Internet Engineering Task Force (IETF)
Request for Comments: 6411
Category: Informational
ISSN: 2070-1721

M. Behringer
F. Le Faucheur
B. Weis
Cisco Systems

October 2011

Applicability of Keying Methods for RSVP Security

Abstract

The Resource reSerVation Protocol (RSVP) allows hop-by-hop integrity protection of RSVP neighbors. This requires messages to be cryptographically protected using a shared secret between participating nodes. This document compares group keying for RSVP with per-neighbor or per-interface keying, and discusses the associated key provisioning methods as well as applicability and limitations of these approaches. This document also discusses applicability of encrypting RSVP messages.

Copyright Notice

Table of Contents

1. Introduction and Problem Statement

The Resource reSerVation Protocol [RFC2205] allows hop-by-hop authentication of RSVP neighbors, as specified in [RFC2747]. In this mode, an integrity object is attached to each RSVP message to transmit a keyed message digest. This message digest allows the recipient to verify the identity of the RSVP node that sent the message and to validate the integrity of the message. Through the inclusion of a sequence number in the scope of the digest, the digest also offers replay protection.

[RFC2747] does not dictate how the key for the integrity operation is derived. Currently, most implementations of RSVP use a statically configured key, per interface or per neighbor. However, to manually configure a key per router pair across an entire network is operationally hard, especially when key changes are to be performed on a regular basis. Effectively, many users of RSVP therefore resort to using the same key throughout their RSVP network, and they change it rarely, if ever, because of the operational burden. However, it is often necessary to change keys due to network operational requirements (e.g., change of operational staff).

This document discusses a variety of keying methods and their applicability to different RSVP deployment environments, for both message integrity and encryption. It is meant as a comparative guide to understand where each RSVP keying method is best deployed and the limitations of each method. Furthermore, it discusses how RSVP hop-by-hop authentication is impacted in the presence of non-RSVP nodes, or subverted nodes, in the reservation path.

"RSVP Security Properties" ([RFC4230]) provides an overview of RSVP security, including RSVP Cryptographic Authentication [RFC2747], but does not discuss key management. It states that "RFC 2205 assumes that security associations are already available". The present document focuses specifically on key management with different key types, including group keys. Therefore, this document complements [RFC4230].

1.1. Terminology

A security domain is defined in this document as two or more nodes that share a common RSVP security policy.

When a key is mentioned in this document, it is a symmetric key. A symmetric key best meets the operational requirements of RSVP deployments and is the only type of key currently explicitly supported for protecting RSVP messages.

2. The RSVP Hop-by-Hop Trust Model

Many protocol security mechanisms used in networks require and use per-peer authentication. Each hop authenticates messages from its neighbor with a shared key or certificate. This is also the model used for RSVP. Trust in this model is transitive. Each RSVP node trusts explicitly only its RSVP next-hop peers, through the message digest contained in the INTEGRITY object. The next-hop RSVP speaker in turn trusts its own peers and so on. See also "RSVP Security Properties" [RFC4230] for more background.

The keys used for protecting RSVP messages can, in particular, be group keys (for example, distributed via the Group Domain of Interpretation (GDOI) [RFC6407], as discussed in [GDOI-MAC]). If a group key is used, the authentication granularity becomes group membership of devices, not (individual) peer authentication between devices.

The trust an RSVP node has to another RSVP node within a common security domain has an explicit and an implicit component. Explicitly, the node trusts the other node to maintain the RSVP messages intact or confidential, depending on whether authentication or encryption (or both) is used. This means only that the message has not been altered or seen by another, non-trusted node. Implicitly, each node trusts the other node to maintain the level of protection specified within that security domain. In any group-keying scheme like GDOI, a node trusts all the other members of the group (because the authentication is now based on group membership, as noted above).

The RSVP protocol can operate in the presence of a non-RSVP router in the path from the sender to the receiver. The non-RSVP hop will ignore the RSVP message and just pass it along. The next RSVP node can then process the RSVP message. For RSVP authentication or encryption to work in this case, the key used for computing the RSVP message digest needs to be shared by the two RSVP neighbors, even if they are not IP neighbors. In the presence of non-RSVP hops, while an RSVP node always knows the next IP hop before forwarding an RSVP message, it does not always know the RSVP next hop. In fact, part of the role of a Path message is precisely to discover the RSVP next hop (and to dynamically re-discover it when it changes, for example, because of a routing change). Thus, the presence of non-RSVP hops impacts operation of RSVP authentication or encryption and may influence the selection of keying approaches.

Figure 1 illustrates this scenario. R2 in this picture does not participate in RSVP; the other nodes do. In this case, R2 will pass on any RSVP messages unchanged and will ignore them.

```
                  ----R3---
                 /         \
  sender----R1---R2(*)       R4----receiver
                 \         /
                  ----R5---

               (*) Non-RSVP hop
```

Figure 1: A Non-RSVP Node in the Path

This creates a challenge for RSVP authentication and encryption. In the presence of a non-RSVP hop, with some RSVP messages such as a PATH message, an RSVP router does not know the RSVP next hop for that message at the time of forwarding it. For example, in Figure 1, R1 knows that the next IP hop for a Path message addressed to the receiver is R2, but it does not necessarily know if the RSVP next hop is R3 or R5. This means that per-interface and per-neighbor keys cannot easily be used in the presence of non-RSVP routers on the path between senders and receivers.

Section 4.3 of [RFC2747] states that "... the receiver MAY initiate an integrity handshake with the sender". If this handshake is taking place, it can be used to determine the identity of the next RSVP hop. In this case, non-RSVP hops can be traversed also using per-interface or per-neighbor keys.

Group keying will naturally work in the presence of non-RSVP routers. Referring back to Figure 1, with group keying, R1 would use the group key to protect a Path message addressed to the receiver and forwards it to R2. Being a non-RSVP node, R2 will ignore and forward the Path message to R3 or R5 depending on the current shortest path as determined by routing. Whether it is R3 or R5, the RSVP router that receives the Path message will be able to authenticate the message successfully using the group key.

3. Applicability of Key Types for RSVP

3.1. Per-Interface and Per-Neighbor Keys

Most current RSVP authentication implementations support per-interface RSVP keys. When the interface is point-to-point (and therefore an RSVP router has only a single RSVP neighbor on each interface), this is equivalent to per-neighbor keys in the sense that a different key is used for each neighbor. In the point-to-point case, the security domain is simply between the router and its neighbor. However, when the interface is multipoint, all RSVP speakers on a given subnet have to belong to the same security domain and share the same key in this model. This makes it unsuitable for deployment scenarios where nodes from different security domains are present on a subnet, for example, Internet exchange points. In such cases, per-neighbor keys are required, and the security domain is between the router and its neighbor.

With per-neighbor keys, each RSVP key is bound to an interface plus a neighbor on that interface. It allows for the existence of different security domains on a single interface and subnet.

Per-interface and per-neighbor keys can be used within a single security domain.

These key types can also be used between security domains, since they are specific to a particular interface or neighbor.

Both monotonically increasing sequence number (e.g., the INTEGRITY object simple sequence numbers [RFC2747], or the Encapsulating Security Payload (ESP) and Authentication Header (AH) anti-replay service [RFC4301] sequence numbers) and time-based anti-replay methods (e.g., the INTEGRITY sequence numbers based on a clock [RFC2747]) can be used with per-neighbor and per-interface keys.

As discussed in the previous section, per-neighbor and per-interface keys can not be used in the presence of non-RSVP hops.

3.2. Group Keys

In the case of group keys, all members of a group of RSVP nodes share the same key. This implies that a node uses the same key regardless of the next RSVP hop that will process the message (within the group of nodes sharing the particular key). It also implies that a node will use the same key on the receiving as on the sending side (when exchanging RSVP messages within the group).

Group keys apply naturally to intra-domain RSVP authentication, where all RSVP nodes are part of the same security domain and implicitly trust each other. The nodes also extended trust to a group key server (GKS), which administers group membership and provides group keys. This is represented in Figure 2.

```
    ......GKS1............
    :    :   :   :        :
    :    :   :   :        :
source--R1--R2--R3-----destination
|                                 |
|<-----domain 1----------------->|
```

Figure 2: A Group Key Server within a Single Security Domain

A single group key cannot normally be used to cover multiple security domains because, by definition, the different domains do not trust each other. They would therefore not be willing to trust the same group key server. For a single group key to be used in several security domains, there is a need for a single group key server, which is trusted by both sides. While this is theoretically possible, in practice it is unlikely that there is a single such entity trusted by both domains. Figure 3 illustrates this setup.

```
             ...............GKS1...................
             :    :   :   :        :   :   :       :
             :    :   :   :        :   :   :       :
         source--R1--R2--R3--------R4--R5--R6--destination
         |                  |    |                      |
         |<-----domain 1--->|    |<-------domain 2----->|
```

Figure 3: A Single Group Key Server across Security Domains

A more practical approach for RSVP operation across security domains, is to use a separate group key server for each security domain, and to use per-interface or per-neighbor keys between the two domains (thus comprising a third security domain). Figure 4 shows this setup.

```
             ....GKS1......        ....GKS2.........
             :    :   :   :        :   :   :       :
             :    :   :   :        :   :   :       :
         source--R1--R2--R3--------R4--R5--R6--destination
         |                  |    |                      |
         |<-----domain 1--->|    |<-------domain 2----->|
                            |<-->|
                           domain 3
```

Figure 4: A Group Key Server per Security Domain

As discussed in Section 2, group keying can be used in the presence of non-RSVP hops.

Because a group key may be used to verify messages from different peers, monotonically increasing sequence number methods are not appropriate. Time-based anti-replay methods (e.g., the INTEGRITY sequence numbers based on a clock [RFC2747]) can be used with group keys.

4. Key Provisioning Methods for RSVP

4.1. Static Key Provisioning

Static keys are preconfigured, either manually or through a network management system. The simplest way to implement RSVP authentication is to use static keys. Static keying can be used with per-interface keys, per-neighbor keys, or group keys.

The provisioning of static keys requires either manual operator intervention on each node or a network management system performing the same task. Time synchronization of static key provisioning and changes is critical in order to avoid inconsistent keys within a security domain.

Static key provisioning is therefore not an ideal model in a large network.

Often, the number of interconnection points across two domains where RSVP is allowed to transit is relatively small and well controlled. Also, the different domains may not be in a position to use an infrastructure trusted by both domains to update keys on both sides. Thus, statically provisioned keys may be applicable to inter-domain RSVP authentication.

Since it is not feasible to carry out a key change at the exact same time in communicating RSVP nodes, some grace period needs to be implemented during which an RSVP node will accept both the old and the new key. Otherwise, RSVP operation would suffer interruptions. (Also with dynamic keying approaches, there can be a grace period where two keys are valid at the same time; however, the grace period in manual keying tends to be significantly longer than with dynamic key rollover schemes.)

4.2. Dynamic Keying

4.2.1. Per-Neighbor and Per-Interface Key Negotiation

To avoid the problem of manual key provisioning and updates in static key deployments, key negotiation between RSVP neighbors could be used to derive either per-interface or per-neighbor keys.

4.2.2. Dynamic Group Key Distribution

With this approach, group keys are dynamically distributed among a set of RSVP routers. For example, [GDOI-MAC] describes a mechanism to distribute group keys to a group of RSVP speakers, using GDOI [RFC6407]. In this solution, each RSVP node requests a group key from a key server as part of an encrypted and integrity-protected key agreement protocol. Once the key server has authenticated and authorized the RSVP nodes, it distributes a group key to the group member. The authentication in this model can be based on public key mechanisms, thereby avoiding the need for static key provisioning.

5. Specific Cases Supporting Use of Group Keying

5.1. RSVP Notify Messages

[RFC3473] introduces the Notify message and allows such messages to be sent in a non-hop-by-hop fashion. As discussed in the Security Considerations section of [RFC3473], this can interfere with RSVP's hop-by-hop integrity and authentication model. [RFC3473] describes how standard IPsec-based integrity and authentication can be used to protect Notify messages.

Group keying may allow use of regular RSVP authentication [RFC2747] for protection of non-hop-by-hop Notify messages. For example, RSVP Notify messages commonly used for traffic engineering in MPLS networks are non-hop-by-hop messages. Such messages may be sent from an ingress node directly to an egress node. Group keying in such a case avoids the establishment of node-to-node keying when node-to-node keying is not otherwise used.

5.2. RSVP-TE and GMPLS

Use of RSVP authentication for RSVP-TE [RFC3209] and for RSVP-TE Fast Reroute [RFC4090] deserves additional considerations.

With the facility backup method of Fast Reroute, a backup tunnel from the Point of Local Repair (PLR) to the Merge Point (MP) is used to protect Label Switched Paths (protected LSPs) against the failure of a facility (e.g., a router) located between the PLR and the MP. During the failure of the facility, the PLR redirects a protected LSP inside the backup tunnel and as a result, the PLR and MP then need to exchange RSVP control messages between each other (e.g., for the maintenance of the protected LSP). Some of the RSVP messages between the PLR and MP are sent over the backup tunnel (e.g., a Path message from PLR to MP), while some are directly addressed to the RSVP node (e.g., a Resv message from MP to PLR). During the rerouted period, the PLR and the MP effectively become RSVP neighbors, while they may not be directly connected to each other and thus do not behave as RSVP neighbors in the absence of failure. This point is raised in the Security Considerations section of [RFC4090] that says: "Note that the facility backup method requires that a PLR and its selected merge point trust RSVP messages received from each other". Such environments may benefit from group keying. A group key can be used among a set of routers enabled for Fast Reroute, thereby easily ensuring that PLR and MP authenticate messages from each other, without requiring prior specific configuration of keys, or activation of key update mechanism, for every possible pair of PLR and MP.

Where RSVP-TE or RSVP-TE Fast Reroute is deployed across AS boundaries (see [RFC4216]), the considerations presented above in Sections 3.1 and 3.2 apply, such that per-interface or per-neighbor keys can be used between two RSVP neighbors in different ASes (independently of the keying method used by the RSVP router to talk to the RSVP routers in the same AS).

[RFC4875] specifies protocol extensions for support of Point-to-Multipoint (P2MP) RSVP-TE. RSVP message integrity mechanisms for hop-by-hop RSVP signaling apply to the hop-by-hop P2MP RSVP-TE signaling (see the Security Considerations in [RFC4875]).

[RFC4206] defines LSP Hierarchy with GMPLS TE and uses non-hop-by-hop signaling. Because it reuses LSP Hierarchy procedures for some of its operations, P2MP RSVP-TE also uses non-hop-by-hop signaling. Both LSP hierarchy and P2MP RSVP-TE rely on the security mechanisms defined in [RFC3473] and [RFC4206] for non hop-by-hop RSVP-TE signaling. Group keying can simplify protection of non-hop-by-hop signaling for LSP Hierarchy and P2MP RSVP-TE.

6. Applicability of IPsec for RSVP

6.1. General Considerations Using IPsec

The discussions about the various keying methods in this document are also applicable when using IPsec [RFC4301] to protect RSVP. Section 1.2 of [RFC2747] states that IPsec is not an optimal choice to protect RSVP. The key argument is that an IPsec security association (SA) and an RSVP SA are not based on the same parameters. Nevertheless, IPsec can be used to protect RSVP. The Security Policy Database (SPD) traffic selectors for related RSVP flows will not be constant. In some cases, the source and destination addresses are end hosts, and sometimes they are RSVP routers. Therefore, traffic selectors in the SPD are expected to specify ANY for the source address and destination addresses, and to specify IP protocol 46 (RSVP).

"The Multicast Group Security Architecture" [RFC3740] defines in detail a "Group Security Association" (GSA). This definition is also applicable in the context discussed here, and allows the use of IPsec for RSVP. The existing GDOI specification [RFC6407] manages group security associations, which can be used by IPsec. An example GDOI policy would be to encrypt or authenticate all packets of the RSVP protocol itself (IP protocol 46). A router implementing GDOI and the AH and/or ESP protocols is therefore able to implement this policy.

Because the traffic selectors for an SA cannot be predicted, SA lookup is expected to use only the Security Parameters Index (SPI) (or SPI plus protocol).

6.2. Comparing AH and the INTEGRITY Object

The INTEGRITY object defined by [RFC2747] provides integrity protection for RSVP also in a group-keying context, as discussed above. AH [RFC4302] is an alternative method to provide integrity protection for RSVP packets.

The RSVP INTEGRITY object protects the entire RSVP message, but does not protect the IP header of the packet nor the IP options (in IPv4) or extension headers (in IPv6).

AH tunnel mode (transport mode is not applicable; see Section 6.4) protects the entire original IP packet, including the IP header of the original IP packet ("inner header"), IP options or extension headers, plus the entire RSVP packet. It also protects the immutable fields of the outer header.

The difference between the two schemes in terms of covered fields is therefore whether the IPv4 header and IP options, or the IPv6 header and extension headers, of the original IP packet are protected (as is the case with AH) or not (as is the case with the INTEGRITY object). Also, AH covers the immutable fields of the outer header.

As described in the next section, IPsec tunnel mode cannot be applied for RSVP traffic in the presence of non-RSVP nodes; therefore, the security associations in both cases, AH and INTEGRITY object, are between the same RSVP neighbors. From a keying point of view, both approaches are therefore comparable.

6.3. Applicability of Tunnel Mode

IPsec tunnel mode encapsulates the original packet, prepending a new IP header plus an ESP or AH sub-header. The entire original packet plus the ESP/AH sub-header is secured. However, in the case of ESP, the new, outer IP header is not cryptographically secured in this process.

Protecting RSVP packets with IPsec tunnel mode works with any of the keying methods described above (per-interface, per-neighbor, or group keying), as long as there are no non-RSVP nodes on the path (however, see the group-keying considerations below). For RSVP messages to be visible and considered at each hop, such a tunnel would not cross routers, but each RSVP node would establish a tunnel with each of its peers, effectively leading to link protection.

In the presence of a non-RSVP hop, tunnel mode cannot be applied because a router upstream from a non-RSVP hop does not know the next RSVP hop, and thus cannot apply the correct tunnel header. The same situation applies to a host attached to the network by a non-RSVP-enabled first hop. This is independent of the key type used.

The use of group keying with ESP tunnel mode where a security gateway places a peer security gateway address as the destination of the ESP packet has consequences. In particular, if a man-in-the-middle attacker redirects the ESP-protected reservation to a different security gateway, the receiving security gateway cannot detect that the destination address was changed. However, it has received and will act upon an RSVP reservation that will be routed along an unintended path. Because RSVP routers encountering the RSVP packet path will not be aware that this is an unintended path, they will act upon it, and the resulting RSVP state along both the intended path and unintended path will be incorrect. Therefore, using group keying with ESP tunnel mode is not recommended, unless address preservation is used (see Section 6.5).

6.4. Non-Applicability of Transport Mode

IPsec transport mode, as defined in [RFC4303] is not suitable for securing RSVP Path messages, since those messages preserve the original source and destination. [RFC4301] states explicitly that "the use of transport mode by an intermediate system (e.g., a security gateway) is permitted only when applied to packets whose source address (for outbound packets) or destination address (for inbound packets) is an address belonging to the intermediate system itself". This would not be the case for RSVP Path messages.

6.5. Applicability of Tunnel Mode with Address Preservation

When the identity of the next-hop RSVP peer is not known, it is not possible to use a tunnel-endpoint destination address in the tunnel mode outer IP header. Section 3.1 of "Multicast Extensions to the Security Architecture for the Internet Protocol" [RFC5374] defines a new tunnel mode: tunnel mode with address preservation. This mode copies the destination and optionally the source address from the inner header to the outer header. Therefore, the encapsulated packet will have the same destination address as the original packet, and will be normally subject to the same routing decisions. While [RFC5374] is focusing on multicast environments, tunnel mode with address preservation can be used also to protect unicast traffic in conjunction with group keying. In this tunnel mode, the RSVP speakers act as security gateways because they maintain the original end-system addresses of the RSVP packets in the tunnel mode outer IP header. This addressing scheme is used by RSVP to ensure that the packets continue along the routed path toward the destination end host.

Tunnel mode with address preservation, in conjunction with group keying, allows the use of AH or ESP for protection of RSVP even in cases where non-RSVP nodes have to be traversed. This is because it allows routing of the IPsec-protected packet through the non-RSVP nodes in the same way as if it were not IPsec protected.

When used with group keying, tunnel mode with address preservation can be used to mitigate redirection attacks where a man-in-the-middle modifies the destination of the outer IP header of the tunnel mode packet. The inbound processing rules for tunnel mode with address preservation (Section 5.2 of [RFC5374]) require that the receiver verify that the addresses in the outer IP header and the inner IP header are consistent. Therefore, the attack can be detected, and RSVP reservations will not proceed along an unintended path.

7. End-Host Considerations

Unless RSVP Proxy entities [RFC5945] are used, RSVP signaling is controlled by end systems and not routers. As discussed in [RFC4230], RSVP allows both user-based security and host-based security. User-based authentication aims at "providing policy based admission control mechanism based on user identities or application" [RFC3182]. To identify the user or the application, a policy element called AUTH_DATA, which is contained in the POLICY_DATA object, is created by the RSVP daemon at the user's host and transmitted inside the RSVP message. This way, a user may authenticate to the Policy Decision Point (or directly to the first-hop router). Host-based security relies on the same mechanisms as between routers (i.e., the INTEGRITY object) as specified in [RFC2747]. For host-based security, per-interface or per-neighbor keys may be used; however, key management with statically provisioned keys can be difficult in a large-scale deployment, as described in Section 4. In principle, an end host can also be part of a group key scheme, such as GDOI. If the end systems are part of the same security domain as the RSVP hops in the network, group keying can be extended to include the end systems. If the end systems and the network are in different zones of trust, group keying cannot be used.

8. Applicability to Other Architectures and Protocols

While, so far, this document discusses only RSVP security assuming the traditional RSVP model as defined by [RFC2205] and [RFC2747], the analysis is also applicable to other RSVP deployment models as well as to similar protocols. These include the following:

- "Aggregation of RSVP for IPv4 and IPv6 Reservations" [RFC3175]: This scheme defines aggregation of individual RSVP reservations, and discusses use of RSVP authentication for the signaling messages. Group keying is applicable to this scheme, particularly when automatic Deaggregator discovery is used, since in that case, the Aggregator does not know ahead of time which Deaggregator will intercept the initial end-to-end RSVP Path message.

- "Generic Aggregate Resource ReSerVation Protocol (RSVP) Reservations" [RFC4860]: This document also discusses aggregation of individual RSVP reservations. Here again, group keying applies and is mentioned in the Security Considerations section.

- "Aggregation of Resource ReSerVation Protocol (RSVP) Reservations over MPLS TE/DS-TE Tunnels" [RFC4804]: This scheme also defines a form of aggregation of RSVP reservation, but this time over MPLS-TE tunnels. Similarly, group keying may be used in such an environment.

- "Pre-Congestion Notification (PCN) Architecture" [RFC5559]: defines an architecture for flow admission and termination based on aggregated pre-congestion information. One deployment model for this architecture is based on Intserv over Diffserv: the Diffserv region is PCN-enabled. Also, RSVP signaling is used end-to-end, but the PCN-domain is a single RSVP hop, i.e., only the PCN-boundary-nodes process RSVP messages. In this scenario, RSVP authentication may be required among PCN-boundary-nodes, and the considerations about keying approaches discussed earlier in this document apply. In particular, group keying may facilitate operations since the ingress PCN-boundary-node does not necessarily know ahead of time which PCN-egress-node will intercept and process the initial end-to-end Path message. From the viewpoint of securing end-to-end RSVP in this scenario (from the end host to the PCN-ingress-node, to the PCN-egress-node, to the other end host), there are a lot of similarities in scenarios involving RSVP Aggregation over aggregate RSVP reservations [RFC3175] [RFC4860], RSVP Aggregation over MPLS-TE tunnels [RFC4804], and RSVP (Aggregation) over PCN ingress-egress aggregates.

9. Summary

The following table summarizes the various approaches for RSVP keying, and their applicability to various RSVP scenarios. In particular, such keying can be used for RSVP authentication (e.g., using the RSVP INTEGRITY object or AH) and/or for RSVP encryption (e.g., using ESP in tunnel mode).

	per-neighbor / per-interface keys	group keys
Works intra-domain	Yes	Yes
Works inter-domain	Yes	No
Works over non-RSVP hops	No	Yes (1)
Dynamic keying	Yes (IKE)	Yes (e.g., GDOI)

Table 1: Overview of Keying Approaches and Their Applicability

(1): RSVP integrity with group keys works over non-RSVP nodes; RSVP encryption with ESP and RSVP authentication with AH work over non-RSVP nodes in tunnel mode with address preservation; RSVP encryption with ESP and RSVP authentication with AH do not work over non-RSVP nodes in tunnel mode.

We also make the following observations:

- All key types can be used statically, or with dynamic key negotiation. This impacts the manageability of the solution, but not the applicability itself.

- For encryption of RSVP messages, IPsec ESP in tunnel mode can be used.

- There are some special cases in RSVP, like non-RSVP hosts, the Notify message (as discussed in Section 5.1, the various RSVP deployment models discussed in Section 8, and MPLS Traffic Engineering and GMPLS discussed in Section 5.2, which would benefit from a group-keying approach.

10. Security Considerations

This entire document discusses RSVP security; this section describes specific security considerations relating to subverted RSVP nodes.

10.1. Subverted Nodes

An undetected subverted node, for example, one that an intruder has gained control over, is still implicitly a trusted node. However, it is a threat to the security of RSVP. Since RSVP authentication is hop by hop and not end to end, a subverted node in the path breaks the chain of trust. This is, to a large extent, independent of the type of keying used.

For per-interface or per-neighbor keying, the subverted node can now introduce fake messages to its neighbors. This can be used in a variety of ways, for example, by changing the receiver address in the Path message or by generating fake Path messages. This allows path states to be created on every RSVP router along any arbitrary path through the RSVP domain. That in itself could result in a form of denial of service by allowing exhaustion of some router resources (e.g., memory). The subverted node could also generate fake Resv messages upstream corresponding to valid Path states. In doing so, the subverted node can reserve excessive amounts of bandwidth thereby possibly performing a denial-of-service attack.

Group keying allows the additional abuse of sending fake RSVP messages to any node in the RSVP domain, not just adjacent RSVP nodes. However, in practice, this can be achieved to a large extent also with per-neighbor or per-interface keys, as discussed above. Therefore, the impact of subverted nodes on the path is comparable for all keying schemes discussed here (per-interface, per-neighbor, and group keys).

11. Acknowledgements

The authors would like to thank everybody who provided feedback on this document. Specific thanks to Bob Briscoe, Hannes Tschofenig, Ran Atkinson, Stephen Kent, and Kenneth G. Carlberg.

12. Informative References

[GDOI-MAC] Weis, B. and S. Rowles, "GDOI Generic Message Authentication Code Policy", Work in Progress, September 2011.

[RFC2205] Braden, B., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.

[RFC2747] Baker, F., Lindell, B., and M. Talwar, "RSVP Cryptographic Authentication", RFC 2747, January 2000.

[RFC3175] Baker, F., Iturralde, C., Le Faucheur, F., and B. Davie, "Aggregation of RSVP for IPv4 and IPv6 Reservations", RFC 3175, September 2001.

[RFC3182] Yadav, S., Yavatkar, R., Pabbati, R., Ford, P., Moore, T., Herzog, S., and R. Hess, "Identity Representation for RSVP", RFC 3182, October 2001.

[RFC3209] Awduche, D., Berger, L., Gan, D., Li, T., Srinivasan, V., and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels", RFC 3209, December 2001.

[RFC3473] Berger, L., "Generalized Multi-Protocol Label Switching (GMPLS) Signaling Resource ReserVation Protocol-Traffic Engineering (RSVP-TE) Extensions", RFC 3473, January 2003.

[RFC3740] Hardjono, T. and B. Weis, "The Multicast Group Security Architecture", RFC 3740, March 2004.

[RFC4090] Pan, P., Swallow, G., and A. Atlas, "Fast Reroute Extensions to RSVP-TE for LSP Tunnels", RFC 4090, May 2005.

[RFC4206] Kompella, K. and Y. Rekhter, "Label Switched Paths (LSP) Hierarchy with Generalized Multi-Protocol Label Switching (GMPLS) Traffic Engineering (TE)", RFC 4206, October 2005.

[RFC4216] Zhang, R. and J. Vasseur, "MPLS Inter-Autonomous System (AS) Traffic Engineering (TE) Requirements", RFC 4216, November 2005.

[RFC4230] Tschofenig, H. and R. Graveman, "RSVP Security Properties", RFC 4230, December 2005.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, December 2005.

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

[RFC4804] Le Faucheur, F., "Aggregation of Resource ReSerVation Protocol (RSVP) Reservations over MPLS TE/DS-TE Tunnels", RFC 4804, February 2007.

[RFC4860] Le Faucheur, F., Davie, B., Bose, P., Christou, C., and M. Davenport, "Generic Aggregate Resource ReSerVation Protocol (RSVP) Reservations", RFC 4860, May 2007.

[RFC4875] Aggarwal, R., Papadimitriou, D., and S. Yasukawa, "Extensions to Resource Reservation Protocol - Traffic Engineering (RSVP-TE) for Point-to-Multipoint TE Label Switched Paths (LSPs)", RFC 4875, May 2007.

[RFC5374] Weis, B., Gross, G., and D. Ignjatic, "Multicast Extensions to the Security Architecture for the Internet Protocol", RFC 5374, November 2008.

[RFC5559] Eardley, P., "Pre-Congestion Notification (PCN) Architecture", RFC 5559, June 2009.

[RFC5945] Le Faucheur, F., Manner, J., Wing, D., and A. Guillou, "Resource Reservation Protocol (RSVP) Proxy Approaches", RFC 5945, October 2010.

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

Authors' Addresses

Michael H. Behringer
Cisco Systems
Village d'Entreprises Green Side
400, Avenue Roumanille, Batiment T 3
Biot - Sophia Antipolis 06410
France

EMail: mbehring@cisco.com
URI: http://www.cisco.com

Francois Le Faucheur
Cisco Systems
Village d'Entreprises Green Side
400, Avenue Roumanille, Batiment T 3
Biot - Sophia Antipolis 06410
France

EMail: flefauch@cisco.com
URI: http://www.cisco.com

Brian Weis
Cisco Systems
170 W. Tasman Drive
San Jose, California 95134-1706
USA

EMail: bew@cisco.com
URI: http://www.cisco.com