a number of large-scale RIP deployments today that successfully use manual configuration of RIPv2 Security Associations. There are also sites that use scripts (e.g., combining Tcl/Expect, PERL, and SSHv2) with a site-specific configuration database and secure console connections to dynamically manage all aspects of their router configurations, including their RIPv2 Security Associations. This last approach is similar to the current IETF approach to Network Configuration (NetConf) standards.

Recent IETF Multicast Security (MSEC) working group efforts into multicast key management appear promising. Several large RIPv2 deployments happen to also have deployed the Kerberos authentication system. Recent IETF work into the use of Kerberos for Internet Key Negotiation (KINK) also seems relevant; one might use Kerberos to support RIPv2 key management functions for use at sites that have already deployed Kerberos. It is hoped that in the future the IETF will standardize a key management protocol suitable for managing RIPv2 Security Associations.

5.4. Assurance Considerations

Users need to understand that the quality of the security provided by this mechanism depends completely on the strength of the implemented authentication algorithms, the strength of the key being used, and the correct implementation of the security mechanism in all communicating RIPv2 implementations. This mechanism also depends on the RIPv2 Authentication Key being kept confidential by all parties. If any of these are incorrect or insufficiently secure, then no real security will be provided to the users of this mechanism.

Use of high-assurance development methods is RECOMMENDED for implementations of this specification, in order to reduce the risk of subtle implementation flaws that might adversely impact the operational risk reduction that this specification seeks to provide.

5.5. Confidentiality and Traffic Analysis Considerations

Confidentiality is not provided by this mechanism. It is generally considered that an IP routing protocol does not require confidentiality, as the purpose of any routing protocols is to disseminate information about the topology of the network.

Protection against traffic analysis is also not provided. Mechanisms such as bulk link encryption SHOULD be used when protection against traffic analysis is required [CKHD89].

5.6. Other Security Considerations

Separately, the receipt of a RIPv2 packet using cryptographic authentication but containing an invalid or unknown Key-ID value might indicate an active attack on the RIP routing subsystem and is a significant security event. Therefore, any actual receipt of a RIPv2 packet using cryptographic authentication and containing an unknown, expired, or otherwise invalid KEY-ID value SHOULD cause a security event to be logged by the implementation. This log item SHOULD include at least the fact that the invalid KEY-ID was received, the source IP address of the packet containing the invalid KEY-ID, the

interface(s) the packet was received on, the KEY-ID received, and the current date/time.

A subtle user-interface consideration also should be noted. If a user interface only permits the entry of human-readable text (e.g., a password in US-ASCII format) for use as a cryptographic key, significant numbers of bits of the cryptographic key in use become predictable, thereby reducing the strength of the key in this context. For this reason, implementations of this specification SHOULD support the entry of RIPv2 cryptographic authentication keys in hexadecimal format.

5.7. Future Security Directions

Specification and deployment of a standards-track key management protocol that supports this RIPv2 cryptographic authentication mechanism would be a significant next step in operational risk reduction and might actually increase the ease of deployment and operation of this mechanism. Such specification is beyond the scope of this document. Recent IETF work in MSEC and KINK working groups appears promising in this regard. Recent IETF work in the NETCONF working group towards standardizing methods for secure configuration management of routers is also relevant.

Finally, we observe that this mechanism is not the final word on RIPv2 authentication. Rather, it is believed that this particular mechanism represents a significant risk reduction over previous methods (e.g., plaintext passwords), while remaining straightforward to implement correctly and also straightforward to deploy.

User communities that believe this mechanism is not adequate to their needs are encouraged to consider using digital signatures with RIPv2. [MBW97] specifies the use of OSPF with Digital signatures; that document might be a starting point for creating such a specification for the RIPv2 protocol. Digital signatures are significantly more expensive computationally and are also significantly more difficult to deploy operationally, as compared with the mechanism specified here. However, it appears likely that much of the mechanism in this document could be reused with digital signatures.

6. Acknowledgments

Fred Baker was co-author of the earlier RIPv2 MD5 Authentication document [AB97]. This document is a direct derivative of that earlier document, though it has been significantly reworked. The current authors would like to thank Bill Burr, Tim Polk, John Kelsey, and Morris Dworkin of (US) NIST for review of versions of this document.

7. Normative References

[BCP14] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[Mal98] Malkin, G., "RIP Version 2", STD 56, RFC 2453, November 1998.

[FIPS-180-2] National Institute of Standards and Technology, "Secure Hash Standard", FIPS PUB 180-2, August 2002, <http://csrc.nist.gov/publications/fips/fips180-2/fips180-2.pdf>.

[FIPS-198] National Institute of Standards and Technology, "The Keyed-Hash Message Authentication Code (HMAC)", FIPS PUB 198, March 2002, <http://csrc.nist.gov/publications/fips/fips198/fips-198a.pdf>.

8. Informative References

[AB97] Baker, F. and R. Atkinson, "RIP-2 MD5 Authentication", RFC 2082, January 1997.

[Bell89] S. Bellovin, "Security Problems in the TCP/IP Protocol Suite", ACM Computer Communications Review, Volume 19, Number 2, pp. 32-48, April 1989.

[CKHD89] Cole Jr, Raymond, Donald Kallgren, Richard Hale, and John R. Davis, "Multilevel Secure Mixed-Media Communication Networks", Proceedings of the IEEE Military Communications Conference (MILCOM ’89), IEEE, 1989.

[Dobb96a] Dobbertin, H., "Cryptanalysis of MD5 Compress", Technical Report, 2 May 1996. (Presented at Rump Session of EuroCrypt 1996.)

[Dobb96b] Dobbertin, H., "The Status of MD5 After a Recent Attack", CryptoBytes, Vol. 2, No. 2, Summer 1996.

[ESC05] Eastlake, D., 3rd, Schiller, J., and S. Crocker, "Randomness Requirements for Security", BCP 106, RFC 4086, June 2005.

[HA94] Haller, N. and R. Atkinson, "On Internet Authentication", RFC 1704, October 1994.

[KMC97] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997.

[Mal94] Malkin, G., "RIP Version 2 - Carrying Additional Information", RFC 1723, November 1994.

[MB94] Malkin, G. and F. Baker, "RIP Version 2 MIB Extension", RFC 1724, November 1994.

[MBW97] Murphy, S., Badger, M., and B. Wellington, "OSPF with Digital Signatures", RFC 2154, June 1997.

[Rivest92] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

Authors' Addresses

R. Atkinson
Extreme Networks
3585 Monroe Street
Santa Clara, CA 95051
USA

Phone: +1 (408) 579-2800
EMail: rja@extremenetworks.com

M. Fanto
(US) National Institute of Standards and Technology
Gaithersburg, MD 20878
USA

Phone: +1 (301) 975-2000
EMail: mattjf@umd.edu
Web: http://csrc.nist.gov

Full Copyright Statement

Intellectual Property

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.