Abstract

This document specifies extensions to traceroute and ping techniques to facilitate additional application information to be carried in UDP, TCP and ICMP traceroute probe messages and ICMP echo request and reply messages. This proposal also allows the receiver to authenticate the source of the traceroute and ping senders.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

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."

This Internet-Draft will expire on May 3, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Specification of Requirements ............................ 3
2. Introduction ............................................. 3
3. Motivation ............................................... 3
4. Trace-Ping Message Extension ............................. 4
   4.1. Trace-Ping Extension Structure ...................... 4
   4.1.1. Trace-Ping Common Header ......................... 5
   4.1.2. Trace-Ping Object ................................ 5
   4.1.2.1. Trace-Ping Authentication Object .............. 6
   4.1.2.2. Trace-Ping Information-Request Object ....... 7
   4.2. Trace-Ping Extension Offset Field .................. 8
   4.2.1. UDP Messages ....................................... 8
   4.2.2. TCP Messages ....................................... 9
   4.2.3. ICMP Messages .................................... 10
   4.2.4. Implementation Discussion ....................... 10
5. Implementation and Operation Considerations ............. 11
   5.1. Traceroute and Ping Probe Sender .................... 11
   5.2. Traceroute and Ping Probe Receiver .................. 11
6. Security Considerations .................................. 12
7. IANA Considerations ..................................... 12
8. Acknowledgements ........................................ 13
9. References ............................................... 13
   9.1. Normative References ............................... 13
   9.2. Informative References ............................. 13
Authors’ Addresses .......................................... 14
1. Specification of Requirements

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 [RFC2119].

2. Introduction

Traceroute and Ping are two of commonly used tools in the diagnosis of network problems. This document proposes the mechanism by which the traceroute probe messages and ICMP echo request/reply messages can be extended to include other user information various applications may want to include; and it can be optionally authenticated by the receiving node(s). These mechanisms are intended for network operators to perform more secured network management and troubleshooting tasks while using traceroute and ping tools. The changes proposed in this document are backward compatible (with the existing traceroute and ping tools) and applicable to both IPv4 and IPv6 networks.

The mechanisms specified in this document apply to the following traceroute and ping probe protocols: UDP [RFC0768], TCP [RFC0793], and ICMP/ICMPv6 [RFC0792] [RFC4443]. This mechanism also applies to the ICMP/ICMPv6 echo reply messages [RFC0792].

This document defines an extension for traceroute and ping probe messages to optionally include authentication signature object. The intermediate and destination nodes can authenticate the sender of the traceroute or ping packet before providing the requested information in the ICMP response. This document also defines an optional Information-Request Object for the traceroute/ping extension. This Object specifies the types of information the sender expects to be included in the traceroute/ping response (i.e., in the ICMP message elicited by the traceroute/ping packet and generated by the intermediate or destination node or nodes).

Other applications can define their own Trace-Ping objects using this extension.

3. Motivation

The current traceroute or ping has no defined mechanism to include application data on the sender side, or to include application data in the ICMP echo reply on the receiver side. Although the [RFC4884] has defined the multi-part message extension in ICMP, it is applied only to the ICMP type 3, 11 and 12 for traceroute reply messages.
Those mechanisms are not applied to traceroute probe messages or ICMP echo request/reply messages.

For security concerns of traceroute or ping packets, one may employ a rudimentary control mechanism to limit the trusted senders by defining on every router the access control lists specifying source addresses of the traceroute and ping message, such mechanism is deemed configuration intensive, static, and error-prone. Moreover, such mechanism would be susceptible to address spoofing. Additionally, such mechanism does not provide the sender with dynamic control of the different kind of extensions to be requested.

The ICMP reply messages has been extended to support multi-part message inside ICMP [RFC4884] for some ICMP types. Some of the applications [RFC5837] [RFC4950] [I-D.shen-icmp-routing-inst] are designed mainly for internal network troubleshooting by network operators. Network providers may want to limit those applications only to trusted senders of traceroute/ping probes due to security or policy reasons by using this mechanism described in this document.

Other applications, for example the TRILL-OAM [I-D.tissa-trill-oam] can use this scheme to extend their OAM application using ICMP echo request and reply for data center troubleshootings.

4. Trace-Ping Message Extension

This proposed extension is to define a Trace-Ping data structure which resides within UDP/TCP/ICMP data field; and to reserve the lowest 4 bits inside a 16-bit field within UDP/TCP/ICMP headers to indicate the extension structure offset location.

In most of the traceroute implementation, there is some private data in probe messages used by traceroute applications. With this "extension offset" defined, the applications can continue to use those private data while supporting this Trace-Ping extension in a deterministic way. This extension applies to both traceroute and ping applications.

4.1. Trace-Ping Extension Structure

The Trace-Ping structure starts in UDP/TCP/ICMP data field location from 0 to 56 octets specified in the Trace-Ping "extension offset", see Section 4.2, in 32-bit boundary. It MUST have exactly one Trace-Ping common header followed by zero or more Trace-Ping Objects.
4.1.1. Trace-Ping Common Header

The Common Header is a 8 octets structure has the following format:

```
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--------------------------------------------------+
| Version | Length | Checksum |
+--------------------------------------------------+
| 0x54726163 | 32 bits. It is defined as Hex value of 0x54726163 in this document. This is used mainly for structure identification of this extension version.
+--------------------------------------------------+
```

The fields of the Common Header are defined as follows:

Version: 4 bits. It is defined as 1 in this document.
Length: 12 bits. The total length of the Trace-Ping data structure specifying number of 32-bit words (includes the common header and all the Objects).
Checksum: 16 bits. The one’s complement of the one’s complement sum of the Trace-Ping data structure, with the checksum field replaced by zero for the purpose of computing the checksum.

4.1.2. Trace-Ping Object

Trace-Ping Object have the following format:

```
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+--------------------------------------------------+
| Length | Class-Num | C-Type |
+--------------------------------------------------+
| 0x54726163 | 32 bits. It is defined as Hex value of 0x54726163 in this document. This is used mainly for structure identification of this extension version.
+--------------------------------------------------+
```

Length: 16 bits. Length of object, measured in octets, including the object header and object payload.

Class-Num: 8 bits. Identifies ICMP Trace-Ping object class.

C-Type: 8 bits. Identifies ICMP Trace-Ping object sub-type.

All the Trace-Ping Objects are optional. This document defines two Trace-Ping Objects below.

4.1.2.1. Trace-Ping Authentication Object

This Object carries the HMAC authentication related information. It verifies both the data integrity and the authenticity of the entire message. This Object has the following format:

```
0                   1                   2                   3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|             Length            |   Class-Num   |     C-Type    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Auth Type                 |    Key ID     | Auth Data Len |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
|                     Auth Data (Variable)                      |
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+ ...
```

Length: Variable, in octets.

Class-Num: IANA allocation from ICMP Trace-Ping extension registry.

C-Type: 1

Auth Type: 16 bits. The following values are proposed:

* Type=0 signifies no authentication.

* Type=1 signifies simple password based authentication.

* Type=2 signifies Cryptographic authentication.

Please note that the above type values are in line with IANA allocated values for other protocols (e.g., OSPF).
Key ID:  8 bits. This allows multiple secret keys to be active simultaneously. Using Key IDs makes the key rollover convenient. Each secret key must be associated with the hash algorithm. This may be done through provisioning on each node.

Auth Data Len:  8 bits. This specifies the length of the authentication data (and allows for the support of current and future authentication schemes).

Auth Data: Variable length. This field carries the result (e.g., HMAC code) of the HMAC algorithm applied over the entire traceroute/ping IP/IPv6 packet. When the Auth data is calculated, the shared key is stored in this field, and the checksum fields in the IP header, UDP/TCP/ICMP header and Trace-Ping common header are set to zero. The result of the algorithm is placed in the Auth Key field. The following lists algorithms that could be commonly supported:

* HMAC-MD5
* HMAC-SHA1
* HMAC-SHA2 variants (e.g., 224, 256, 384, 512, etc.)

At least HMAC-MD5 and HMAC-SHA1 algorithms should be supported on all the nodes compliant with this specification.

4.1.2.2. Trace-Ping Information-Request Object

The Information-Request Object has the following format:

```
+---------------+---------------+---------------+---------------+
| Length         | Class-Num     | C-Type        |
|                |               |               |
+---------------+---------------+---------------+
| Info Request  |               |               |
+---------------+---------------+---------------+

Length:  8

Class-Num: IANA allocation, the same Class-Num value as in Section 4.1.2.1.
C-Type: 2

Info-Req: 32 bits. This bitflag field lists the request items the probe sender is interested. The bit number ranges from the right most bit to the left most bit. Currently defined as the following:

<table>
<thead>
<tr>
<th>Bit Number</th>
<th>Information Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>MPLS label related attributes</td>
</tr>
<tr>
<td>1</td>
<td>Interface related attributes</td>
</tr>
<tr>
<td>2</td>
<td>IP/IPv6 address related attributes</td>
</tr>
<tr>
<td>3</td>
<td>Routing Instance related attributes</td>
</tr>
<tr>
<td>4</td>
<td>Nexthop(s) related attributes</td>
</tr>
<tr>
<td>5</td>
<td>Device role related attributes</td>
</tr>
</tbody>
</table>

4.2. Trace-Ping Extension Offset Field

This Trace-Ping "extension offset" field is defined as the lowest nibble within a 16-bit field, and it specifies the position at which the Trace-Ping extension data structure begins. The value represents 32-bit words ranges from 0x0 to 0xF, with value 0xF as reserved. Thus the position of the Trace-Ping data structure can start from 0 to 56 octets inside TCP, UDP or ICMP request and reply message data field. The Trace-Ping "extension offset" field value 0xF indicates there is no Trace-Ping data structure inside the message data field.

The Trace-Ping "extension offset" field is defined as the following:

```
  0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
|                       | Ext-Off |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
```

Ext-Off: 4 bits. The value (Ext-Off) represents the Trace-Ping data structure start position in 32-bit words. The Ext-Off value 0xF is reserved.

The rest of the 12 bits out of this 16-bit field is not changed by this proposal. For application usage detail in terms of different traceroute/ping probe types, see Section 4.2.4.

4.2.1. UDP Messages

In the UDP traceroute/ping probe case, this 16-bit field is the UDP source port field in UDP header [RFC0768]. The "Ext-Off" specifies the Trace-Ping extension structure start location inside UDP data
field. The below diagram shows the "Ext-Off" position in the UDP header "Source Port" field.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port    |Ext-Off|        Destination Port       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Length                        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       UDP data octets...                      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.2.2. TCP Messages

In the TCP traceroute/ping probe case, this 16-bit field is the TCP source port field in TCP header [RFC0793]. The "Ext-Off" specifies the Trace-Ping extension structure start location inside TCP data field. The below diagram shows the "Ext-Off" position in the TCP header of the "Source Port" field.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|        Source Port    |Ext-Off|        Destination Port       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Sequence Number                     |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Acknowledgment Number               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       Data |[U|A|P|R|S|F]          |
| Offset| Reserved |R|C|S|S|Y|I|           Window                   |
|       |           |G|K|H|T|N|N|                       |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Checksum                |         Urgent Pointer        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                        Options                 |    Padding                   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                       data                              |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

4.2.3. ICMP Messages

In the ICMP traceroute/ping request/reply case, this 16-bit field is the "Identifier" field of ICMP type 0 or 8 structure [RFC0792], The "Ext-Off" specifies the Trace-Ping extension structure start location inside ICMP data field of the type 0 (ICMP echo reply) and type 8 (ICMP echo request) message. The below diagram shows the "Ext-Off" position in the ICMP type 0 or 8 "Identifier" field.

```
0                   1                   2                   3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Code      |          Checksum             |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|       Identifier      |Ext-Off|        Sequence Number        |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Data ...           |
+-+-+-+-+-+-+-+-+
```

The ICMP echo reply is normally implemented by copying the data field of the ICMP echo request into the ICMP echo reply data field. When this Trace-Ping extension is used, it will overwrite part or whole portion of the original data field. The ICMP echo reply packet can be longer in length then the ICMP echo request in the case application has more data to include then the original echo request data field length. The ICMP echo reply application data using Trace-Ping extension is located by the "Ext-Off" field.

If the ICMP echo request and reply both support this Trace-Ping extension, then the upper 12-bits of the Id field of the ICMP echo request SHOULD be copied to the upper 12-bits of the Id field of the ICMP echo reply message. The "Ext-Off" field of the echo request and reply messages can be different if this is necessary.

4.2.4. Implementation Discussion

In the majority of today’s traceroute implementations, the application process identifier (process-ID) is used as the UDP source port for UDP type of traceroute probe; in TCP implementation, the process-ID is used as the TCP source port for traceroute probe; and the process-ID is also used as the ICMP Identifier of ICMP type 8 message. With this extension, an implementation can use the highest 12-bits of the source port field for UDP/TCP header and ID field for ICMP type 8 message to encode this process information since the lowest 4-bits are now reserved for the Trace-Ping "extension offset".
Ping implementation is similar to traceroute, it either uses the process-ID or an internally generated number inside the ICMP echo request ID field and in UDP/TCP source port field. An implementation now can use the highest 12-bits of the field and leave the lowest 4-bits for the Trace-Ping extension.

5. Implementation and Operation Considerations

There is no change in this extension for the normal traceroute/ping implementation and operation except for reserving the lowest 4 bits in the UDP/TCP source port field and ICMP Id field of type 0 and type 8 message. The implementations for the sender can use the same semantics with this 16-bit field; and it makes no difference to the receivers if they don’t support this extension.

5.1. Traceroute and Ping Probe Sender

The sender supports this extension MAY include the Trace-Ping structure in it’s traceroute/ping probe to specify the request types, authentication key, or other application objects. The sender SHOULD set the "extension offset" value to 0xF if there is no Trace-Ping structure present inside the probe. The sender MAY request one or multiple types of information defined in the Trace-Ping "Info-Req" Object.

5.2. Traceroute and Ping Probe Receiver

When the traceroute/ping probe receiver, the intermediate and destination node, processes the incoming probe, it MAY check the Trace-Ping structure to verify if the sender is from an authenticated host and to see what types of information it requested. This check is only needed when the receiver tries to authenticate the probe sender, or when the receiver is formatting the ICMP and ICMPv6s that support multi-part messages or this Trace-Ping extension messages inside the echo reply object, and it has certain internal information that can be included in the ICMP packets.

If the Trace-Ping "extension offset" value is not 0xF, the Trace-Ping structure may be present. The receiver MUST verify the integrity of the data structure by examining the "version" field, the Magic-Number value, and the length of the Trace-Ping structure. It MUST perform the checksum to verify the Trace-Ping data structure. If the authentication Object is present and the local policy requires it to perform the verification, the receiver MUST use it’s locally stored shared key to validate the checksum in the Object. Multiple Authentication Keys can be used which can be useful in the case the probes are from trusted peer networks.
If the Trace-Ping "Info-Req" Object is included, the receiver SHOULD fetch the related information when formatting the ICMP packets, but MUST NOT include information that has the corresponding bitflag cleared.

Even if the authentication fails, the receiver MAY still send the regular ICMP reply back to the sender, without the requested or internal information, as if this Trace-Ping extension is not supported.

6. Security Considerations

This extension enhances the security of traceroute and ping operation in a backwards-compatible fashion. The mechanism allows the receiver to verify the sender of the traceroute/ping packet such that certain sensitive application, interface and network related information can be supplied in the internal network or across trusted networks.

The use of Cryptographic authentication (i.e., an Auth Type value of 2) allows for a strong authentication mechanism since the keys cannot be discerned by intercepting the packets. The proposed Keyed authentication does not prevent replay attacks. However, in the case of replay attacks, since the packet source IP/IPv6 address of the traceroute/ping probe can not be changed, there is no easy way for the attacker to retrieve the ICMP messages.

A router needs to protect against purposefully-bogus Traceroute packets with extensions that fail the authentication, as a high rate of messages can require significant processing time. [RFC1812] specifies how rate-limiting is applied to the generation of ICMP messages, and this rate-limiting deters the threat when applied before checking the Authentication. Additionally, when using Cryptographic authentication, the HMAC includes the source IP address, which means the HMAC will not validate if the traceroute/ping packet is sent over a NAT.

7. IANA Considerations

The Trace-Ping Extension contains Trace-Ping Objects. IANA is requested to assign a new Class-Num for the Trace-Ping extension, and a sub-registry under Trace-Ping extension to include c-types. This document has defined c-type 1 and 2 for authentication and information-request objects. c-types 3-0xF6 are allocated through Expert Review [RFC5226]. C-types 0xF7 to 0xFF are reserved for private use.
IANA should also establish a registry for Trace-Ping Info-Request Bits under the information-request sub-registry. This document defines bits 0 - 5 in Section 4.1.2.2. Bits 6-29 are allocated through Expert Review. Bits 30 - 31 are reserved for private use.

8. Acknowledgements

Many thanks to Dan Wing, Tony Li, and Tissa Senevirathne for their insightful comments and valuable suggestions regarding this document.

9. References

9.1. Normative References


9.2. Informative References


[RFC4884] Bonica, R., Gan, D., Tappan, D., and C. Pignataro,


Authors' Addresses

Naiming Shen
Cisco Systems
225 West Tasman Drive
San Jose, CA  95134
USA
Email: naiming@cisco.com

Carlos Pignataro
Cisco Systems
7200 Kit Creek Road
Research Triangle Park, NC  27709
USA
Email: cpignata@cisco.com

Rajiv Asati
Cisco Systems
7025 Kit Creek Road
Research Triangle Park, NC  27709
USA
Email: rajiva@cisco.com
Enke Chen
Cisco Systems
170 West Tasman Drive
San Jose, CA  95134
USA

Email: enkechen@cisco.com

Alia K. Atlas
Juniper Networks
10 Technology Park Drive
Westford, MA  01886
USA

Email: akatlas@juniper.net