Shared Use of Experimental TCP Options

Abstract

   This document describes how the experimental TCP option codepoints
   can concurrently support multiple TCP extensions, even within the
   same connection, using a new IANA TCP experiment identifier. This
   approach is robust to experiments that are not registered and to
   those that do not use this sharing mechanism. It is recommended for
   all new TCP options that use these codepoints.

Status of This Memo

   This is an Internet Standards Track document.

   This document is a product of the Internet Engineering Task Force
   (IETF). It represents the consensus of the IETF community. It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG). Further information on
   Internet Standards is available in Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6994.

Table of Contents

1. Introduction

   TCP includes options to enable new protocol capabilities that can be
   activated only where needed and supported [RFC793]. The space for
   identifying such options is small -- 256 values, of which 30 are
   assigned at the time of this document's publication [IANA]. Two of
   these codepoints (253, 254) are allocated to support experiments
   [RFC4727]. These values are intended for testing purposes or for use
   when an assigned codepoint is either not warranted or available,
   e.g., based on the maturity status of the defined capability (i.e.,
   Experimental or Informational, rather than Standards Track).

   Here, the term "experimental TCP options" refers to options that use
   the TCP experimental option codepoints [RFC4727]. Such experiments
   can be described in an RFC of any status (e.g., Experimental,
   Informational, etc.) and are intended to be used in controlled
   environments and are allowed in public deployments (when not enabled
   as default [RFC3692]). Nothing prohibits the deployment of multiple
   experiments in the same environment -- controlled or public.
   Further, some protocols are specified in Experimental or
   Informational RFCs, which either include parameters or design choices
   not yet understood or which might not be widely deployed [RFC2026].
   Typically, these TCP options are not eligible to receive assigned
   codepoints [RFC2780], so they need a way to share their use of the
   experimental codepoints.

   There is currently no mechanism to support shared use of the TCP
   experimental option codepoints, either by different experiments on
   different connections or for more than two experimental options in
   the same connection. Experimental options 253 and 254 are already
   deployed in operational code to support an early version of TCP

authentication. Option 253 is also documented for the experimental
TCP Cookie Transaction option [RFC6013]. This shared use results in
collisions in which a single codepoint can appear multiple times in a
single TCP segment and for which each use is ambiguous.

Other codepoints have been used without assignment (known as
"squatting"), notably 31-32 (TCP cookie transactions, as originally
distributed and in its API doc) and 76-78 (tcpcrypt) [Bi11] [Si11].
Commercial products reportedly also use unassigned options 33, 69-70,
and 76-78. Even though these uses are unauthorized, they currently
impact legitimate assignees.

Both such misuses (squatting on both experimental and assigned
codepoints) are expected to continue, but there are several
approaches that can alleviate the impact on cooperating protocol
designers. One proposal relaxes the requirements for assignment of
TCP options, allowing them to be assigned more readily for protocols
that have not been standardized through the IETF process [RFC5226].
Another proposal assigns a larger pool to the TCP experiment option
codepoints and manages their sharing through IANA coordination
[Ed11].

The approach proposed in this document does not require additional
TCP option codepoints and is robust to those who choose either not to
support it or not to register their experiments. The solution adds a
field to the structure of the experimental TCP option. This field is
populated with an "experiment identifier" (ExID) defined as part of a
specific option experiment. The ExID helps reduce the probability of
a collision of independent experimental uses of the same option
codepoint, both for those who follow this document (using registered
ExIDs) and those who do not (squatters who either ignore this
extension or do not register their ExIDs).

The solution proposed in this document is recommended for all new
protocols that use TCP experimental option codepoints. The
techniques described here may also enable shared use of other
experimental codepoints, but that issue is out of scope for this
document.

2. 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 [RFC2119].

   In this document, these words will appear with that interpretation
   only when in ALL CAPS. Lowercase uses of these words are not to be
   interpreted as carrying RFC 2119 significance.

In this document, the characters ">>" preceding an indented line(s)
indicates a compliance requirement statement using the key words
listed above. This convention aids reviewers in quickly identifying
or finding the explicit compliance requirements of this RFC.

3. TCP Experimental Option Structure

TCP options have the current common structure [RFC793], in which the
first byte is the codepoint (Kind) and the second byte is the length
of the option in bytes (Length):

```
        0               1               2               3
        01234567 89012345 67890123 45678901
        +--------+--------+--------+--------+
        |  Kind  | Length |       ...       |
        +--------+--------+--------+--------+
        |   ...
        +--------
```
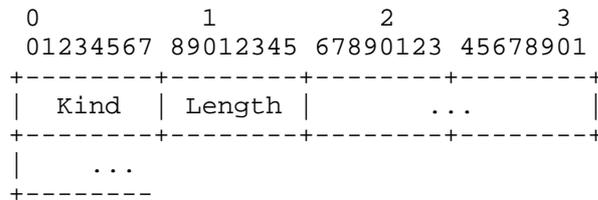
         Figure 1. TCP Option Structure [RFC793]

This document extends the option structure for experimental
codepoints (253, 254) with an experiment identifier (ExID), which is
either 2 or 4 bytes in length. The ExID is used to differentiate
experiments and is the first field after Kind and Length, as follows:

```
        0               1               2               3
        01234567 89012345 67890123 45678901
        +--------+--------+--------+--------+
        |  Kind  | Length |       ExID      |
        +--------+--------+--------+--------+
        | option contents...
        +--------+--------+--------+---
```
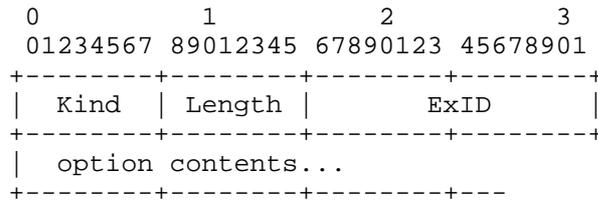
        Figure 2. TCP Experimental Option with a 16-bit ExID

```
        0               1               2               3
        01234567 89012345 67890123 45678901
        +--------+--------+--------+--------+
        |  Kind  | Length |       ExID      |
        +--------+--------+--------+--------+
        |   ExID (con't)  | option contents...
        +--------+--------+--------+---
```
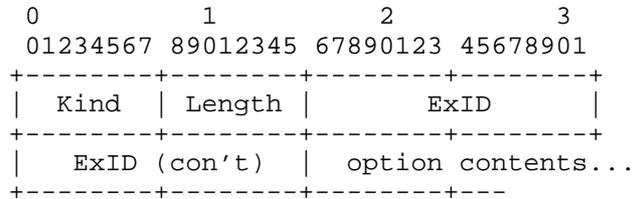
        Figure 3. TCP Experimental Option with a 32-bit ExID

   This mechanism is encouraged for all TCP options that are not yet
   eligible for assigned codepoints:

   >> Protocols requiring new TCP option codepoints that are not
      eligible for assigned values SHOULD use the existing TCP
      experimental option codepoints (253, 254) with ExIDs as described
      in this document.

   This mechanism is encouraged for all TCP options using the current
   experimental codepoints in controlled environments:

   >> All protocols using the TCP experimental option codepoints (253,
      254), even those deployed in controlled environments, SHOULD use
      ExIDs as described in this document.

   This mechanism is required for all TCP options using the current
   experimental codepoints that are publicly deployed, whether enabled
   by default or not:

   >> All protocols using the TCP experimental option codepoints (253,
      254) that are deployed outside controlled environments, such as in
      the public Internet, MUST use ExIDs as described in this document.

   Once a TCP option uses the mechanism in this document, registration
   of the ExID with IANA is required:

   >> All protocols using ExIDs as described in this document MUST
      register those ExIDs with IANA.

      Applicants register their desired ExID by contacting IANA [IANA].

3.1. Selecting an ExID

   ExIDs are selected at design time, when the protocol designer first
   implements or specifies the experimental option. ExIDs can be either
   16 bits or 32 bits. In both cases, the value is stored in the header
   in network-standard (big-endian) byte order. ExIDs combine
   properties of IANA registered codepoints with "magic numbers".

   >> All ExIDs MUST be either 16 bits or 32 bits long.

   Use of the ExID, whether 16 bit or 32 bit, helps reduce the
   probability of a false positive collision with those who either do
   not register their experiment or who do not implement this mechanism.

In order to conserve TCP option space, either for use within a
specific option or to be available for other options:

>> Options implementing the mechanism of this document SHOULD use
   16-bit ExIDs, except where explicitly motivating the need for
   32-bit ExIDs, e.g., to avoid false positives or maintain alignment
   with an expected future assigned codepoint.

ExIDs are registered with IANA using "first come, first served"
(FCFS) priority based on the first two bytes. Those two bytes are
thus sufficient to interpret which experimental option is contained
in the option field.

>> All ExIDs MUST be unique based on their first 16 bits.

The second two bytes serve as a "magic number". A magic number is a
self-selected codepoint whose primary value is its unlikely collision
with values selected by others. Magic numbers are used in other
protocols, e.g., bootstrap protocol (BOOTP) [RFC951] and DHCP
[RFC2131].

Using the additional magic number bytes helps the option contents
have the same byte alignment in the TCP header as they would have if
(or when) a conventional (non-experiment) TCP option codepoint is
assigned. Use of the same alignment reduces the potential for
implementation errors, especially in using the same word-alignment
padding, if the same software is later modified to use a conventional
codepoint. Use of the longer, 32-bit ExID further decreases the
probability of such a false positive compared to those using shorter,
16-bit ExIDs.

Use of the ExID does consume TCP option space but enables concurrent
use of the experimental codepoints and provides protection against
false positives, leaving less space for other options (including
other experiments). Use of the longer, 32-bit ExID consumes more
space, but provides more protection against false positives.

3.2. Impact on TCP Option Processing

The ExID number is considered part of the TCP option, not the TCP
option header. The presence of the ExID increases the effective
option Length field by the size of the ExID. The presence of this
ExID is thus transparent to implementations that do not support TCP
options.

During TCP processing, ExIDs in experimental options are matched
against the ExIDs for each implemented protocol. The remainder of
the option is specified by the particular experimental protocol.

   >> Experimental options with ExIDs that do not match implemented
      protocols MUST be ignored.

   The ExID mechanism must be coordinated during connection
   establishment, just as with any TCP option.

   >> TCP ExID, if used in any TCP segment of a connection, MUST be
      present in TCP SYN segments of that connection.

   >> TCP experimental option ExIDs, if used in any TCP segment of a
      connection, SHOULD be used in all TCP segments of that connection
      in which any experimental option is present.

   Use of an ExID uses additional space in the TCP header and requires
   additional protocol processing by experimental protocols. Because
   these are experiments, neither consideration is a substantial
   impediment; a finalized protocol can avoid both issues with the
   assignment of a dedicated option codepoint later.

4. Reducing the Impact of False Positives

   False positives occur where the registered ExID of an experiment
   matches the value of an option that does not use ExIDs. Such
   collisions can cause an option to be interpreted by the incorrect
   processing routine. Use of checksums or signatures may help an
   experiment use the shorter ExID while reducing the corresponding
   increased potential for false positives.

   >> Experiments that are not robust to ExID false positives SHOULD
      implement other detection measures, such as checksums or minimal
      digital signatures over the experimental options they support.

5. Migration to Assigned Options

   Some experiments may transition away from being experimental and
   become eligible for an assigned TCP option codepoint. This document
   does not recommend a specific migration plan to transition from use
   of the experimental TCP options/ExIDs to use of an assigned
   codepoint.

   However, once an assigned codepoint is allocated, use of an ExID
   represents unnecessary overhead. As a result:

   >> Once a TCP option codepoint is assigned to a protocol, that
      protocol SHOULD NOT continue to use an ExID as part of that
      assigned codepoint.

This document does not recommend whether or how an implementation of
an assigned codepoint can be backward compatible with use of the
experimental codepoint/ExID.

However, some implementers may be tempted to include both the
experimental and assigned codepoint in the same segment, e.g., in a
SYN to support backward compatibility during connection
establishment. This is a poor use of limited resources; so, to
ensure conservation of the TCP option space:

>> A TCP segment MUST NOT contain both an assigned TCP option
   codepoint and a TCP experimental option codepoint for the same
   protocol.

Instead, a TCP that intends backward compatibility might send
multiple SYNs with alternates of the same option and discard all but
the most desired successful connection. Although this approach may
resolve more slowly or require additional effort at the endpoints, it
is preferable to excessively consuming TCP option space.

6. Rationale

The ExIDs described in this document combine properties of IANA
FCFS-registered values with magic numbers. Although IANA FCFS
registries are common, so too are those who either fail to register
or who 'squat' by deliberately using codepoints that are assigned to
others. The approach in this document is intended to recognize this
reality and be more robust to its consequences than would be a
conventional IANA FCFS registry.

Existing ID spaces were considered as ExIDs in the development of
this mechanism, including IEEE Organizationally Unique Identifier
(OUI) and IANA Private Enterprise Numbers (PENs) [IEEE802] [OUI]
[RFC1155].

OUIs are 24-bit identifiers that are combined with 24 to 40 bits of
privately assigned space to create identifiers that are commonly
assigned to a unique piece of hardware. OUIs are already longer than
the smaller ExID value, and obtaining an OUI is costly (currently
$1,885.00 USD). An OUI could be obtained for each experiment, but
this could be considered expensive. An OUI already assigned to an
organization could be shared if extended (to support multiple
experiments within an organization), but this would either require
coordination within an organization or an IANA registry; the former
is prohibitive, and the latter is more complicated than having IANA
manage the entire space.

PENs were originally used in the Simple Network Management Protocol
(SNMP) [RFC1157]. PENs are identifiers that can be obtained without
cost from IANA [PEN]. Despite the current registry, the size of the
PEN assignment space is currently undefined and has only recently
been proposed (as 32 bits) [IANA-PEN]. PENs are currently assigned
to organizations, and there is no current process for assigning them
to individuals. Finally, if the PENs are 32 bits as expected, they
would be larger than needed in many cases.

7. Security Considerations

   The mechanism described in this document is not intended to enhance,
   nor does it weaken the existing state of security for TCP option
   processing.

8. IANA Considerations

   IANA has created a "TCP Experimental Option Experiment Identifiers
   (TCP ExIDs)" registry. The registry records both 16-bit and 32-bit
   ExIDs, as well as a reference (description, document pointer, or
   assignee name and e-mail contact) for each entry. ExIDs are
   registered for use with both of the TCP experimental option
   codepoints, i.e., with TCP options with values of 253 and 254.

   Entries are assigned on a First Come, First Served (FCFS) basis
   [RFC5226]. The registry operates FCFS on the first two bytes of the
   ExID (in network-standard order) but records the entire ExID (in
   network-standard order). Some examples are:

   o 0x12340000 collides with a previous registration of 0x1234abcd

   o 0x5678 collides with a previous registration of 0x56780123

   o 0xabcd1234 collides with a previous registration of 0xabcd

   IANA will advise applicants of duplicate entries to select an
   alternate value, as per typical FCFS processing.

   IANA will record known duplicate uses to assist the community in both
   debugging assigned uses as well as correcting unauthorized duplicate
   uses.

   IANA should impose no requirements on making a registration other
   than indicating the desired codepoint and providing a point of
   contact. A short description or acronym for the use is desired but
   should not be required.

9. References

9.1. Normative References

   [RFC793]   Postel, J., "Transmission Control Protocol", STD 7, RFC
               793, September 1981.

   [RFC2119]   Bradner, S., "Key words for use in RFCs to Indicate
               Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC4727]   Fenner, B., "Experimental Values In IPv4, IPv6, ICMPv4,
               ICMPv6, UDP, and TCP Headers", RFC 4727, November 2006.

   [RFC5226]   Narten, T. and H. Alvestrand, "Guidelines for Writing an
               IANA Considerations Section in RFCs", BCP 26, RFC 5226,
               May 2008.

9.2. Informative References

   [Bi11]   Bittau, A., Boneh, D., Hamburg, M., Handley, M., Mazieres,
               D., and Q. Slack, "Cryptographic protection of TCP
               Streams", Work in Progress, September 2012.

   [Ed11]
               Eddy, W., "Additional TCP Experimental-Use Options", Work
               in Progress, August 2011.

   [IANA]   IANA, <http://www.iana.org/>.

   [IANA-PEN] Liang, P. and A. Melnikov, "Private Enterprise Number
               (PEN) Practices and Internet Assigned Numbers: Authority
               (IANA) Considerations for Registration Procedures", Work
               in Progress, June 2012.

   [IEEE802]   IEEE, "IEEE Standard for Local and Metropolitan Area
               Networks: Overview and Architecture", IEEE 802-2001, 8
               March 2002.

   [OUI]   IEEE, "Organizationally Unique Identifier (OUI) or
               'Company_ID'",
               <http://standards.ieee.org/develop/regauth/oui/>.

   [PEN]   IANA, "Private Enterprise Numbers",
               <http://www.iana.org/assignments/enterprise-numbers>.

   [RFC951]   Croft, W. and J. Gilmore, "Bootstrap Protocol", RFC 951,
               September 1985.

   [RFC1155] Rose, M. and K. McCloghrie, "Structure and Identification
              of Management Information for TCP/IP-Based Internets", STD
              16, RFC 1155, May 1990.

   [RFC1157] Case, J., Fedor, M., Schoffstall, M., and J. Davin,
              "Simple Network Management Protocol (SNMP)", RFC 1157, May
              1990.

   [RFC2026] Bradner, S., "The Internet Standards Process -- Revision
              3", BCP 9, RFC 2026, October 1996.

   [RFC2131] Droms, R., "Dynamic Host Configuration Protocol", RFC
              2131, March 1997.

   [RFC2780] Bradner, S. and V. Paxson, "IANA Allocation Guidelines For
              Values In the Internet Protocol and Related Headers", BCP
              37, RFC 2780, March 2000.

   [RFC3692] Narten, T., "Assigning Experimental and Testing Numbers
              Considered Useful", BCP 82, RFC 3692, January 2004.

   [RFC6013] Simpson, W., "TCP Cookie Transactions (TCPCT)", RFC 6013,
              January 2011.

   [Si11]   Simpson, W., "TCP Cookie Transactions (TCPCT) Sockets
              Application Program Interface (API)", Work in Progress,
              April 2011.

10. Acknowledgments

   This document was motivated by discussions on the IETF TCPM mailing
   list and by Wes Eddy's proposal [Ed11]. Yoshifumi Nishida, Pasi
   Sarolathi, and Michael Scharf provided detailed feedback.

   This document was originally prepared using 2-Word-v2.0.template.dot.

Author's Address

   Joe Touch
   USC/ISI
   4676 Admiralty Way
   Marina del Rey, CA 90292-6695 U.S.A.

   Phone: +1 (310) 448-9151
   EMail: touch@isi.edu