Internet Engineering Task Force (IETF)
Request for Comments: 7474
Updates: 2328, 5709
Category: Standards Track
ISSN: 2070-1721
April 2015

M. Bhatia
Ionos Networks
S. Hartman
Painless Security
D. Zhang
Huawei Technologies Co., Ltd.
A. Lindem, Ed.
Cisco

Security Extension for OSPFv2 When Using Manual Key Management

Abstract

The current OSPFv2 cryptographic authentication mechanism as defined in RFCs 2328 and 5709 is vulnerable to both inter-session and intra-session replay attacks when using manual keying. Additionally, the existing cryptographic authentication mechanism does not cover the IP header. This omission can be exploited to carry out various types of attacks.

This document defines changes to the authentication sequence number mechanism that will protect OSPFv2 from both inter-session and intra-session replay attacks when using manual keys for securing OSPFv2 protocol packets. Additionally, we also describe some changes in the cryptographic hash computation that will eliminate attacks resulting from OSPFv2 not protecting the IP header.

Status of This Memo

Table of Contents

1. Introduction

The OSPFv2 cryptographic authentication mechanism as described in [RFC2328] uses per-packet sequence numbers to provide protection against replay attacks. The sequence numbers increase monotonically so that attempts to replay stale packets can be thwarted. The sequence number values are maintained as a part of neighbor adjacency state. Therefore, if an adjacency is taken down, the associated sequence numbers get reinitialized and neighbor adjacency formation starts over again. Additionally, the cryptographic authentication mechanism does not specify how to deal with the rollover of a sequence number when its value wraps. These omissions can be exploited by attackers to implement various replay attacks ([RFC6039]). In order to address these issues, we define extensions to the authentication sequence number mechanism.

The cryptographic authentication as described in [RFC2328] and later updated in [RFC5709] does not include the IP header. This omission can be exploited to launch several attacks as the source address in the IP header is not protected. The OSPF specification, for broadcast and NBMA (Non-Broadcast Multi-Access) networks, requires implementations to use the source address in the IP header to determine the neighbor from which the packet was received. Changing the IP source address of a packet to a conflicting IP address can be exploited to produce a number of denial-of-service attacks [RFC6039]. If the packet is interpreted as coming from a different neighbor, the received sequence number state for that neighbor may be incorrectly updated. This attack may disrupt communication with a legitimate neighbor. Hello packets may be reflected to cause a neighbor to appear to have one-way communication. Additionally, Database Description packets may be reflected in cases where the per-packet sequence numbers are sufficiently divergent in order to disrupt an adjacency [RFC6863]. This is the IP-layer issue described in point 18 in Section 4 of [RFC6862].

[RFC2328] states that implementations MUST offer keyed MD5 authentication. It is likely that this will be deprecated in favor of the stronger algorithms described in [RFC5709] and required in [RFC6094].

This document defines a few simple changes to the cryptographic authentication mechanism, as currently described in [RFC5709], to prevent such IP-layer attacks.

1.1. Requirements Language

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 RFC 2119 [RFC2119].

When used in lowercase, these words convey their typical use in common language, and are not to be interpreted as described in RFC 2119 [RFC2119].

2. Replay Protection Using Extended Sequence Numbers

In order to provide replay protection against both inter-session and intra-session replay attacks, the OSPFv2 sequence number is expanded to 64 bits with the least significant 32-bit value containing a strictly increasing sequence number and the most significant 32-bit value containing the boot count. OSPFv2 implementations are required to retain the boot count in non-volatile storage for the deployment life of the OSPF router. The requirement to preserve the boot count is also placed on SNMP agents by the SNMPv3 security architecture (refer to snmpEngineBoots in Section 2.2 of [RFC3414]).

Since there is no room in the OSPFv2 packet for a 64-bit sequence number, it will occupy the 8 octets following the OSPFv2 packet and MUST be included when calculating the OSPFv2 packet digest. These additional 8 octets are not included in the OSPFv2 packet header length but are included in the OSPFv2 header Authentication Data length and the IPv4 packet header length.

The lower-order 32-bit sequence number MUST be incremented for every OSPF packet sent by the OSPF router. Upon reception, the sequence number MUST be greater than the sequence number in the last OSPF packet of that type accepted from the sending OSPF neighbor. Otherwise, the OSPF packet is considered a replayed packet and dropped. OSPF packets of different types may arrive out of order if they are prioritized as recommended in [RFC4222].

OSPF routers implementing this specification MUST use available mechanisms to preserve the sequence number's strictly increasing property for the deployed life of the OSPFv2 router (including cold restarts). This is achieved by maintaining a boot count in non-volatile storage and incrementing it each time the OSPF router loses its prior sequence number state. The SNMPv3 snmpEngineBoots variable [RFC3414] MAY be used for this purpose. However, maintaining a separate boot count solely for OSPF sequence numbers has the advantage of decoupling SNMP reinitialization and OSPF reinitialization. Also, in the rare event that the lower-order

32-bit sequence number wraps, the boot count can be incremented to preserve the strictly increasing property of the aggregate sequence number. Hence, a separate OSPF boot count is RECOMMENDED.

3. OSPF Packet Extensions

The OSPF packet header includes an authentication type field, and 64 bits of data for use by the appropriate authentication scheme (determined by the type field). Authentication types 0, 1, and 2 are defined [RFC2328]. This section defines Authentication type 3.

When using this authentication scheme, the 64-bit Authentication field (as defined in Appendix D.3 of [RFC2328]) in the OSPF packet header (as defined in Appendix A.3.1 of [RFC2328] and [RFC6549]) is changed as shown in Figure 1. The sequence number is removed and the Key ID is extended to 32 bits and moved to the former position of the sequence number.

Additionally, the 64-bit sequence number is moved to the first 64 bits following the OSPFv2 packet and is protected by the authentication digest. These additional 64 bits or 8 octets are included in the IP header length but not the OSPF header packet length.

Finally, the 0 field at the start of the OSPFv2 header authentication is extended from 16 bits to 24 bits.

```
    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
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |  Version #  |     Type      |        Packet length            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Router ID                            |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                           Area ID                             |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |           Checksum            | Instance ID   |  AuType       |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                              0                | Auth Data Len |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                          Key ID                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   |                    OSPF Protocol Packet                       |
   ~                                                               ~
   |                                                               |
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Sequence Number (Boot Count)                              |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |     Sequence Number (Strictly Increasing Packet Counter)      |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
   |                                                               |
   ~                 Authentication Data                           ~
   |                                                               |
   +-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 1: Extended Sequence Number Packet Extensions

4. OSPF Packet Key Selection

This section describes how this security solution selects long-lived keys from key tables. [RFC7210]. In this context, we are selecting the key and corresponding Security Association (SA) as defined in Section 3.2 of [RFC5709]. Generally, a key used for OSPFv2 packet authentication should satisfy the following requirements:

- o For packet transmission, the key validity interval as defined by SendLifetimeStart and SendLifetimeEnd must include the current time.

- o For packet reception, the key validity interval as defined by AcceptLifetimeStart and AcceptLifetimeEnd must include the current time.

- The key must be valid for the desired security algorithm.

In the remainder of this section, additional requirements for keys are enumerated for different scenarios.

4.1. Key Selection for Unicast OSPF Packet Transmission

Assume that a router R1 tries to send a unicast OSPF packet from its interface I1 to the interface I2 of a remote router R2 using security protocol P via interface I at time T. First, consider the circumstances where R1 and R2 are not connected with a virtual link. R1 then needs to select a long-lived symmetric key from its key table. Because the key should be shared by both R1 and R2 to protect the communication between I1 and I2, the key should satisfy the following requirements:

- The Peers field contains the area ID or, if no key containing the area ID is present, the string "all".

- The Direction field is either "out" or "both".

- The Interfaces field matches I1 or "all".

- If multiple keys match the Interface field, keys that explicitly match I1 should be preferred over keys matching "all". If there are still multiple keys that match, the key with the most recent SendLifetimeStart will be selected. This will facilitate graceful key rollover.

- The Key ID field in the OSPFv2 header (refer to Figure 1) will be set to the selected key's LocalKeyName.

When R1 and R2 are connected to a virtual link, the Peers field must identify the virtual endpoint rather than the virtual link. Since there may be virtual links to the same router, the transit area ID must be part of the identifier. Hence, the key should satisfy the following requirements:

- The Peers field includes both the virtual endpoint's OSPF router ID and the transit area ID for the virtual link in the form of the transit area ID, followed by a colon, followed by the router ID. If no such key exists, then a key with the Peers field set to the transit area ID is used, followed by a key with the Peers field set to "all".

- The Interfaces field is not used for key selection on virtual links.

- o The Direction field is either "out" or "both".

- o If multiple keys match the Peers field, keys that explicitly match the router ID should be preferred, followed by keys with a transit area specified, followed by keys with the Peers field set to "all". If there are still multiple keys that match, the key with the most recent SendLifetimeStart will be selected. This will facilitate graceful key rollover.

- o The Key ID field in the OSPFv2 header (refer to Figure 1) will be set to the selected key's LocalKeyName.

4.2. Key Selection for Multicast OSPF Packet Transmission

If a router R1 sends an OSPF packet from its interface I1 to a multicast address (i.e., AllSPFRouters or AllDRouters), it needs to select a key according to the following requirements:

- o First, try a key with the Peers field containing the area ID to which the interface belongs. If no key exists, try a key with the Peers field "all".

- o The Interfaces field matches the interface over which the packet is sent or "all".

- o The Direction field is either "out" or "both".

- o If multiple keys match the Interface field, keys that explicitly match I1 should be preferred over keys matching "all". If there are still multiple keys that match, the key with the most resent SendLifetimeStart will be selected. This will facilitate graceful key rollover.

- o The Key ID field in the OSPFv2 header (refer to Figure 1) will be set to the selected key's LocalKeyName.

4.3. Key Selection for OSPF Packet Reception

When cryptographic authentication is used, the ID of the authentication key is included in the authentication field of the OSPF packet header. Using this Key ID, it is straight forward for a receiver to locate the corresponding key. The simple requirements are:

- o The interface on which the key was received is associated with the key's interface.

- o The Key ID obtained from the OSPFv2 packet header corresponds to the neighbor's PeerKeyName. Since OSPFv2 keys are symmetric, the LocalKeyName and PeerKeyName for OSPFv2 keys will be identical. Hence, the Key ID will be used to select the correct local key.

- o The Direction field is either "in" or "both".

- o The Peers field matches as described in Sections Section 4.1 and Section 4.2.

5. Securing the IP Header

This document updates the definition of the Apad constant, as it is defined in [RFC5709], to include the IP source address from the IP header of the OSPFv2 protocol packet. The overall cryptographic authentication process defined in [RFC5709] remains unchanged. To reduce the potential for confusion, this section minimizes the repetition of text from RFC 5709 [RFC5709]. The changes are:

RFC 5709, Section 3.3 describes how the cryptographic authentication must be computed. In RFC 5709, the First-Hash includes the OSPF packet and Authentication Trailer. With this specification, the 64-bit sequence number will be included in the First-Hash along with the Authentication Trailer and OSPF packet.

RFC 5709, Section 3.3 also requires the OSPFv2 packet's Authentication Trailer (which is the appendage described in RFC 2328, Appendix D.4.3, page 233, items (6)(a) and (6)(d)) to be filled with the value Apad. Apad is a hexadecimal constant with the value 0x878FE1F3 repeated (L/4) times, where L is the length of the hash being used and is measured in octets rather than bits.

OSPF routers sending OSPF packets must initialize the first 4 octets of Apad to the value of the IP source address that would be used when sending the OSPFv2 packet. The remainder of Apad will contain the value 0x878FE1F3 repeated (L - 4)/4 times, where L is the length of the hash, measured in octets. The basic idea is to incorporate the IP source address from the IP header in the cryptographic authentication computation so that any change of IP source address in a replayed packet can be detected.

When an OSPF packet is received, implementations MUST initialize the first 4 octets of Apad to the IP source address from the IP header of the incoming OSPFv2 packet. The remainder of Apad will contain the value 0x878FE1F3 repeated (L - 4)/4 times, where L is the length of the hash, measured in octets. Besides changing the value of Apad, this document does not introduce any other changes to the authentication mechanism described in [RFC5709]. This would prevent all attacks where a rogue OSPF router changes the IP source address of an OSPFv2 packet and replays it on the same multi-access interface or another interface since the IP source address is now included in the cryptographic hash computation and modification would result in the OSPFv2 packet being dropped due to an authentication failure.

6. Mitigating Cross-Protocol Attacks

In order to prevent cross-protocol replay attacks for protocols sharing common keys, the two-octet OSPFv2 Cryptographic Protocol ID is appended to the authentication key prior to use. Refer to the IANA Considerations (Section 9).

[RFC5709], Section 3.3 describes the mechanism to prepare the key used in the hash computation. This document updates the text under "(1) PREPARATION OF KEY" as follows:

> The OSPFv2 Cryptographic Protocol ID is appended to the Authentication Key (K) yielding a Protocol-Specific Authentication Key (Ks). In this application, Ko is always L octets long. While [RFC2104] supports a key that is up to B octets long, this application uses L as the Ks length consistent with [RFC4822], [RFC5310], and [RFC5709]. According to [FIPS-198], Section 3, keys greater than L octets do not significantly increase the function strength. Ks is computed as follows:
>
> If the Protocol-Specific Authentication Key (Ks) is L octets long, then Ko is equal to Ks. If the Protocol-Specific Authentication Key (Ks) is more than L octets long, then Ko is set to H(Ks). If the Protocol-Specific Authentication Key (Ks) is less than L octets long, then Ko is set to the Protocol-Specific Authentication Key (Ks) with zeros appended to the end of the Protocol-Specific Authentication Key (Ks) such that Ko is L octets long.

Once the cryptographic key (Ko) used with the hash algorithm is derived, the rest of the authentication mechanism described in [RFC5709] remains unchanged other than one detail that was unspecified. When XORing Ko and Ipad of Opad, Ko MUST be padded with zeros to the length of Ipad or Opad. It is expected that implementations of [RFC5709] perform this padding implicitly.

7. Backward Compatibility

This security extension uses a new authentication type, AuType in the OSPFv2 header (refer to Figure 1). When an OSPFv2 packet is received and the AuType doesn't match the configured authentication type for the interface, the OSPFv2 packet will be dropped as specified in RFC 2328 [RFC2328]. This guarantees backward-compatible behavior consistent with any other authentication type mismatch.

8. Security Considerations

This document rectifies the manual key management procedure that currently exists within OSPFv2, as part of Phase 1 of the KARP Working Group. Therefore, only the OSPFv2 manual key management mechanism is considered. Any solution that takes advantage of the automatic key management mechanism is beyond the scope of this document.

The described sequence number extension offers most of the benefits of more complicated mechanisms without their attendant challenges. There are, however, a couple drawbacks to this approach. First, it requires the OSPF implementation to be able to save its boot count in non-volatile storage. If the non-volatile storage is ever repaired or upgraded such that the contents are lost or the OSPFv2 router is replaced, the authentication keys MUST be changed to prevent replay attacks.

Second, if a router is taken out of service completely (either intentionally or due to a persistent failure), the potential exists for reestablishment of an OSPFv2 adjacency by replaying the entire OSPFv2 session establishment. However, this scenario is extremely unlikely, since it would imply an identical OSPFv2 adjacency formation packet exchange. Without adjacency formation, the replay of OSPFv2 hello packets alone for an OSPFv2 router that has been taken out of service should not result in any serious attack, as the only consequence is superfluous processing. Of course, this attack could also be thwarted by changing the relevant manual keys.

This document also provides a solution to prevent certain denial-of-service attacks that can be launched by changing the source address in the IP header of an OSPFv2 protocol packet.

Using a single crypto sequence number can leave the router vulnerable to a replay attack where it uses the same source IP address on two different point-to-point unnumbered links. In such environments where an attacker can actively tap the point-to-point links, it's recommended that the user employ different keys on each of those unnumbered IP interfaces.

9. IANA Considerations

This document registers a new code point from the "OSPF Shortest Path First (OSPF) Authentication Codes" registry:

o 3 - Cryptographic Authentication with Extended Sequence Numbers.

This document also registers a new code point from the "Authentication Cryptographic Protocol ID" registry defined under "Keying and Authentication for Routing Protocols (KARP) Parameters":

o 3 - OSPFv2.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC2328] Moy, J., "OSPF Version 2", STD 54, RFC 2328, April 1998, <http://www.rfc-editor.org/info/rfc2328>.

[RFC5709] Bhatia, M., Manral, V., Fanto, M., White, R., Barnes, M., Li, T., and R. Atkinson, "OSPFv2 HMAC-SHA Cryptographic Authentication", RFC 5709, October 2009, <http://www.rfc-editor.org/info/rfc5709>.

10.2. Informative References

[FIPS-198] US National Institute of Standards and Technology, "The Keyed-Hash Message Authentication Code (HMAC)", FIPS PUB 198-1, July 2008.

[RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997, <http://www.rfc-editor.org/info/rfc2104>.

[RFC3414] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", STD 62, RFC 3414, December 2002, <http://www.rfc-editor.org/info/rfc3414>.

[RFC4222] Choudhury, G., Ed., "Prioritized Treatment of Specific OSPF Version 2 Packets and Congestion Avoidance", BCP 112, RFC 4222, October 2005, <http://www.rfc-editor.org/info/rfc4222>.

[RFC4822] Atkinson, R. and M. Fanto, "RIPv2 Cryptographic Authentication", RFC 4822, February 2007, <http://www.rfc-editor.org/info/rfc4822>.

[RFC5310] Bhatia, M., Manral, V., Li, T., Atkinson, R., White, R., and M. Fanto, "IS-IS Generic Cryptographic Authentication", RFC 5310, February 2009, <http://www.rfc-editor.org/info/rfc5310>.

[RFC6039] Manral, V., Bhatia, M., Jaeggli, J., and R. White, "Issues with Existing Cryptographic Protection Methods for Routing Protocols", RFC 6039, October 2010, <http://www.rfc-editor.org/info/rfc6039>.

[RFC6094] Bhatia, M. and V. Manral, "Summary of Cryptographic Authentication Algorithm Implementation Requirements for Routing Protocols", RFC 6094, February 2011, <http://www.rfc-editor.org/info/rfc6094>.

[RFC6549] Lindem, A., Roy, A., and S. Mirtorabi, "OSPFv2 Multi-Instance Extensions", RFC 6549, March 2012, <http://www.rfc-editor.org/info/rfc6549>.

[RFC6862] Lebovitz, G., Bhatia, M., and B. Weis, "Keying and Authentication for Routing Protocols (KARP) Overview, Threats, and Requirements", RFC 6862, March 2013, <http://www.rfc-editor.org/info/rfc6862>.

[RFC6863] Hartman, S. and D. Zhang, "Analysis of OSPF Security According to the Keying and Authentication for Routing Protocols (KARP) Design Guide", RFC 6863, March 2013, <http://www.rfc-editor.org/info/rfc6863>.

[RFC7210] Housley, R., Polk, T., Hartman, S., and D. Zhang, "Database of Long-Lived Symmetric Cryptographic Keys", RFC 7210, April 2014, <http://www.rfc-editor.org/info/rfc7210>.

Acknowledgments

Thanks to Ran Atkinson for help in the analysis of errata for RFC 6506, which led to clarifications in this document.

Thanks to Gabi Nakibly for pointing out a possible attack on P2P links.

Thanks to Suresh Krishnan for comments made during the Gen-Art review. In particular, thanks for pointing out an ambiguity in the initialization of Apad.

Thanks to Shaun Cooley for the security directorate review.

Thanks to Adrian Farrel for comments during the IESG last call.

Authors' Addresses

Manav Bhatia
Ionos Networks
Bangalore
India

EMail: manav@ionosnetworks.com

Sam Hartman
Painless Security

EMail: hartmans-ietf@mit.edu

Dacheng Zhang
Huawei Technologies Co., Ltd.
Beijing
China

EMail: dacheng.zhang@gmail.com

Acee Lindem (editor)
Cisco
United States

EMail: acee@cisco.com