SPEERMINT Routing Architecture Message Flows
draft-ietf-speermint-flows-02

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on October 23, 2007.

Copyright Notice
Abstract

This draft provides the message flows associated with the SPEERMINT, SIP Peering and Multimedia Interconnect, routing architecture. This document provides examples of many different message flows relative to varying peering scenarios.

Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [1].

Table of Contents

1. Introduction......................................................................3
2. Peering Message flows....................................................6
3. On-Demand Peering........................................................8
   3.1. Transport Layer Security........................................8
   3.2. Proxy Authentication: Subscribe/Notify....................10
4. Static Peering..................................................................11
   4.1. IPSec....................................................................12
   4.2. Co-Location..........................................................12
5. Federation Based Peering.................................................13
   5.1. Simple Federation Match........................................14
   5.2. No federation match..............................................14
   5.3. Federation Referral...............................................15
   5.4. Federation Specific Call Processing..........................16
      5.4.1. Central Federation Proxy..................................17
      5.4.2. VPN Based Federations....................................18
      5.4.3. TLS Based Federation......................................18
6. Media Relay......................................................................18
7. Peering Domain Information Exchange.................................20
   7.1. Domain Routes......................................................20
   7.2. Authentication Credentials.....................................21
8. Peering Message Flow Phases............................................22
   8.1. Discovery Phase..................................................23
   8.2. Policy Exchange Phase.........................................24
   8.3. Security Establishment Phase...................................25
   8.4. Signaling Exchange Phase......................................26
   8.5. Media Exchange Phase..........................................26
9. Security Considerations..................................................27
10. IANA Considerations....................................................27
11. Acknowledgments........................................................28
1. Introduction

This document shows the message flows associated with the most relevant SPEERMINT routing architecture peering scenarios. Most of the message diagrams were based on previous work described within existing IETF standards documents.

The document focuses on the messages exchanged for the purpose of Layer 5 peering [7] between two domains. Messages exchanged for the purpose of setting up SIP sessions within a domain are considered out of scope and were already defined in other IETF documents.

The draft separates the Layer 5 peering scenarios in two major peering scenarios.

- **On-demand**: In this scenario the SIP proxies in domains A and B establish a peering relationship driven by the necessity to deliver a SIP message to another domain. This is sometimes referred as the "email" model.

- **Static**: In this scenario the peering relationship between proxies A and B is statically provisioned independent of the establishment of any SIP session between users in different domains.

Normally, media for a given SIP session follows a different path, traversing a different device (most commonly a router) when crossing peering domains. Alternatively, media for a given session can be directed to traverse the same device used for Layer 5 peering, i.e., the same device that handles signaling when crossing domains. This produces two different models:

- **Decomposed**: In this model SIP proxies perform Layer 5 peering and media is sent directly between the User Agent’s (UA’s) involved in the session. Signaling and media follow different paths.
o Collapsed: In this model the device that performs Layer 5 peering also processes the associated media when crossing domains. In the light of SPEERMINT these devices may need to process media mainly when peering involves SIP entities in private address spaces. This function is usually referred to as media relay and is usually performed by a B2BUA or SBC (Session Border Controller). See [6] for a complete discussion of SBC functions. The decomposed or basic peering model picture is shown below. It is worth mentioning that Proxy 1 and 2 can be separated by any number of layer 3 hops. We will refer to this picture throughout this document.

Figure 1 Basic Peering Picture.
The collapsed model is shown below:

```
                        ..............................          ..............................
                        .                          .          .                            .
                        .                +-------+ .          . +-------+                  .
                        .                |       | .          . |       |                  .
                        .                |  DNS  | .          . | DNS   |                  .
                        .                |   1   | .          . |  2    |                  .
                        .                |       | .          . |       |                  .
                        .                +-------+ .          . +-------+                  .
                        .                    |     .          .     |                      .
                        .                    |     .          .     |                      .
                        .                +-------+ .          . +-------+                  .
                        .                | &     |--------------| &     |                  .
                        .                | other |**************| other |
                        .                | funct | .          . | funct | 
                        .                |       | .          . |       |                      .
                        .                +-------+ .          . +-------+                  .
                        .                |       | .          . |       |                  .
                        .                |       | .          . |       |                  .
                        .                | UA 1  | .          . | UA 2  |                  .
                        .                +-------+ .          . +-------+                  .
                        ..............................          ..............................
```

**Figure 2** Collapsed Peering Picture.

In a decomposed model, the signaling function (SF) and the media function (MF) are implemented in separate entities. A B2BUA is generally on the SIP path in the SF. The vertical control protocol between the SF and MF is out of the scope of this document. The decomposed model is shown below:
2. Peering Message flows

We first depict what we call the basic message flow. The various scenarios differ mostly of how and when peering is implemented. As mentioned earlier peering can be establish following the arrival of a message at a border proxy or statically following an agreement between both domains.
Alice   Proxy 1   DNS   Proxy 2   Bob

<table>
<thead>
<tr>
<th>Peering</th>
<th>Peering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Phase</td>
</tr>
<tr>
<td>[Static]</td>
<td>[On-Demand]</td>
</tr>
</tbody>
</table>

INVITE <---------------------->
100  

NAPTR | NAPTR |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Query</td>
<td>Query</td>
</tr>
</tbody>
</table>

"SIPS+D2T" <----------
SRV | SRV |
| Query | Query |

<--------
SRV Reply | SRV Reply |

<table>
<thead>
<tr>
<th>Peering</th>
<th>Peering</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Phase</td>
</tr>
<tr>
<td>[On-Demand]</td>
<td>[On-Demand]</td>
</tr>
</tbody>
</table>

INVITE <---------------------->
100  

180 <---------------------->
200  

ACK | ACK |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Both Way RTP Media</td>
<td>Both Way RTP Media</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>penno</td>
<td>penno</td>
<td>penno</td>
<td>penno</td>
<td>penno</td>
</tr>
</tbody>
</table>
In the collapsed model, media would follow the path shown below. All other signaling call flows remain the same, except a B2BUA is used instead of a proxy.

```
<table>
<thead>
<tr>
<th>Alice</th>
<th>B2BUA 1</th>
<th>DNS</th>
<th>B2BUA 2</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Both Way RTP Media</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;========&gt;</td>
<td>&lt;======&gt;</td>
<td></td>
<td>&lt;========&gt;</td>
<td></td>
</tr>
</tbody>
</table>
```

The following sections show the message flows in several different scenarios broken in two categories, on-demand and static.

3. On-Demand Peering

In the on demand peering scenario, the relationship between proxies in domains A and B is driven by the arrival of a SIP message at proxy A directed to a user in domain B (or vice-versa).

3.1. Transport Layer Security

In the case this is in fact the first call between those two VSPs, than this call will trigger the establishment of the TLS connection. Otherwise we can assume the TLS connection has been established by some other means.
Alice   Proxy 1       DNS      Proxy 2      Bob
---|------|------|--------|--------|----|
INVITE<------>100<-------->

NAPTR
Query
--------->
NAPTR
Reply
"SIPS+D2T"
<--------
SRV
Query
-------->
SRV
Reply
<--------

Peering
[TLS Connection]
<------------------>
INVITE
-------------->
100
<-------------->
180
<-----------
200
<---------
ACK
--------->
ACK
------------->
Both Way RTP Media
<=======================================>

penno                  Expires October 23, 2007                [Page 9]
TBD: DNS exchange could present proxy 1 with a set of peering policies that need to be met for the peering with proxy 2 too succeed.

3.2. Proxy Authentication: Subscribe/Notify

In the following example message flow, the authentication credentials exchange method may take place before any INVITE is sent by ALICE. The P2Key is sent by Proxy 2’s NOTIFY and is included within subsections of the peering policy event package (PeerPlcyEvtPkg). The P2Key may be stored on Proxy 1 for the duration of the policy subscription. When the subscription expires, the P2Key becomes invalid. At any time before the subscription expires, the P2Key MAY be updated or refreshed as described in [8]. The message flow and authentication exchange may occur in either direction, but for simplicity reasons is only shown unilaterally.
4. Static Peering

In the static peering scenario the relationship between proxies A and B is not driven by a SIP session, but before hand through manual provisioning.
4.1. IPSec

In this model an IPSec connection between proxies A and B is provisioned following an agreement between the two domains.

4.2. Co-Location

In this scenario the two proxies are co-located in a physically secure location and/or are members of a segregated network. In this case messages between Proxy 1 and Proxy 2 would be sent as clear text.
5. Federation Based Peering

The Domain Policy DDDS framework [12] can be used to integrate on-demand peering and static peering into one unified setup. The main idea is that the target can use its domain to publish peering-related information in the DNS. Federations as defined in [13] are one way how source and destination network can find a common set of procedures for the peering.

Federation based peering is thus not a substitute to the various authentication, routing, and QoS procedures which are described in this document.

The following examples demonstrate how Alice can use this scheme to dynamically select the correct peering mechanisms when talking to Bob.

The overall message flow is similar to the one from section 3.1. The DP-DDDS queries the DNS for the same NAPTR records as the algorithm from RFC 3263 [3]. While the originating network behavior according to [3] depends solely on the results retrieved from DNS, the DP-DDDS also uses a set of local configuration options to drive the source network behavior. The following examples thus list both the sender configuration and the answers from the DNS.
5.1. Simple Federation Match

The simplest case is when Alice and Bob share membership in one federation ("http://example.com/Wonderland") which stipulates further call-setup according to section 3.1.

Configuration at Alice’s DNS list Alice’s federations (which includes http://example.com/Wonderland) and rules what do to when a federation is chosen for a call.

NAPTR RRset at Bob’s domain includes:

IN NAPTR 10 50 "u" "D2P+SIP:fed" (!^.*$!http://example.com/small-federation!" . )
IN NAPTR 20 50 "u" "D2P+SIP:fed" (!^.*$!http://example.com/Wonderland!" . )

<table>
<thead>
<tr>
<th>Alice</th>
<th>Proxy 1</th>
<th>DNS</th>
<th>Proxy 2</th>
<th>Bob</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVITE</td>
<td>--------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>100</td>
<td>NAPTR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>NAPTR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Reply</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>NAPTR</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parse D2P+SIP RRs</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Federation match successful</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Parse NAPTR with &quot;SIPS+D2T&quot;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRV</td>
<td>Query</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[Rest according to section 3.1]</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

5.2. No federation match

If Bob does not share a federation with Alice, e.g. by just being a member of the "small-federation", then no direct peering is possible between Alice and Bob.
Bob’s Domain contains:
IN NAPTR 10 50 "u" "D2P+SIP:fed" ("!^.*$!http://example.com/small-federation!" .)

Alice     Proxy 1   DNS     Proxy 2   Bob

INVITE
------>
100
<------
NAPTR
Query
------>
NAPTR
Reply
<-------
Parse D2P+SIP RRs
Federation match
failed.
Bob offers no alternative ways
No peering is possible.

If no matching federations or referrals are found, Alice can either
fall back to PSTN routing or use a transit VSP.

5.3. Federation Referral

If Bob buys transit services from Carol, he can announce this in a
"D2P+SIP" NAPTR record. We now have at Bob’s domain:

IN NAPTR 10 50 "u" "D2P+SIP:fed" ("!^.*$!http://example.com/small-federation!" .)
IN NAPTR 20 50 "u" "D2P+SIP" "" carol.example.com.

If Carol is a member of the Wonderland federation, then we have

$ORIGIN carol.example.com
IN NAPTR 10 50 "u" "D2P+SIP:fed" ("!^.*$!http://example.com/Wonderland!" .)
5.4. Federation Specific Call Processing

The output of the federation matching step in the Domain Policy DDDS application is a federation name and a destination domain (which differs from the original destination domain if referrals were followed).

Federations as defined in [13] can specify their own specific rules on how the actual call-setup is to be performed between two federation members. If Alice is a member of more than one federation.
then Alice’s peering SIP proxy needs to adapt its behavior to the rules of the federation this call is traversing.

The following subsections provide some examples of what a federation could imply for the call processing.

5.4.1. Central Federation Proxy

Federation rules can dictate that calls are to be routed via a federation-maintained central SIP proxy. In that case no further NAPTR/SRV/A lookups are made. Instead, the INVITE will be sent directly via a preconfigured TLS connection to that proxy. This proxy acts as a redirect proxy.

The following message flow provides an example describing this process:

```
Peer Proxy | Federation Proxy | Peer Proxy | Bob

| INVITE               | | | |
| <------------------->| | | |
| 302                 | | | |
| <------------------->| | | |
| ACK                 | | | |
| <------------------->| | | |
| INVITE              | | | |
| <------------------->| 100 | | |
| 180                 | | | |
| <------------------->| 200 | | |
| ACK                 | | | |
| <------------------->| | | |
| Both Way RTP Media  | | | |
| | | | |
| BYE                 | | | |
| <------------------->| 200 | | |
| 200                 | | | |
```
5.4.2. VPN Based Federations

If a federation has established some sort of VPN which connects the SIP elements of all participating VSPs, then matching that federation will cause:

Proxy1 to use e.g. a private DNS within that VPN for further lookups and will direct all further traffic to be routed into that VPN.

IPsec based VPNs are a special case of this.

5.4.3. TLS Based Federation

One of the simplest cases is a TLS based federation.

In that case the federation rules may prescribe the default NAPTR/SRV lookups and only affect the selection of the correct X.509 certificate for the TLS connection.

6. Media Relay

In the event that a calling and/or called entity are part of a private network and the NAT/FW at the CPE is VoIP unaware or the client uses a NAT traversal method, the SIP proxy must find a way to modify the private addresses that remain in the signaling payload (in addition to threading media through the NAT/FW). This modifying process is sometimes referred to as Far-end NAT Traversal (FE-NTRV).

The core of the FE-NTRV process is media relaying. The signaling entity relays media between the two endpoints as a result of the repairing process and to guarantee NAT/FW traversal (symmetric RTP).

It is important to understand that media relay can be use independent of NAT/FW as a way to direct media to a certain device for processing. In the context of SPEERMINT, media relay could be used to enable the collapsed model and/or perform FE-NTRV.
ALICE | NAT/FW | Media Relay | Bob
10.10.1.2 | Signaling:128.16.5.10 | Media:168.12.1.8

| INVITE | INVITE | INVITE |
|------------------> | ------------------> | ------------------> |
s:10.10.1.2:9082 | s:140.1.1.1:23040 | s:128.16.5.10:5060 |
d:128.16.5.10:5060 | d:128.16.5.10:5060 | d:128.16.5.10:5060 |
c= 10.10.1.2 | c= 10.10.1.2 | c= 168.12.1.2 |
m= 11032 | m= audio 11032 | m= audio 3600 |

Media Relay creates a pair of media relay ports. The first port, 3600, is for receiving media from the called party and the 2nd port, 7600, is for receiving media from the calling party. As we do not know what the transport address of the calling party will be (post NAPT), any media received from the called party will be dropped.

| 200 OK | 200 OK | 200 OK |
|<------------------ | <------------------ | <------------------ |
s:128.16.5.10:5060 | s:128.16.5.10:5060 | s:128.16.5.10:5060 |
d:10.10.1.2:9082 | d:140.1.1.1:23040 | d:192.32.6.2:5060 |
c= 168.12.1.8 | c= 168.12.1.8 | c= 192.32.6.2 |
m= audio 7600 | m= audio 7600 | m= audio 9080 |

Media Relay updates remote dest. as 192.32.6.2:9080

| ACK (...) | | |
|------------------> | | |
| | | |

Expires October 23, 2007
7. Peering Domain Information Exchange

7.1. Domain Routes

In some cases, it may be required to exchange specific domain route information between peers. The following describes a method for a relationship between proxies in domains A and B to exchange domain routes using a SIP peering policy event package. This event package may contain specific sections, which will provide routing information for the peering proxy server to update its routing table with new peering routes. This method utilizes a SUBSCRIBE method, and routes may be updated through expiry timers and subscription refreshes as defined in [8].
7.2. Authentication Credentials

In some cases, authorization credentials for authentication methods such as HTTP digest may want to be exchanged and utilized by domain proxies for authenticating new message requests from subscribers intended for a UA in another domain. The following describes a method for a relationship between proxies in domains A and B to exchange authentication information using a SIP peering policy event package. This event package may contain specific sections, which will provide authentication methods to be used for authenticating to the peer’s proxy. This method utilizes a SUBSCRIBE method similar to the method described in section 3.2.
8. Peering Message Flow Phases

The message flow phases are Discovery, Policy Exchange, Security Establishment, Signaling Exchange, and Media Exchange. The following flow provides an overview of the phases. Each of the phases is described individually in the following subsections. In the following flow, the policy and peering proxy have been combined; however, these two functions may be separated. Also, the signaling and media exchange phase descriptions have been omitted for clarity purposes, because their functionality has not changed for the purposes of peering. However, they have been explained further in the following subsections.
8.1. Discovery Phase

The first phase of static or dynamic peering requests is discovery. The discovery process can be summarized by querying the Location Function to determine the next phase in the message flow. The discovery phase can take place via a local or external federation location function. Examples of the function may be comprised of an ENUM/DNS or redirect server. After the discovery phase has completed, the peering process will progress to a subsequent phase, usually the policy or authentication phase. The following message flows provide examples of the discovery phase.

Discovery phase utilizing an ENUM/DNS server as a location function:

<table>
<thead>
<tr>
<th>Alice</th>
<th>Peer Proxy</th>
<th>DNS</th>
<th>Peer Proxy</th>
</tr>
</thead>
<tbody>
<tr>
<td>INVITE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>100 Trying</td>
<td>180 Ringing</td>
<td></td>
</tr>
<tr>
<td>180 Ringing</td>
<td>200OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td>200OK</td>
<td>ACK</td>
<td>ACK</td>
<td></td>
</tr>
<tr>
<td>ACK</td>
<td></td>
<td>Both Way RTP Media</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>penno</td>
<td>Expires October 23, 2007</td>
<td>[Page 23]</td>
<td></td>
</tr>
</tbody>
</table>
8.2. Policy Exchange Phase

Since the originating peer proxy does not know if the destination AOR is a PF or a SF, it must progress with a normal dialog request with the assumption it is a SF. In the event a request fails due to an authentication failure (401 Unauthorized), and no known authentication credentials exist or no longer appear to be working, the requesting proxy may issue a SUBSCRIBE [8] request to the attempted peer’s AOR received through the discovery phase. The SUBSCRIBE request should be a request to attain a, currently, undefined peering policy event package. In some cases, the requesting proxy already knows it must attain the peering policy event package, and may forego the initial INVITE attempt and issue a SUBSCRIBE request instead. Once this phase is completed, after extracting and following any specific received policies, the authentication phase is attempted as the policy permits or requires. The following message flow provides an example of the policy exchange phase. The following message flow assumes the discovery phase has already completed using one of the methods described in section 12.1.
8.3. Security Establishment Phase

The security establishment phase follows the described methods in previous sections of this document. After the originating proxy receives the policy event package, it extracts the necessary security policy information. The security policy may contain many different combinations of security requirements. For example, it may contain a simple digest authentication method or may require TLS with digest authentication. This is determined by the destination peer, and must be followed to successfully complete this phase. This phase follows standard methods described in [2], so the following flow provides an example of this phase, but does not incorporate all possibilities. This phase assumes the previous phases were successfully completed or purposefully omitted per peering implementation.
8.4. Signaling Exchange Phase

The signaling exchange phase is a necessary step to progress towards establishing peering. This phase may incorporate the security exchange phase, but it is not required. This phase follows standard methods described in [2], so the following flow provides an example of this phase, but does not incorporate all possibilities.

\[
\begin{array}{c|c|c}
\text{Peer Proxy} & \text{Signaling Exch. Phase} & \text{Peer Proxy} \\
\hline
\text{INVITE} & \text{[TLS Connection]} & \text{INVITE} \\
\hline
\text{401 Unauthorized} & \text{INVITE} & \text{} \\
\hline
\text{100 Trying} & \text{180 Ringing} & \text{} \\
\hline
\text{200 OK} & \text{ACK} & \text{BYE} \\
\hline
\text{200 OK} & \text{} & \text{} \\
\hline
\end{array}
\]

8.5. Media Exchange Phase

The media exchange phase is negotiated and established during the signaling exchange phase. This phase follows standard methods described in [2], so the following flow provides an example of this phase, but does not incorporate all possibilities.

\[
\begin{array}{c|c|c|c}
\text{Alice} & \text{Peer Proxy} & \text{Peer Proxy} & \text{Bob} \\
\hline
\text{INVITE} & \text{INVITE} & \text{[TLS Connection]} & \text{[TLS Connection]} \\
\hline
\text{100 Trying} & \text{401 Unauthorized} & \text{} & \text{} \\
\hline
\text{200 OK} & \text{} & \text{} & \text{} \\
\hline
\end{array}
\]
9. Security Considerations

The level of security required during the establishment and maintenance of a SIP peering relationship between two proxies can vary greatly. In general all security considerations related to the SIP protocol are also applicable in a peering relationship.

If the two proxies communicate over an insecure network, and consequently are subject to attacks, the use of TLS or IPSec would be advisable.

If there is physical security and the proxies are co-located, or the proxies are situated in a segregated network (such as a VPN), one could argue that basic filtering based on IP address is enough.

10. IANA Considerations

N/A
11. Acknowledgments

Thanks to Otmar Lendl for the federation call flows.

12. References

12.1. Normative References


[6] ETSI TS 102 333: "Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Gate control protocol".


12.2. Informative References


Author’s Addresses

Daryl Malas
Level 3 Communications LLC
1025 Eldorado Blvd.
Broomfield, CO 80021
USA
EMail: daryl.malas@level3.com

Sohel Khan, Ph.D.
Comcast Cable Communications
U.S.A
Email: sohel_khan@cable.comcast.com

Reinaldo Penno
Juniper Networks
1194 N Mathilda Avenue
Sunnyvale, CA
USA
Email: rpenno@juniper.net

Adam Uzelac
Global Crossing
1120 Pittsford Victor Road
PITTSFORD, NY 14534
USA
Email: adam.uzelac@globalcrossing.com
Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY, THE IETF TRUST AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The IETF Trust (2007).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

Acknowledgment

Funding for the RFC Editor function is currently provided by the Internet Society.