Internet Draft
Expiration Date: November 2002

R. Bonica
WorldCom
K. Kompella
Juniper Networks
D. Meyer
Cisco Systems
June 2002

Tracing Requirements for Generic Tunnels
draft-bonica-tunneltrace-03.txt

1. Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of [RFC-2026].

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.

2. Abstract

This document specifies requirements for a generic route tracing application. The application must provide all functionality that "traceroute" [RFC 2151] currently provides. It also must provide enhanced capabilities with regard to tunnel tracing (e.g., tracing through IP-in-IP tunnels, tracing through MPLS LSPs).

3. 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].

4. Introduction

Currently, the IETF supports the following tunneling technologies:

Generic Routing Encapsulation (GRE)
Multiprotocol Label Switching (MPLS)
IP over Optical (IPO)
IP Security Protocol (IPSEC)
IP in IP
Layer 2 Tunneling Protocol (L2TP)

Although these tunneling technologies provide operators with many useful features, they also present management challenges. Operators require a generic route tracing application that they can use to verify tunnel paths and diagnose tunnel faults.

This document specifies requirements for that generic route tracing application. It also specifies requirements for a protocol that will support that application.

5. Review of Existing Functionality

Currently, network operators use "traceroute" to identify the path toward any destination in an IP network. Section 3.4 of [RFC-2151] provides a thorough description of traceroute. Although traceroute is very reliable and very widely deployed, it is deficient with regard to tunnel tracing.

Depending upon tunnel type, traceroute may display an entire tunnel as a single IP hop, or it may display a tunnel as a collection of IP hops, without indicating that they are part of a tunnel.

For example, assume that engineers are using IP tunnels in an IP network. Assume also that they configure a tunnel so that the head-end router does not copy the TTL value from the inner IP header to outer IP header. Instead, the head-end router always sets the outer TTL value to its maximum permitted value. When engineers trace routes through the network, traceroute will always display the tunnel as a single IP hop, hiding all components except the tail-end interface.

Now assume that engineers are using MPLS to support an IP network. Assume also that engineers configure an MPLS LSP so that the ingress router propagates the TTL value from the IP header to the MPLS header. When engineers trace routes through the network, traceroute will display the LSP as a series of IP hops, without indicating that they are part of a tunnel.

6. Application Requirements

Network operators require a new route-tracing application. The new application must provide all functionality that traceroute currently provides. It also must provide enhanced tunnel tracing capabilities.

The following list provides specific requirements for the new route-tracing application:

1) Support the notion of a security token as part of the tunnel trace request. The security token identifies the tracer's privileges in tracing tunnels. Network elements will use this security token to determine whether or not to return the requested information to the tracer. In particular, appropriate privileges are required for items (2), (3), (5), (8), (12), and (13).

2) Support in-line traces. An in-line trace reveals the path between the host upon which the route-tracing application executes and any interface in an IP network.

3) Support third party traces. A third party trace reveals the path between any two points in an IP network. The application that initiates a third party trace need not execute upon a host or router that is part of the traced path.

4) When tracing through a tunnel, either as part of an in-line trace or a third party trace, display the tunnel either as a single IP hop or in detail. The user's request determines how the application displays tunnels, subject to the user having permission to do this.

5) When displaying a tunnel in detail, include the tunnel type (e.g., GRE, MPLS), the tunnel name (if applicable) and the tunnel identifier (if applicable). Also, include tunnel components and round trip delay across each component.

6) Support the following tunneling technologies: GRE, MPLS, IPSEC, IP/O, IP-in-IP, L2TP. Be easily extensible to suppport new tunnel technologies.

7) Trace through nested, heterogeneous tunnels (e.g., IP-in-IP over MPLS).

8) At the users request, trace either through the forwarding plane or the control plane.

9) Support control plane tracing for all tunnel types. When tracing through the control plane, the device at the head-end of a hop reports hop details.

10) Support tracing through forwarding plane for all tunnel types that implement TTL decrement (or some similar mechanism). When tracing through the forwarding plane, the device at the tail-end of a hop reports hop details.

11) Support tracing through the forwarding plane for all tunnel types that implement TTL decrement, regardless of whether the tunnel engages in TTL propagation. (That is, support tunnel tracing regardless of

whether the TTL value is copied from an inner header to an outer header at tunnel ingress).

12) When tracing through the control plane, display the MTU associated with each hop.

13) When tracing through the forwarding plane, display the MTU associated with each hop in the reverse direction.

7. Protocol Requirements

Implementers require a new protocol that supports the application described above. This protocol reveals the path between two points in an IP network. When access policy permits, the protocol also reveals tunnel details.

7.1. Information Requirements

The protocol elicits a series of traceResponse messages. Each traceResponse message represents a hop that connects the head-end of the traced path to the tail-end of the traced path. A hop can be either a top-level IP hop or lower-level hop that is contained by a tunnel.

The protocol also supports a traceProbe message. Each traceProbe message elicits exactly one traceResponse message.

7.2. Transport Layer Requirements

UDP carries traceProbe and traceResponse messages to their destinations.

7.3. Routing Requirements

The device that hosts the route tracing application must maintain an IP route to the head end of the traced path. It must also maintain an IP route to the head end of each tunnel for which it is requesting tunnel details. The device that hosts the tunnel tracing application need not maintain a route to any other device that supports the traced path.

All of the devices mentioned above must maintain an IP route back to the device that hosts the route tracing application.

In order for the protocol to provide tunnel details, all devices contained by a tunnel must maintain an IP route to the device that hosts the tunnel ingress.

7.4. Maintaining State

The protocol must be stateless.

8. Security Considerations

A configurable access control policy determines the degree to which features described herein are delivered. The access control policy requires user identification and authorization.

As stated above, the new protocol must not introduce security holes nor consume excessive resources (e.g., CPU, bandwidth). It also must not be exploitable by those launching DoS attacks.

9. References

[RFC-2026], Bradner, S., "Internet Standards Process Revision 3", RFC 2026, Harvard University, October 1996.

[RFC-2119], Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, Harvard University, March 1997

[RFC-2151], Kessler, G., Shepard, S., A Primer On Internet and TCP/IP Tools and Utilities, RFC 2151, Hill Associates, Inc., June 1997

[RFC-2434] T. Narten and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", RFC 2434, October, 1998.

[RFC-2637] Hamzeh, K. et. al., "Point-to-Point Tunneling Protocol (PPTP)", RFC 2637, July, 1999.

10. Acknowledgements

Thanks to Randy Bush and Steve Bellovin for their comments.

11. Author's Addresses

Ronald P. Bonica
WorldCom
22001 Loudoun County Pkwy
Ashburn, Virginia, 20147
Phone: 703 886 1681
Email: rbonica@mci.net

Kireeti Kompella
Juniper Networks, Inc.
1194 N. Mathilda Ave.
Sunnyvale, California 94089
Email: kireeti@juniper.net

Dave Myers
Cisco Systems
170 Tasman Drive
San Jose, California 94025
Email: dmm@cisco.com

12. Full Copyright Statement