Applicability of Keying Methods for RSVP Security
draft-ietf-tsvwg-rsvp-security-groupkeying-02.txt

Status of this Memo

Abstract

The Resource reSerVation Protocol (RSVP) allows hop-by-hop
authentication of RSVP neighbors. This requires messages to be
cryptographically signed 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. The present document also discusses applicability
of group keying to RSVP encryption.

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 authenticity 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 key per router pair across an entire network is
   operationally hard, especially for key changes. Effectively, many
   users of RSVP therefore resort to the same key throughout their RSVP
   network, and change it rarely if ever, because of the operational
   burden. [RFC3562] however recommends regular key changes, at least
   every 90 days.

   The present document discusses the various keying methods and their
   applicability to different RSVP deployment environments, for both
   message integrity and encryption. It does not recommend any
   particular method or protocol (e.g., RSVP authentication versus IPsec
   AH), but is meant as a comparative guideline to understand where each
   RSVP keying method is best deployed, and its limitations.
   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.

   The document "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].

2. **The RSVP Hop-by-Hop Trust Model**

   Many protocol security mechanisms used in networks require and use
   per peer authentication. Each hop authenticates 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
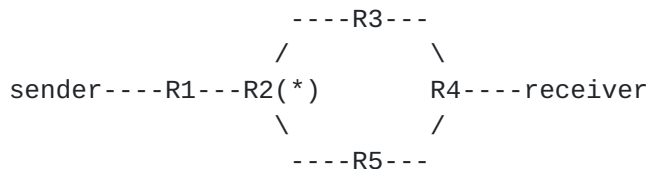   the INTEGRITY object. The next hop RSVP speaker in turn trusts its

own peers and so on. See also the document "RSVP security
properties" [RFC4230] for more background.

The keys used for generating the RSVP messages can, in particular, be
group keys (for example distributed via GDOI [RFC3547], as discussed
in [I-D.weis-gdoi-mac-tek]).

The trust an RSVP node has to another RSVP node 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 each other node with which
it has a trust relationship established via the mechanisms here to
adhere to the protocol specifications laid out by the various
standards. Note that in any group keying scheme like GDOI a node
trusts explicitly as well as implicitly all the other members of the
group.

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. However, 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 necessarily not 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.

By contrast, 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 sign a Path message addressed to the
receiver and forwards it to R2. Being a non-RSVP node, R2 and 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 it successfully with the group key.

[3](#). Applicability of Key Types for RSVP

[3.1](#). Interface and neighbor based keys

Most current RSVP authentication implementations support interface
based RSVP keys. When the interface is point-to-point (and therefore
an RSVP router only has a single RSVP neighbor on each interface),
this is equivalent to neighbor based keys in the sense that a
different key is used for each neighbor. However, when the interface
is multipoint, all RSVP speakers on a given subnet have to share the
same key in this model, which makes it unsuitable for deployment
scenarios where different trust groups share a subnet, for example
Internet exchange points. In such a case, neighbor based keys are
required.

With neighbor based keys, an RSVP key is bound to an interface plus a
neighbor on that interface. It allows the distinction of different
trust groups on a single subnet. (Assuming that layer-2 security is
correctly implemented to prevent layer-2 attacks.)

Per interface and per neighbor keys can be used within a single
security domain. As mentioned above, per interface keys are only
applicable when all the nodes reachable on the specific interface
belong to the same security domain.

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

level keys can only be deployed safely when all the reachable
neighbors on the interface belong to the same security domain.

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

Here, 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, since
all RSVP nodes implicitly trust each other. Using group keys, they
extend this trust to the group key server. This is represented in
Figure 2.

```
        ......GKS1.............
        :    :   :   :          :
        :    :   :   :          :
   source--R1--R2--R3-----destination
    |                             |
    |<-----domain 1------------------>|
```
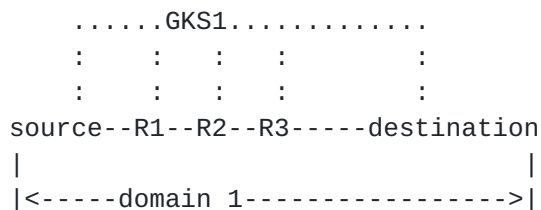
         Figure 2: Group Key Server within a single security domain

A single group key cannot normally be used to cover multiple security
domains however, because by definition the different domains do not
trust each other and would 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
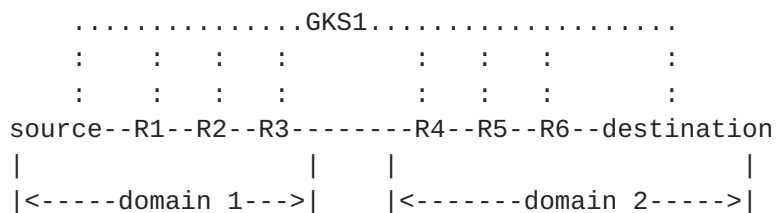to use per interface or per neighbor authentication between the two
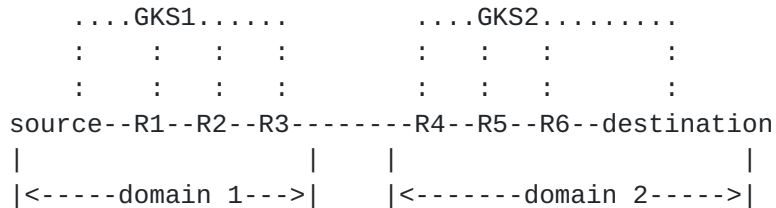domains. Figure 4 shows this set-up.

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

                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.

4. Key Provisioning Methods for RSVP

4.1. Static Key Provisioning

The simplest way to implement RSVP authentication is to use static,
preconfigured keys. Static keying can be used with interface based
keys, neighbor based keys or group keys.

However, such static key provisioning is expensive on the operational
side, since no secure automated mechanism can be used, and initial
provisioning as well as key updates require configuration. This
method is therefore mostly useful for small deployments, where key
changes can be carried out manually, or for deployments with
automated configuration tools which support key changes.

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, manually configured keys may be applicable to inter-domain RSVP
authentication.

Since it is not feasible to carry out the key change at the exact
same time on both sides, 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. (Note that
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. Neighbor and Interface Based 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 interface or neighbor based keys. However, existing
key negotiation protocols such as IKEv1 [RFC2409] or IKEv2 [RFC4306]
may not be appropriate in all environments because of the relative
complexity of the protocols and related operations.

4.2.2. Dynamic Group Key Distribution

With this approach, group keys are dynamically distributed among a
set of RSVP routers. For example, [I-D.weis-gdoi-mac-tek] describes
a mechanism to distribute group keys to a group of RSVP speakers,
using GDOI [RFC3547]. In this solution, a key server authenticates
each of the RSVP nodes independently, and then distributes a group
key to the entire group.

5. Specific Cases

5.1. RSVP Notify Messages

[RFC3473] introduces the Notify message and allows such Notify
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. We observe that, alternatively,
in some environments, group keying may allow use of regular RSVP
authentication ([RFC2747]) for protection of non-hop-by-hop Notify
messages. For example, this may be applicable to controlled
environments where nodes invoking notification requests are known to
belong to the same key group as nodes generating Notify messages.

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." We
observe that 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 a 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
section 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. In its security considerations section,
[RFC4875] points out that RSVP message integrity mechanisms for hop-
by-hop RSVP signaling apply to the hop-by-hop P2MP RSVP-TE signaling.
In turn, we observe that the considerations in this document on
keying methods apply equally to P2MP RSVP-TE for the hop-by-hop
signaling.

[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. We note that the observation in section 3.1 of this
document about use of group keying for protection of non-hop-by-hop
messages apply to 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 to protect RSVP. Note that
   [RFC2747] states in section 1.2 that IPsec is not an optimal choice
   to protect RSVP. The key argument is that an IPsec SA and an RSVP SA
   are not based on the same parameters. However, when using group
   keying, IPsec can be used to protect RSVP. The potential issues and
   solutions using group keying are:

   o [RFC2747] specifies in section 4.2, bullet 3, that both the key
      identifier and the sending system address are used to uniquely
      determine the key. In a group keying scenario it would be
      necessary to either store a list of senders to do this, or to not
      use the sending system address to determine the key. Both methods
      are valid, and one of the two approaches must be chosen. The pros
      and cons are beyond the scope of this document.
   o Anti-replay protection in a group keying scenario requires some
      changes to the way [RFC2747] defines anti-replay. Possible
      solutions are discussed in detail in [I-D.weis-gdoi-mac-tek]).
      For example, when using counter-based methods with various senders
      in a single SA, the same counter may be received more than once,
      this conflicts with [RFC2747], which states that each counter
      value may only be accepted once. Time based approaches are a
      solution for group keying scenarios.

   The document "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 standard
   [RFC3547] contains all relevant policy options to secure RSVP with
   IPsec, and no extensions are necessary. An example GDOI policy would
   be to encrypt all packets of the RSVP protocol itself (IP protocol
   46). A router implementing GDOI and IPsec protocols is therefore
   able to implement RSVP encryption.

6.2. Using IPsec ESP

   In both tunnel mode and transport mode, ESP does not protect the
   header (in tunnel mode the outer header). This is an issue with
   group keying when using ESP to secure the RSVP packets: the packet
   header could be modified by a man-in-the-middle attack, replacing the
   destination address with another RSVP router in the network. This
   router will receive the packet, use the group key to decrypt the
   encapsulated packet, and then act on the RSVP packet. This way an
   attacker cannot create new reservations or affect existing ones, but

he can "re-direct" reservations to parts of the network off the
actual reservation path, thereby potentially denying resources to
other applications on that part of the network.

6.3. Using IPsec AH

The INTEGRITY object defined by [RFC2747] provides integrity
protection for RSVP also in a group keying context, as discussed
above. IPsec AH is an alternative method to provide integrity
protection for RSVP. Both methods provide comparable security.

6.4. Applicability of Tunnel Mode

IPsec tunnel mode encapsulates the original packet, prepending a new
IP tunnel header plus an ESP or AH sub-header. The entire original
packet plus the ESP/AH subheader is secured. In the case of ESP the
new, outer IP header however is not cryptographically secured in this
process. This leads to the problem described in Section 6.2. AH
tunnel mode also secures the outer header, and is therefore not
subject to these man-in-the-middle attacks.

Protecting RSVP packets with IPsec tunnel mode works with any of the
above described keying methods (interface, neighbor or group based),
as long as there are no non-RSVP nodes on the path. Note that 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 can not be applied,
because a router upstream a non-RSVP hop does not know the next RSVP
hop, and can thus not apply the correct tunnel header. This is
independent of the key type used.

6.5. 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. [RFC4303] 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.

7. End Host Considerations

Unless RSVP Proxy entities ([I-D.ietf-tsvwg-rsvp-proxy-approaches]

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." 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. INTEGRITY object) as specified in [RFC2747].
For host-based security, interface or neighbor based keys may be
used, however, key management with pre-shared 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 zone of trust as the network
itself, 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 only discusses 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:

   o 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.
   o 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.
   o Aggregation of Resource ReSerVation Protocol (RSVP) Reservations
      over MPLS TE/DS-TE Tunnels [RFC4804]([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.
   o Pre-Congestion Notification (PCN): [I-D.ietf-pcn-architecture]
      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, RSVP signalling 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 Egress PCN-boundary-node will
intercept and process the initial end-to-end Path message. Note
that from the viewpoint of securing end-to-end RSVP, 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 RSVP authentication or IPsec AH) and/ or for RSVP encryption
(e.g., using IPsec ESP in tunnel mode).

	Neighbor/interface based 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 (eg GDOI)

    Table 1: Overview of keying approaches and their applicability

(1): RSVP authentication with group keys works over non-RSVP nodes;
RSVP encryption with IPsec ESP Tunnel mode does not.

We also make the following observations:

o All key types can be used statically, or with dynamic key
   negotiation. This impacts the managability of the solution, but
   not the applicability itself.
o For encryption of RSVP messages IPsec ESP in tunnel mode can be
   used. There is however a security concern, see Section 6.2.

o 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
   a specific security considerations relating to subverted RSVP nodes

10.1. Subverted RSVP Nodes

   A subverted node is defined here as an untrusted node, for example
   because an intruder has gained control over it. 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 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 interface keys, as discussed above.
   Therefore the impact of subverted nodes on the path is comparable,
   independently whether per-interface, per-neighbor or group keys are
   used.

11. Acknowledgements

   The authors would like to thank everybody who provided feedback on
   this document. Specific thanks to Bob Briscoe, Hannes Tschofenig and
   Brian Weis.

12. Changes to Previous Version

   This section provides a change log. It will be removed in the final
   document:

12.1. changes from behringer-00 to behringer-01

   o New section "Applicability to Other Architectures and Protocols":
      Goal is to clarify the scope of this document: The idea presented
      here is also applicable to other architectures
      (PCN[I-D.ietf-pcn-architecture], RFC3175 and RFC4860, etc.
   o Clarified the scope of this document versus RFC4230 (in the
      introduction, last paragraph).
   o Added a section on "End Host Considerations".
   o Expanded section 5.5 (RSVP Encryption) to clarify that GDOI
      contains all necessary mechanisms to do RSVP encrpyion.
   o Tried to clarify the "trust to do what?" question raised by Bob
      Briscoe in a mail on 26 Jul 2007. See the section on trust model.
   o Lots of small editorial changes (references, typos, figures, etc).
   o Added an Acknowledgements section.

12.2. changes from behringer-01 to ietf-00

   o various edits to make it clearer that draft-weis-gdoi-for-rsvp is
      an example of how dynamic group keying could be achieved for RSVP
      and not necessarily the recommended solution

12.3. changes from ietf-00 to ietf-01

   o Significant re-structuring of the entire document, to improve the
      flow, and provide more consistency in various sections.
   o Moved the "Subverted RSVP nodes" discussion into the security
      considerations section.
   o Added a "summary" section.
   o Complete re-write of the old section 5.5 (RSVP encryption), and
      "promotion" to a separate section.
   o Changed reference ID.weis-gdoi-for-rsvp to the new draft ID.weis-
      gdoi-mac-tek
   o in several places, explicitly mentioned "encryption" for RSVP (in
      parallel to authentication).
   o Various minor edits.

12.4. changes from ietf-01 to ietf-02

   o Re-wrote and re-structured the section on IPsec (section 6).
   o Re-wrote the section on RSVP-TE and GMPLS (section 5.2).

o Various editorial changes.

13. Informative References

[I-D.ietf-pcn-architecture]
          Eardley, P., "Pre-Congestion Notification (PCN)
          Architecture", draft-ietf-pcn-architecture-08 (work in
          progress), October 2008.

[I-D.ietf-tsvwg-rsvp-proxy-approaches]
          Faucheur, F., Manner, J., Wing, D., and A. Guillou, "RSVP
          Proxy Approaches",
          draft-ietf-tsvwg-rsvp-proxy-approaches-06 (work in
          progress), October 2008.

[I-D.weis-gdoi-mac-tek]
          Weis, B. and S. Rowles, "GDOI Generic Message
          Authentication Code Policy", draft-weis-gdoi-mac-tek-00
          (work in progress), July 2008.

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

[RFC2409] Harkins, D. and D. Carrel, "The Internet Key Exchange
          (IKE)", RFC 2409, November 1998.

[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.

[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.

[RFC3547] Baugher, M., Weis, B., Hardjono, T., and H. Harney, "The
          Group Domain of Interpretation", RFC 3547, July 2003.

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

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

   [RFC4306] Kaufman, C., "Internet Key Exchange (IKEv2) Protocol",
              RFC 4306, 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.

Authors' Addresses

   Michael H. Behringer
   Cisco Systems Inc
   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 Inc
   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