Network Working Group
Internet-Draft
Expires: December 30, 2002

M. Blanchet
Viagenie

July 1, 2002

IPv6 over IPv4 profile for Tunnel Setup Protocol (TSP)
draft-vg-ngtrans-tsp-v6v4profile-01.txt

Status of this Memo

Copyright Notice

Abstract

This document proposes a tunnel profile to setup IPv6 over IPv4 tunnels to be used with the Tunnel Setup Protocol (TSP) [8].

Table of Contents

1. Rationale for an IPv6 tunnel setup protocol

Many IPv6 transition techniques uses tunnelling to overlay an IPv6 network over an IPv4 network. Some are manual, some are automatic like 6to4 by embedding the IPv4 address of the gateway in the IPv6 address, some are semi-automatic like the tunnel broker.

The operation of the protocol defined in this document, known as Tunnel Setup Protocol, allows dual stack (IPv4/IPv6) nodes to negotiate the establishment of a configured tunnel (IPv6 over IPv4) to a Tunnel Broker according to the IPv6 Tunnel Broker model proposed in RFC 3053 [1]

The protocol solves the problem of authentication and the negotiation between any dual stack node and Tunnel Broker by proposing a set of messages to be exchanged between nodes and Tunnel Brokers. Moreover, it enables the two parties to negociate prefix, dns delegation and routing info.

2. Terminology

Tunnel Broker (TB) In a Tunnel Broker model, the broker is taking charge of all communication between Tunnel Servers (TS) and Tunnel Clients (TC). Tunnel clients query brokers for a tunnel and the broker find a suitable tunnel server, ask the Tunnel server to setup the tunnel and send the tunnel information to the Tunnel Client.

Tunnel Server (TS) Tunnel Servers are providing the specific tunnel service to a Tunnel Client. It can reveive the tunnel request from a Tunnel Broker (as in the Tunnel Broker model) or directly from the Tunnel Client as in the Tunnel Setup Protocol option.

Tunnel Client (TC) The Tunnel Client is the entity that need a tunnel for a particular service or connectivity. A Tunnel Client can be a host or a router.

3. Why a Tunnel Setup Protocol

There are current proposals about deploying configured tunnels over IPv4 network. The Tunnel Broker method (RFC3053) [1] intends to use Web browers and servers to allow end-users to request configured tunnel but there is no real negociation between end-user and Tunnel Broker. If end-users use dynamic IPv4 addresses, a manual operation must be done to update the Tunnel Broker. This manual operation implies to be done over a security layer to ensure a secure authentication of end-users.

The IPv6 over IPv4 tunnels for home to Internet access method [5] is proposing a secure method to solve the problem of dynamic IP address changes at end-users sides by using neighbor discovery protocol [2] functions and IPsec. This proposed method is dependant of IPsec implementors that have to modify their implementations to handle virtual interfaces for IPv6.

A Tunnel Broker implementation with a web interface revealed many practical problems :

- Using Web interfaces for Tunnel Broker limits the scalability of deploying IPv6 networks and hosts at very large scale over Internet. Web interface requires manual operation from end-users.

- End-users that uses dynamic IPv4 addresses must go back manually to the Tunnel Broker's web interface each time their IPv4 address changes

The Tunnel Setup Protocol (TSP) approach proposes a negociation of tunnel parameters between Tunnel clients and Tunnel Servers. The proposed protocol is able to handle different kinds of tunnel over IPv4 such as IPv6 configured tunnel, DVMRP tunnels over IPv4 for multicast and others. In the current document, examples of the protocol are focused on IPv6 configured tunnel.

4. The IPv6 over IPv4 tunnel profile

4.1 Overview

This profile uses the included DTD for the xml format of the message. The dtd contains the description of the tunnel XML message. This message is used by the TSP compliant server to provide IPv6 over tunnels service. Action for the specified tunnel is provided in the 'action' atribute of the 'tunnel' message. Valid actions for this profile are : 'create', 'info' and 'delete'.

The 'create' action is used to request a new tunnel or update an existing tunnel. The 'info' action is used to request current properties of an existing tunnel. The 'delete' action is used to remove an existing tunnel from the server.

The 'tunnel' message contains three elements:

client Client's information

server Server's information

broker List of other server's

4.2 Client element

The client element contains 2 elements: 'address' and 'router'. These elements are used to describe the client needs and will be used by the server to create the appropriate tunnel. This is the only element sent by a client.

The 'address' element is used to identify the client IPv4 endpoint of the tunnel. The client MUST send only an IPv4 address to the server. The server will then return the IPv6 address endpoint and domain name inside the 'client' element when the tunnel is created or updated.

Optionaly a client can send a 'router' element to ask for a prefix delegation. The 'router' element contains the 'router protocol' attribute which specify the routing method to be use between the server and the client. If no attribute is specified the the routing will use static routes. Routing method may include 'rip' or 'bgp'. If 'bgp' is used, the client MUST sent a valid AS number within the 'as' element.

4.3 Server element

The 'server' element contains 2 elements: 'address' and 'router'. These elements are used to describe the server's tunnel endpoint. The 'address' element is used to provide both IPv4 and IPv6 addresses of the server's tunnel endpoint, while the 'router' element provides information for the routing method choosen by the client.

4.4 broker element

The 'broker' element is used by a server to provide a alternate list of servers to a client in the case where the server is not able to provide the requested tunnel.

The 'broker' element will contain a series of 'address' element.

5. Tunnel request

This section presents multiple examples of requests.

5.1 Host Tunnel request and Reply

A simple tunnel request consist of a 'tunnel' element which contains only an 'address' element

Simple tunnel request made by a client.

```
      -- Successful TCP Connection --
      C:VERSION=1.0 CR LF
      S:CAPABILITY TUNNEL=V6V4 AUTH=ANONYMOUS CR LF
      C:AUTHENTICATE ANONYMOUS CR LF
      S:200 Authentication successful CR LF
      C:Content-length: 123 CR LF
        <tunnel action="create" type="v6v4">
           <client>
               <address type="ipv4">1.1.1.1</address>
           </client>
        </tunnel> CR LF
      S: Content-length: 234 CR LF
         200 OK CR LF
         <tunnel action="info" type="v6v4" lifetime="1440">
           <server>
              <address type="ipv4">206.123.31.114</address>
              <address type="ipv6">3ffe:b00:c18:ffff:0000:0000:0000:0000</
address>
           </server>
           <client>
              <address type="ipv4">1.1.1.1</address>
              <address type="ipv6">3ffe:b00:c18:ffff::0000:0000:0000:0001</
address>
              <address type="dn">userid.domain</address>
           </client>
         </tunnel> CR LF
```

5.2 Router Tunnel request with a /48 prefix delegation, and reply

A tunnel request with prefix consist of a 'tunnel' element which contains 'address' element and a 'router' element.

Tunnel request with prefix and static routes.

```
C: Content-length: 234 CR LF
   <tunnel action="create" type="v6v4">
    <client>
     <address type="ipv4">1.1.1.1</address>
     <router>
      <prefix length="48"/>
      <dns_server>
       <address type="ipv4">2.3.4.5</address>
       <address type="ipv4">2.3.4.6</address>
       <address type="ipv6">3ffe:0c00::1</address>
      </dns_server>
```

```
        </router>
       </client>
```

```
      </tunnel> CR LF
   S: Content-length: 234 CR LF
      200 OK CR LF
      <tunnel action="info" type="v6v4" lifetime="1440">
       <server>
        <address type="ipv4">206.123.31.114</address>
        <address type="ipv6">3ffe:b00:c18:ffff:0000:0000:0000:0000</address>
       </server>
       <client>
        <address type="ipv4">1.1.1.1</address>
        <address type="ipv6">3ffe:b00:c18:ffff::0000:0000:0000:0001</address>
        <address type="dn">userid.domain</address>
        <router>
         <prefix length="48">3ffe:0b00:c18:1234::</prefix>
         <dns_server>
          <address type="ipv4">2.3.4.5</address>
          <address type="ipv4">2.3.4.6</address>
          <address type="ipv6">3ffe:0c00::1</address>
         </dns_server>
        </router>
       </client>
      </tunnel> CR LF
```

5.3 Router Tunnel Request with a /48 prefix delegation and RIP routing, and Reply

A tunnel request with prefix consist of a 'tunnel' element which contains 'address' element and a 'router' element.

Tunnel request with prefix and RIP routing.

```
   C: Content-length: 234 CR LF
      <tunnel action="create" type="v6v4">
       <client>
        <address type="ipv4">1.1.1.1</address>
        <router protocol="rip">
         <prefix length="48"/>
         <dns_server>
          <address type="ipv4">2.3.4.5</address>
          <address type="ipv4">2.3.4.6</address>
          <address type="ipv6">3ffe:0c00::1</address>
         </dns_server>
        </router>
       </client>
      </tunnel> CR LF
   S: Content-length: 234 CR LF
```

```
200 OK CR LF
<tunnel action="info" type="v6v4" lifetime="1440">
 <server>
  <address type="ipv4">206.123.31.114</address>
  <address type="ipv6">3ffe:b00:c18:ffff:0000:0000:0000:0000</address>
 </server>
 <client>
  <address type="ipv4">1.1.1.1</address>
  <address type="ipv6">3ffe:b00:c18:ffff::0000:0000:0000:0001</address>
  <address type="dn">userid.domain</address>
  <router protocol="rip">
   <prefix length="48">3ffe:0b00:c18:1234::</prefix>
   <dns_server>
    <address type="ipv4">2.3.4.5</address>
    <address type="ipv4">2.3.4.6</address>
    <address type="ipv6">3ffe:0c00::1</address>
   </dns_server>
  </router>
 </client>
 </tunnel>
```

5.4 Router Tunnel Request with a /48 prefix delegation and BGP peering, and Reply

A tunnel request with prefix consist of a 'tunnel' element which contains 'address' element and a 'router' element.

Tunnel request with prefix and BGP peering.

```
C: Content-length: 234
   <tunnel action="create" type="v6v4">
     <client>
      <address type="ipv4">1.1.1.1</address>
      <router protocol="bgp">
       <prefix length="48"/>
       <as number="12345"/>
       <dns_server>
        <address type="ipv4">2.3.4.5</address>
        <address type="ipv4">2.3.4.6</address>
        <address type="ipv6">3ffe:0c00::1</address>
       </dns_server>
      </router>
     </client>
    </tunnel>

S: Content-length: 234
   200 OK
```

```
<tunnel action="info" type="v6v4" lifetime="1440">
 <server>
  <address type="ipv4">206.123.31.114</address>
  <address type="ipv6">3ffe:b00:c18:ffff:0000:0000:0000:0000</address>
  <router protocol="bgp">
   <as number="23456"/>
  </router>
 </server>
 <client>
  <address type="ipv4">1.1.1.1</address>
  <address type="ipv6">3ffe:b00:c18:ffff::0000:0000:0000:0001</address>
  <address type="dn">userid.domain</address>
  <router protocol="bgp">
   <prefix length="48">3ffe:0b00:c18:1234::</prefix>
   <as number="12345"/>
   <dns_server>
    <address type="ipv4">2.3.4.5</address>
    <address type="ipv4">2.3.4.6</address>
    <address type="ipv6">3ffe:0c00::1</address>
   </dns_server>
  </router>
 </client>
</tunnel>
```

6. Error codes

This profile dependant error codes are :

501 Invalid IPv4 address

502 Invalid or duplicate nicname

503 Invalid AS number

504 Router function not supported

505 No more tunnels available

506 IPv4 prefix already used for existing tunnel

507 Requested prefix length cannot be assigned

508 Routing protocol setup error

509 DNS delegation setup error

514 Protocol error

517 Unsupported router protocol

518 Unsupported prefix length

519 Invalid as number

520 Missing prefix length

if a list of tunnel servers is following the error code as a referal service, then 1000 is added to the error code.

7. IANA Considerations

The TUNNELTYPE "v6v4" is registered for this document.

8. Security considerations

This protocol is also in accordance with guidelines for IPv6 transition [6] about possible abuse against IPv6 transition technologies.

9. Acknowledgements

Jun-Ichiro Itojun Hagino was, as usual, a great helper in refining and commenting this work.

This work has been done on a team basis so all people here contributed to the original work: Andre Cormier, Regis Desmeules, Florent Parent, Jocelyn Picard, Guy Turcotte.

References

[1] Durand, A., Fasano, P., Guardini, I. and D. Lento, "IPv6 Tunnel Broker", RFC 3053, January 2001.

[2] Narten, T., Nordmark, E. and W. Simpson, "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[3] Myers, J., "Simple Authentication and Security Layer (SASL)", RFC 2222, October 1997.

[4] Leach, P. and C. Newman, "Using Digest Authentication as a SASL Mechanism", RFC 2831, May 2000.

[5] Durand, A., "IPv6 over IPv4 tunnels for home to Internet access method", July 2000.

[6] Hagino, J., "Possible abuse against IPv6 transition technologies", July 2000.

[7] 2, 1., "MIME-type extension for IPv6 configured tunnels", 1 1.

[8] Blanchet, M., "Tunnel Setup Protocol", July 2001.

Author's Address

Marc Blanchet
Viagenie
2875 boul. Laurier, bureau 300
Sainte-Foy, QC G1V 2M2
Canada

Phone: +1 418 656 9254
EMail: Marc.Blanchet@viagenie.qc.ca
URI: http://www.viagenie.qc.ca/

Appendix A. IPv6 over IPv4 tunnel DTD

DTD

```
<?xml version="1.0"?>

<!DOCTYPE tunnel  [

<!ELEMENT tunnel        (server?,client?,broker?)>

  <!ATTLIST tunnel action   (create|info|list) #REQUIRED >
  <!ATTLIST tunnel type     (v6v4|broker)      #REQUIRED >
  <!ATTLIST tunnel lifetime CDATA              "1440"    >

<!ELEMENT server        (address+,router?)>

<!ELEMENT client        (address+,router?)>

<!ELEMENT broker        (adress+)>

<!ELEMENT router        (prefix?,dns_server?,as?)>
  <!ATTLIST router protocol (rip|bgp) "">

<!ELEMENT dns_server    (address+)>

<!ELEMENT as EMPTY>
  <!ATTLIST as number CDATA #REQUIRED>

<!ELEMENT prefix        (#PCDATA)>
  <!ATTLIST prefix length CDATA #REQUIRED>

<!ELEMENT address       (#PCDATA)>
  <!ATTLIST address type (ipv4|ipv6|dn) #REQUIRED>

]>
```

Full Copyright Statement

Acknowledgement

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