IPv6 Segment Routing Security Considerations
draft-vyncke-6man-segment-routing-security-02

Abstract

   Segment Routing (SR) allows a node to steer a packet through a
   controlled set of instructions, called segments, by prepending a SR
   header to the packet. A segment can represent any instruction,
   topological or service-based. SR allows to enforce a flow through
   any path (topological, or application/service based) while
   maintaining per-flow state only at the ingress node to the SR domain.

   Segment Routing can be applied to the IPv6 data plane with the
   addition of a new type of Routing Extension Header. This document
   analyzes the security aspects of the Segment Routing Extension Header
   (SRH) and how it is used by SR capable nodes to deliver a secure
   service.

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].

Copyright Notice

Table of Contents

1. Introduction

   This document analyzes the security threat model, the security issues
   and proposed solutions related to the new routing header for segment
   routing with an IPv6 data plane.

The Segment Routing Header (SRH) is simply another type of the routing header as described in RFC 2460 [RFC2460] and is:

o inserted by a SR edge router when entering the segment routing domain or by the originating host itself. The source host can even be outside the SR domain;

o inspected and acted upon when reaching the destination address of the IP header per RFC 2460 [RFC2460].

Per RFC2460 [RFC2460], routers on the path that simply forward an IPv6 packet (i.e. the IPv6 destination address is none of theirs) will never inspect and process the content of SRH. Routers whose one interface IPv6 address equals the destination address field of the IPv6 packet MUST to parse the SRH and, if supported and if the local configuration allows it, MUST act accordingly to the SRH content.

According to RFC2460 [RFC2460], the default behavior of a non SR-capable router upon receipt of an IPv6 packet with SRH destined to an address of its, is to:

o ignore the SRH completely if the Segment Left field is 0 and proceed to process the next header in the IPv6 packet;

o discard the IPv6 packet if Segment Left field is greater than 0, it MAY send a Parameter Problem ICMP message back to the Source Address.

1.1. Segment Routing Documents

Segment Routing terminology is defined in [I-D.ietf-spring-segment-routing] and in [I-D.ietf-spring-problem-statement]. Segment Routing use cases are described in [I-D.filsfils-spring-segment-routing-use-cases]. Segment Routing protocol extensions are defined in [I-D.ietf-isis-segment-routing-extensions], and [I-D.ietf-ospf-ospfv3-segment-routing-extensions].

Segment Routing IPv6 use cases are described in [I-D.ietf-spring-ipv6-use-cases]. And the IPv6 Segment Routing header is described in [I-D.previdi-6man-segment-routing-header].

2. Threat model

2.1. Source routing threats

Using a SRH is similar to source routing, therefore it has some well-known security issues as described in RFC4942 [RFC4942] section 2.1.1 and RFC5095 [RFC5095]:

o amplification attacks: where a packet could be forged in such a way to cause looping among a set of SR-enabled routers causing unnecessary traffic, hence a Denial of Service (DoS) against bandwidth;

o reflection attack: where a hacker could force an intermediate node to appear as the immediate attacker, hence hiding the real attacker from naive forensic;

o bypass attack: where an intermediate node could be used as a stepping stone (for example in a De-Militarized Zone) to attack another host (for example in the datacenter or any back-end server).

2.2. Applicability of RFC 5095 to SRH

First of all, the reader must remember this specific part of section 1 of RFC5095 [RFC5095], "A side effect is that this also eliminates benign RH0 use-cases; however, such applications may be facilitated by future Routing Header specifications.". In short, it is not forbidden to create new secure type of Routing Header; for example, RFC 6554 (RPL) [RFC6554] also creates a new Routing Header type for a specific application confined in a single network.

In the segment routing architecture described in [I-D.ietf-spring-segment-routing] there are basically two kinds of nodes (routers and hosts):

o nodes within the SR domain, which is within one single administrative domain, i.e., where all nodes are trusted anyway else the damage caused by those nodes could be worse than amplification attacks: traffic interception, man-in-the-middle attacks, more server DoS by dropping packets, and so on.

o nodes outside of the SR domain, which is outside of the administrative segment routing domain hence they cannot be trusted because there is no physical security for those nodes, i.e., they can be replaced by hostile nodes or can be coerced in wrong behaviors.

The main use case for SR consists of the single administrative domain where only trusted nodes with SR enabled and configured participate

   in SR: this is the same model as in RFC6554 [RFC6554]. All non-
   trusted nodes do not participate as either SR processing is not
   enabled by default or because they only process SRH from nodes within
   their domain.

   Moreover, all SR nodes ignore SRH created by outsiders based on
   topology information (received on a peering or internal interface) or
   on presence and validity of the HMAC field. Therefore, if
   intermediate nodes ONLY act on valid and authorized SRH (such as
   within a single administrative domain), then there is no security
   threat similar to RH-0. Hence, the RFC 5095 [RFC5095] attacks are
   not applicable.

2.3. Service stealing threat

   Segment routing is used for added value services, there is also a
   need to prevent non-participating nodes to use those services; this
   is called 'service stealing prevention'.

2.4. Topology disclosure

   The SRH may also contains IPv6 addresses of some intermediate SR-
   nodes in the path towards the destination, this obviously reveals
   those addresses to the potentially hostile attackers if those
   attackers are able to intercept packets containing SRH. On the other
   hand, if the attacker can do a traceroute whose probes will be
   forwarded along the SR path, then there is little learned by
   intercepting the SRH itself. Also the clean-bit of SRH can help by
   removing the SRH before forwarding the packet to potentially a non-
   trusted part of the network.

2.5. ICMP Generation

   Per section 4.4 of RFC2460 [RFC2460], when destination nodes (i.e.
   where the destination address is one of theirs) receive a Routing
   Header with unsupported Routing Type, the required behavior is:

   o If Segments Left is zero, the node must ignore the Routing header
      and proceed to process the next header in the packet.

   o If Segments Left is non-zero, the node must discard the packet and
      send an ICMP Parameter Problem, Code 0, message to the packet's
      Source Address, pointing to the unrecognized Routing Type.

   This required behavior could be used by an attacker to force the
   generation of ICMP message by any node. The attacker could send
   packets with SRH (with Segment Left set to 0) destined to a node not
   supporting SRH. Per RFC2460 [RFC2460], the destination node could

generate an ICMP message, causing a local CPU utilization and if the source of the offending packet with SRH was spoofed could lead to a reflection attack without any amplification.

It must be noted that this is a required behavior for any unsupported Routing Type and not limited to SRH packets. So, it is not specific to SRH and the usual rate limiting for ICMP generation is required anyway for any IPv6 implementation and has been implemented and deployed for many years.

3. Security fields in SRH

This section summarizes the use of specific fields in the SRH; they are integral part of [I-D.previdi-6man-segment-routing-header] and they are again described here for reader's sake. They are based on a key-hashed message authentication code (HMAC).

The security-related fields in SRH are:

o HMAC Key-id, 8 bits wide;

o HMAC, 256 bits wide (optional, exists only if HMAC Key-id is not 0).

The HMAC field is the output of the HMAC computation (per RFC 2104 [RFC2104]) using a pre-shared key identified by HMAC Key-id and of the text which consists of the concatenation of:

o the source IPv6 address;

o First Segment field;

o an octet whose bit-0 is the clean-up bit flag and others are 0;

o HMAC Key-id;

o all addresses in the Segment List.

The purpose of the HMAC field is to verify the validity, the integrity and the authorization of the SRH itself. If an outsider of the SR domain does not have access to a current pre-shared secret, then it cannot compute the right HMAC field and the first SR router on the path processing the SRH and configured to check the validity of the HMAC will simply reject the packet.

The HMAC field is located at the end of the SRH simply because only the router on the ingress of the SR domain needs to process it, then all other SR nodes can ignore it (based on local policy) because they

trust the upstream router. This is to speed up forwarding operations
because SR routers which do not validate the SRH do not need to parse
the SRH until the end.

The HMAC Key-id field allows for the simultaneous existence of
several hash algorithms (SHA-256, SHA3-256 ... or future ones) as
well as pre-shared keys. This allows for pre-shared key roll-over
when two pre-shared keys are supported for a while when all SR nodes
converged to a fresher pre-shared key. The HMAC Key-id field is
opaque, i.e., it has neither syntax not semantic except as an index
to the right combination of pre-shared key and hash algorithm and
except that a value of 0 means that there is no HMAC field. It could
also allow for interoperation among different SR domains if allowed
by local policy and assuming a collision-free Key Id allocation.

When a specific SRH is linked to a time-related service (such as
turbo-QoS for a 1-hour period) where the DA, Segment ID (SID) are
identical, then it is important to refresh the shared-secret
frequently as the HMAC validity period expires only when the HMAC
Key-id and its associated shared-secret expires.

3.1. Selecting a hash algorithm

The HMAC field in the SRH is 256 bit wide. Therefore, the HMAC MUST
be based on a hash function whose output is at least 256 bits. If
the output of the hash function is 256, then this output is simply
inserted in the HMAC field. If the output of the hash function is
larger than 256 bits, then the output value is truncated to 256 by
taking the least-significant 256 bits and inserting them in the HMAC
field.

SRH implementations can support multiple hash functions but MUST
implement SHA-2 [FIPS180-4] in its SHA-256 variant.

NOTE: SHA-1 is currently used by some early implementations used for
quick interoperations testing, the 160-bit hash value must then be
right-hand padded with 96 bits set to 0. The authors understand that
this is not secure but is ok for limited tests.

3.2. Performance impact of HMAC

While adding a HMAC to each and every SR packet increases the
security, it has a performance impact. Nevertheless, it must be
noted that:

o the HMAC field is used only when SRH is inserted by a device (such
   as a home set-up box) which is outside of the segment routing
   domain. If the SRH is added by a router in the trusted segment

routing domain, then, there is no need for a HMAC field, hence no performance impact.

o   when present, the HMAC field MUST only be checked and validated by the first router of the segment routing domain, this router is named 'validating SR router'. Downstream routers MAY NOT inspect the HMAC field.

o   this validating router can also have a cache of <IPv6 header + SRH, HMAC field value> to improve the performance. It is not the same use case as in IPsec where HMAC value was unique per packet, in SRH, the HMAC value is unique per flow.

o   Last point, hash functions such as SHA-2 have been optmized for security and performance and there are multiple implementations with good performance.

With the above points in mind, the performance impact of using HMAC is minimized.

3.3. Pre-shared key management

The field HMAC Key-id allows for:

o   key roll-over: when there is a need to change the key (the hash pre-shared secret), then multiple pre-shared keys can be used simultaneously. The validating routing can have a table of <HMAC Key-id, pre-shared secret> for the currently active and future keys.

o   different algorithm: by extending the previous table to <HMAC Key-id, hash function, pre-shared secret>, the validating router can also support simultaneously several hash algorithms (see section Section 3.1)

The pre-shared secret distribution can be done:

o   in the configuration of the validating routers, either by static configuration or any SDN oriented approach;

o   dynamically using a trusted key distribution such as [RFC6407]

The intent of this document is NOT to define yet-another-key-distribution-protocol.

4. Deployment Models

4.1. Nodes within the SR domain

A SR domain is defined as a set of interconnected routers where all routers at the perimeter are configured to insert and act on SRH. Some routers inside the SR domain can also act on SRH or simply forward IPv6 packets.

The routers inside a SR domain can be trusted to generate SRH and to process SRH received on interfaces that are part of the SR domain. These nodes MUST drop all SRH packets received on an interface that is not part of the SR domain and containing a SRH whose HMAC field cannot be validated by local policies. This includes obviously packet with a SRH generated by a non-cooperative SR domain.

If the validation fails, then these packets MUST be dropped, ICMP error messages (parameter problem) SHOULD be generated (but rate limited) and SHOULD be logged.

4.2. Nodes outside of the SR domain

Nodes outside of the SR domain cannot be trusted for physical security; hence, they need to request by some trusted means (outside of the scope of this document) a complete SRH for each new connection (i.e. new destination address). The received SRH MUST include a HMAC Key-id and HMAC field which is computed correctly (see Section 3).

When an outside node sends a packet with an SRH and towards a SR domain ingress node, the packet MUST contain the HMAC Key-id and HMAC field and the the destination address MUST be an address of a SR domain ingress node .

The ingress SR router, i.e., the router with an interface address equals to the destination address, MUST verify the HMAC field with respect to the HMAC Key-id.

If the validation is successful, then the packet is simply forwarded as usual for a SR packet. As long as the packet travels within the SR domain, no further HMAC check needs to be done. Subsequent routers in the SR domain MAY verify the HMAC field when they process the SRH (i.e. when they are the destination).

If the validation fails, then this packet MUST be dropped, an ICMP error message (parameter problem) SHOULD be generated (but rate limited) and SHOULD be logged.

4.3. SR path exposure

As the intermediate SR nodes addresses appears in the SRH, if this
SRH is visible to an outsider then he/she could reuse this knowledge
to launch an attack on the intermediate SR nodes or get some insider
knowledge on the topology. This is especially applicable when the
path between the source node and the first SR domain ingress router
is on the public Internet.

The first remark is to state that 'security by obscurity' is never
enough; in other words, the security policy of the SR domain MUST
assume that the internal topology and addressing is known by the
attacker. A simple traceroute will also give the same information
(with even more information as all intermediate nodes between SID
will also be exposed). IPsec Encapsulating Security Payload
[RFC4303] cannot be use to protect the SRH as per RFC4303 the ESP
header must appear after any routing header (including SRH).

To prevent a user to leverage the gained knowledge by intercepting
SRH, it it recommended to apply an infrastructure Access Control List
(iACL) at the edge of the SR domain. This iACL will drop all packets
from outside the SR-domain whose destination is any address of any
router inside the domain. This security policy should be tuned for
local operations.

4.4. Impact of BCP-38

BCP-38 [RFC2827], also known as "Network Ingress Filtering", checks
whether the source address of packets received on an interface is
valid for this interface. The use of loose source routing such as
SRH forces packets to follow a path which differs from the expected
routing. Therefore, if BCP-38 was implemented in all routers inside
the SR domain, then SR packets could be received by an interface
which is not expected one and the packets could be dropped.

As a SR domain is usually a subset of one administrative domain, and
as BCP-38 is only deployed at the ingress routers of this
administrative domain and as packets arriving at those ingress
routers have been normally forwarded using the normal routing
information, then there is no reason why this ingress router should
drop the SRH packet based on BCP-38. Routers inside the domain
commonly do not apply BCP-38; so, this is not a problem.

5. IANA Considerations

There are no IANA request or impact in this document.

6. **Manageability Considerations**

    TBD

7. **Security Considerations**

    Security mechanisms applied to Segment Routing over IPv6 networks are
    detailed in Section 3.

8. **Acknowledgements**

    The authors would like to thank Dave Barach and Stewart Bryant for
    their contributions to this document.

9. **References**

9.1. **Normative References**

[FIPS180-4]
            National Institute of Standards and Technology, "FIPS
            180-4 Secure Hash Standard (SHS)", March 2012,
            <http://csrc.nist.gov/publications/fips/fips180-4/
            fips-180-4.pdf>.

[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.

[RFC4303]   Kent, S., "IP Encapsulating Security Payload (ESP)", RFC
            4303, December 2005.

[RFC5095]   Abley, J., Savola, P., and G. Neville-Neil, "Deprecation
            of Type 0 Routing Headers in IPv6", RFC 5095, December
            2007.

[RFC6407]   Weis, B., Rowles, S., and T. Hardjono, "The Group Domain
            of Interpretation", RFC 6407, October 2011.

9.2. **Informative References**

   [I-D.filsfils-spring-segment-routing-use-cases]
            Filsfils, C., Francois, P., Previdi, S., Decraene, B.,
            Litkowski, S., Horneffer, M., Milojevic, I., Shakir, R.,
            Ytti, S., Henderickx, W., Tantsura, J., Kini, S., and E.
            Crabbe, "Segment Routing Use Cases", draft-filsfils-
            spring-segment-routing-use-cases-01 (work in progress),
            October 2014.

   [I-D.ietf-isis-segment-routing-extensions]
            Previdi, S., Filsfils, C., Bashandy, A., Gredler, H.,
            Litkowski, S., Decraene, B., and J. Tantsura, "IS-IS
            Extensions for Segment Routing", draft-ietf-isis-segment-
            routing-extensions-03 (work in progress), October 2014.

   [I-D.ietf-ospf-ospfv3-segment-routing-extensions]
            Psenak, P., Previdi, S., Filsfils, C., Gredler, H.,
            Shakir, R., Henderickx, W., and J. Tantsura, "OSPFv3
            Extensions for Segment Routing", draft-ietf-ospf-ospfv3-
            segment-routing-extensions-02 (work in progress), February
            2015.

   [I-D.ietf-spring-ipv6-use-cases]
            Brzozowski, J., Leddy, J., Leung, I., Previdi, S.,
            Townsley, W., Martin, C., Filsfils, C., and R. Maglione,
            "IPv6 SPRING Use Cases", draft-ietf-spring-ipv6-use-
            cases-03 (work in progress), November 2014.

   [I-D.ietf-spring-problem-statement]
            Previdi, S., Filsfils, C., Decraene, B., Litkowski, S.,
            Horneffer, M., and R. Shakir, "SPRING Problem Statement
            and Requirements", draft-ietf-spring-problem-statement-03
            (work in progress), October 2014.

   [I-D.ietf-spring-segment-routing]
            Filsfils, C., Previdi, S., Bashandy, A., Decraene, B.,
            Litkowski, S., Horneffer, M., Shakir, R., Tantsura, J.,
            and E. Crabbe, "Segment Routing Architecture", draft-ietf-
            spring-segment-routing-01 (work in progress), February
            2015.

   [I-D.previdi-6man-segment-routing-header]
            Previdi, S., Filsfils, C., Field, B., and I. Leung, "IPv6
            Segment Routing Header (SRH)", draft-previdi-6man-segment-
            routing-header-05 (work in progress), January 2015.

   [RFC2104] Krawczyk, H., Bellare, M., and R. Canetti, "HMAC: Keyed-
            Hashing for Message Authentication", RFC 2104, February
            1997.

[RFC2827]   Ferguson, P. and D. Senie, "Network Ingress Filtering:
            Defeating Denial of Service Attacks which employ IP Source
            Address Spoofing", BCP 38, RFC 2827, May 2000.

[RFC4942]   Davies, E., Krishnan, S., and P. Savola, "IPv6 Transition/
            Co-existence Security Considerations", RFC 4942, September
            2007.

[RFC6554]   Hui, J., Vasseur, JP., Culler, D., and V. Manral, "An IPv6
            Routing Header for Source Routes with the Routing Protocol
            for Low-Power and Lossy Networks (RPL)", RFC 6554, March
            2012.

Authors' Addresses

   Eric Vyncke (editor)
   Cisco Systems, Inc.
   De Kleetlaann 6A
   Diegem 1831
   Belgium

   Email: evyncke@cisco.com

   Stefano Previdi
   Cisco Systems, Inc.
   Via Del Serafico, 200
   Rome   00142
   Italy

   Email: sprevidi@cisco.com

   David Lebrun
   Universite Catholique de Louvain
   Place Ste Barbe, 2
   Louvain-la-Neuve, 1348
   Belgium

   Email: david.lebrun@uclouvain.be