Network Working Group
Internet-Draft
Intended status: Informational
Expires: April 12, 2007

S. Viswanathan
B. Weis
Cisco Systems
R. Bonica
Juniper Networks
A. Lange
Alcatel
O. Wheeler
BT

October 9, 2006

Authentication-Key Rollover mechanism for Routing and Management Protocols
draft-viswanathan-keyrollover-00.txt

Status of this Memo

Copyright Notice

Abstract

This memo discusses the authentication for routing and management protocols based on preconfigured keys,the need and basis for key rollover, and an mechanism to seamlessly rollover the authentication keys. It is intended for an application where secure administrative access to all the end-points of the protocol connection is normally available.

The strategy described herein improves upon the current practice where a key is preconifigured at all endpoints and the key rollover is done manually within a short synchronized window to avoid connection drops due to key mismatch.

Table of Contents

1. Conventions Used In This Document

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 RFC2119 [RFC2119].

2. Terminology

The following terms are used in this document:

key-list - A data structure used by the routing or management protocols.The key-list is a list of keys.

Key identifier - An identifier that signifies the key attributes associated with the authentication.

key - A member of the key-list. Each key contains an identifier, information that can be used to authenticate the protocol message, information that determines when the key can be used to authenticate an outbound message and information that determines when the key can be used to authenticate an inbound protocol message.

key lifetime - Denotes the window when a key remains in active state.

active key - A key used to generate authentication information for an outbound protocol message. Each key chain contains exactly one active key. The "active flag" on a key indicates whether a particular key qualifies to be active.

eligible key - Each key-list contains zero or more eligible keys. The receiving stattion uses the shared secret from a key to authenticate an incoming protocol message only if that key is eligible. The "eligible flag" on a key indicates whether a particular key is eligible.

3. Introduction

Many routing protocols authenticate messages by including a message authentication code (MAC) in message. To spoof a message, an attacker would not only have to approximate a valid message, but would also have to obtain the key that was used to calculate the MAC. This key never appears in the message stream.

RFC 3562 addresses key management considerations regarding one such MD5 based authentication scheme. Based upon the strength of the MD5 hashing algorithm, RFC 3562 recommends that keys be changed at least every 90 days.

Unfortunately, the authentication mechanisms described above permit keys to be changed during the lifetime of a routing adjacency only so long as the change is synchronized at both ends. This limitation has proven to be a significant deterrent to the effective deployment. This memo addresses that limitation.

Using other out-of-band key negotiation protocols like IKE present a different set of overheads and requirements that is out-of-scope for this document.

The need for an automated mechanism to rollover the keys at both endpoints is critical, and this document addresses a scheme to meet this requirement using the preconfigured keys.

4. Proposal

This memo proposes an authentication-key rollover mechanism for routing and management applications by extending the pre-configured key usage to a key-list as follows:

Network operators associate a key-list with each protected protocol connection. Each key-list includes a list of keys.Each key is associated with a unique identifier and several other data items that are described in Section 5 of this document.

The key identifier and the associated key used for computing the digest from the sending station must be identically configured on all the authenticating receiving stations. Whenever the protocol generates an outbound message,it searches the key-list for an active key. Section 6 of this document describes the active key selection criteria. If it does not find an active key, it discards the outbound message. However, if the protocol finds an active key, it calculates a MAC using information from the active key as per the protocol specification for authentication.

The receiving application associates its inbound message with a local key-list based upon its configuration. It then searches the associated key-list for a key whose identifier matches that which was specified by the incoming protocol message option. If it finds such a key and that key satisfies the eligibility criteria described in Section 7 of this document, the application uses the information from that key to calculate and verify the MAC as per its authentication handling specification. If no matching eligible key is found then it MUST declare an authentication failure and discard the protocol message.

5. Key Chain Attributes

This section describes information requirements for the key-list. It does not mandate any specific implementation.

A keychain is a set of keys, where each key is {A[i], K[i], V[i], S[i], T[i], S'[i], T'[i],F[i], F'[i]}:

For the purpose of this document, key[i] is defined as the key whose identifer is equal to i.

i - Key identifier, integer. The key identier range depends on the procotol specifics.
AK - Active key, integer. Indicates the choice of the key[i] amongst all the keys in the key-list to generate MAC by the sender.
AT - Accept tolerance, integer. It indicates the level of tolerance of clock skews.

A[i] - Authentication algorithm to use with key[i].
K[i] - Shared secret to use with key[i].
V[i] - A vector that determines whether key[i] is to be used to generate MACs for inbound protocol messages, outbound protocol messages, or both.
S[i] - Start time from which key[i] can be used by the sender.
T[i] - End time after which key[i] cannot be used by the sender.
S'[i] - Start time from which key[i] can be used by the receiver.
T'[i] - End time after which key[i] cannot be used by the receiver.
F[i] - Active flag that denotes the choice of the key for generating MACs for the outbound protocol messages. Only one key from the entire key-list is chosen as the active key.
F'[i]- Elibile flag that denotes the eligibility of the key for generating MACs for the verification of inbound protocol messages.

A[i] and K[i] MUST be configured symmetrically on all peers. That is, if key[i] is configured on two peer systems, A[i] and K[i] must be configured identically on each system.

S[i], T[i], S'[i] and T'[i] are measured from a defined epoch that must have a known relationship to UTC. For the purposes of discussion, times are assumed to be measured in seconds since that epoch, although this is not a requirement.

The range of values that can be specified for S[i], T[i], S'[i] and T'[i] includes two special values. The first special value is called NOW, and it represents the system time when the key is examined (as opposed to when the key is configured). The second special value, called INFINITY, represents a time beyond the end of the epoch.

S[i] and T[i] define a time-window during which a key can be used for sending. S'[i] and T'[i] define a time-window during which a key can be used for receiving.

AT, the accept tolerance defines the connection's tolerance to clock-skew on either system. The accept tolerance can be measured in the order of seconds, though a special tag for INFINITY can be provisioned.

Within a key-list, time-windows for sending can overlap. Likewise, within a key-list, time-windows for receiving can overlap. Typically, network operators will configure key-lists so that there are no gaps between time-windows for either sending or receiving. Implementations should issue a warning when network operators configure key-lists with gaps between time-windows. A gap of sufficient length can cause the the protocol connection/session to fail due to timeout.

The active flag F[i] is set when the system time >= the S[i] and is reset when the the system time equals or exceeds T[i].

In general, network operators should avoid reusing shared secrets. The degree to which an operator can reuse keys is defined by local security policy.

During the lifetime of a protocol connection/session, network operators may add and delete keys from the keychain. However, the network operator must ensure that an active key is always configured on all endpoints.

Implementations are free to implement key chains in any manner that satisfies the above stated information requirements. For example, implementations can translate the above stated information requirements directly into a data structure. Alternatively, they can implement one key-list for sending and another for receiving. In this case, the implementation may omit data items that do not apply to either the sending or receiving key-list.

Likewise, implementations can implement S[i] and T[i] as a start-time and an end-time. Alternatively, implementations can implement S[i] and T[i] as a start-time and a duration, or they can infer the T[i] from the S[i] of the next key.

6. Key rollover criteria

The key usage is strictly bound by the lifetime specification. It can only be used between S[i] and T[i], or between S'[i] and T'[i]. However, there may be a need to rollover a key while will within its active lifetime window. The authenticating protocol semantics(like sequence number wrap arounds) may also dictate a rollover based on the volume of data authenticated using the key K[i] triggering a rollover to the next active key.

7. Active Key Selection

The following paragraphs describe how the sending application selects the active key from its key-list. Implementations SHOULD support this key selection process; implementations MAY also support other active key selection mechanisms as a configurable option.

First, the application identifies all candidate keys that meet the following requirements:

- V[i] specifies that the key can be used for either sending or sending and receiving.
- the system time >= S[i]
- the system time < T[i]
- Active flag F[i] is set

Because the time-windows specified by S[i] and T[i] can overlap, it is possible that multiple keys will satisfy the above stated criteria. When this occurs, TCP chooses between the candidate keys by applying following rules in the order that they are listed below:

- prefer the youngest key (i.e., the key whose value for S[i] is greatest)
- When there is a tie based upon the above stated criterion, select the key whose identifier is numerically smallest.

In the case that an active key has already been deemed rolledover due the volume based criteria imposed by the application, then that key's active flag is reset, and the active key selection process is repeated.

The selection process described above is guaranteed to return zero or one active keys. If no active key is returned, the protocol discards the outbound message.

8. Key Eligibility

When the application receives a protocol message that includes the Authentication Option, it searches that connection's key-list for a key whose identifier is identical to Key ID specified by the incoming authentication Option. It uses that key to authenticate the incoming protocol message providing that the key is eligible to be used.

Implementations SHOULD support the following process for determining key eligibility; implementations MAY also support other eligible key selection mechanisms as a configurable option.

A key is eligible if all of the following criteria are met:

- V[i] specifies that the key can be used for either receiving or sending and receiving.
- A[i] is equal to the algorithm specified by the Authentication Option from the incoming protocol message
- the system time >= S'[i]
- the system time < T'[i]

If the protocol does not find a key whose identifier is identical to the Key ID specified by the incoming authentication Option, it MUST declare an authentication failure and discard the message. Likewise, it MUST declare an authentication failure if it finds the key but the key is not eligible.

9. Clock Synchronization

9.1. Overlapping lifetime

Clocks do not need to be synchronized accurately between the sending and receiving systems. The only requirements are that the key used to generate the MAC on the sending system is also configured on the receiving system and that the time ovelap between sender's active key and the receiver's eligible key is great enough to compensate for clock skew.

Receipt of a protocol message whose authentication data was generated using a key other than the one that is currently active on the receiving system does not constitute an error. It may indicate only that clocks are not synchronized between the sending and receiving systems.

9.2. Accept tolerance

To overcome the issues due to clock skews at the endpoints without needing to configure overlapping lifetime, a configurable tolerance level that the operator perceives to be acceptable is proposed. The tolerance level indicates the window of tolerance where-in a key is still considered eligible. In other words, a key is considered eligible from AT seconds prior to S'[i], upto AT seconds after T'[i].

10. Application Considerations

The mechanisms described in this memo are intended for use with routing and management applications that manipulate key set contents. The Key identifier is the critical component in handling keyrollover detection optimally. Protocols specifications that do not carry the key identifier in their authentication option header present an overhead in rollover detection. Depending on the number of eligible keys that are configured, the MAC computation and verification may need to be done on one or more of those.

11. Implications

11.1. Performance

The performance hit in calculating digests may inhibit the use of authentication option. Performance will vary depending upon processor type, authentication algorithm, packet size and number of MAC calculations per second.Protocols that do not carry the key identifier in its authentication option may at worst need to repeat the MAC caluculations for all keys that are eligble, thereby affecting performance.

12. Operational Considerations

Network operators may experience an operational need to make a key become both active and eligible immediately. In order to satisfy this need, the network operator should execute the following sequence:

Configure the key on both TCP peers with i equal to the lowest free value. On both systems, set S[i] and T[i] to INFINITY. This will cause the key to be perpetually inactive (for sending). Also set S'[i] to NOW and T'[i] to INFINITY. This will cause the key to be perpetually eligible (for receiving).

Once the above step has been completed, on both systems, set S[i] to NOW. This will cause the key[i] to become active. Now it is safe to remove or deactivate all other keys.

13. Contributors

The following individuals contributed to this document:

Chandrashekhar Appanna (achandra@cisco.com)

Anantha Ramaiah (ananth@cisco.com)

David McGrew (mcgrew@cisco.com)

Satish Mynam (mynam@cisco.com)

Andy Heffernan (ahh@juniper.net)

Kapil Jain (kapil@juniper.net)

14. Security Considerations

Management of authentication keys has been a significant operational problem, both in terms of key synchronization and key selection. For example, current guidance [RFC3562] warns against sharing RFC 2385 keys between systems, and recommends changing keys according to a schedule. The same general operational issues are relevant for the management of MAC keys.

15. IANA Considerations

None.

16. References

16.1. Normative References

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, September 1981.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998.

16.2. Informative References

[I-D.bonica-tcp-auth]
Bonica, R., Weis, B., Viswanathan, S., Lange, A., and O. Wheeler, "Authentication for TCP-based Routing and Management Protocols, draft-bonica-tcp-auth-05 (work in progress)", July 2006.

[RFC2385] Heffernan, A., "Protection of BGP Sessions via the TCP MD5 Signature Option", RFC 2385, August 1998.

[RFC3562] Leech, M., "Key Management Considerations for the TCP MD5 Signature Option", RFC 3562, July 2003.

Authors' Addresses

Sriram Viswanathan
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
US

Phone: +1 408-527-8830
Email: sriram_v@cisco.com

Brian Weis
Cisco Systems
170 W. Tasman Drive
San Jose, CA 95134
US

Phone: +1 408-526-6198
Email: rbonica@juniper.net

Ronald Bonica
Juniper Networks
2251 Corporate Park Drive
Herndon, VA 20171
US

Email: bew@cisco.com

Andrew Lange
Alcatel
710 E. Middlefield Road
Mountain View, CA 94043
US

Email: andrew.lange@alcatel.com

Owen N. Wheeler
BT
British Telecommunications plc
Adastral Park
Martlesham Heath
IPSWICH, Suffolk IP5 3RE
GB

Email: owen.wheeler@bt.com

Full Copyright Statement

Intellectual Property

Acknowledgment

Funding for the RFC Editor function is provided by the IETF Administrative Support Activity (IASA).