TCPM Working Group
Internet Draft
Intended status: Proposed Standard
Expires: November 2012

J. Touch
USC/ISI

May 30, 2012

Shared Use of Experimental TCP Options
draft-ietf-tcpm-experimental-options-01.txt

Status of this Memo

Copyright Notice

Abstract

This document describes how TCP option codepoints can support concurrent experiments using a magic number field. This mechanism avoids the need for a coordinated registry, and is backward-compatible with currently known uses.

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 31 are assigned at the time this document was published [IANA]. Two of these codepoints are allocated to support experiments (253, 254) [RFC4727]. These numbers are intended for testing purposes, and implementations need to assume they can be used for other purposes, but this is often not the case.

There is no mechanism to support shared use of the experimental option codepoints. Experimental options 253 and 254 are 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 each use is ambiguous.

Other codepoints have been used without assignment, 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 and 76-78 as well. Even though these uses are inappropriate, they can impact legitimate assignees.

There are a variety of proposed approaches to address this issue. The first is to relax 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]. A second would be to assign a larger pool to options, and to manage their sharing through IANA coordination [Ed11].

This document proposes a solution that does not require additional codepoints and also avoids IANA involvement. The solution involves adding a field to the structure of the experimental TCP option. This field is typically populated with a fixed "magic number" defined as part of a specific option experiment. The magic number helps reduce the probability of a collision of independent experimental uses of the same option codepoint. This feature increases the number of bytes used by experimental options, but the size can be reduced when the experiment is converted to a standard protocol with a conventional codepoint assignment.

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. Lower case 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, where the first byte is the codepoint (Kind) and the second is the length of the option in bytes (Length):

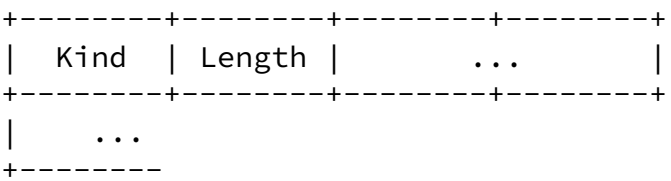

Figure 1 TCP Option Structure [RFC793]

This document extends the option structure for experimental codepoints (253, 254) with a magic number. The magic number is used to differentiate different experiments, and is the first field after the Kind and Length, as follows:

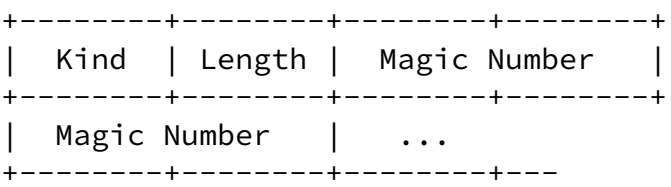

Figure 2 TCP Experimental Option with a Magic Number

>> Protocols using the TCP experimental option codepoints (253, 254) SHOULD use magic numbers as described in this document.

Magic numbers are used in other protocols, e.g., BOOTP [RFC951] and DHCP [RFC2131]. Here they help ensure that concurrent experiments that share the same TCP option codepoint do not interfere.

The magic number is selected by the protocol designer when an experimental option is defined. The magic number is selected any of a variety of ways, e.g., using the Unix time() command or bits selected by an arbitrary function (such as a hash).

>> The magic number size and value SHOULD be selected to reduce the probability of collision.

This document does not proscribe a minimum magic number size. However, a reasonable suggested size is 32 bits, in network standard byte order:

>> The magic number SHOULD be 32 bits, but MAY be either longer or shorter.

The magic number is considered part of the TCP option, not the TCP option header. The presence of the magic number increases the effective option Length field by the size of the magic number. The

presence of this magic number is thus transparent to implementations that do not support TCP options where it is used.

During TCP processing, experimental options are matched against both the experimental codepoints and the magic number value for each implemented protocol.

>> Experimental options that have magic numbers that do not match implemented protocols MUST be ignored.

The remainder of the option is specified by the particular experimental protocol. This includes the possibility that the magic number could appear in only a subset of instances of the option. Because TCP option capabilities are negotiated during connection establishment, the magic number might be omitted afterwards (e.g., in non-SYN segments).

>> Experimental option magic numbers, if used, MUST be present in TCP SYN segments.

The specification of an experimental option needs to describe whether the magic number appears in non-SYN segments. If the magic number does not appear in all segments, the experimental option may need to be rejected during connection negotiation because options for different experiments in non-SYN segments may not be distinguishable. As a result, this document recommends that:

>> Experimental option magic numbers, if used, SHOULD be used in all

TCP segments where the option is present.

Use of a magic number 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.

3.1. Reducing the Impact of False Positives

False positives are always possible, where a magic number matches the value of a field in the legacy use of these options or a protocol that does not implement the mechanism described in this document.

>> Protocols that are not robust to magic number false positives SHOULD implement other measures to ensure they process options for their protocol only, such as checksums or digital signatures among

cooperating parties of their protocol. Such measures SHOULD supplement, rather than substitute for, the use of magic numbers.

Use of checksums or signatures may help an experiment use a shorter magic number while reducing the corresponding increased potential for false positives. However this document recommends magic numbers are used together with such checksums/signatures, not as a substitute thereof. Magic numbers are static and thus more easily identify the experiment using the experimental option; they can also be more efficiently interpreted at the TCP receiver.

3.2. Migration to Assigned Options

This document does not address a specific migration plan to avoid the use of magic numbers once an experimental TCP option is considered for operational deployment, e.g., if it transitions to proposed standard. The expectation is that such options would be assigned their own TCP codepoints and their specifications updated to avoid the need to support the experimental codepoint.

4. Security Considerations

The mechanism described in this document is not intended to provide

security for TCP option processing.

5. IANA Considerations

This document has no IANA considerations. This section should be removed prior to publication.

6. References

6.1. Normative References

[RFC793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, Sep. 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, Nov. 2006.

6.2. Informative References

[Bi11] Bittau, A., D. Boneh, M. Hamburg, M. Handley, D. Mazieres, Q. Slack, "Cryptographic protection of TCP Streams (tcpcrypt)", work in progress, draft-bittau-tcp-crypt-02, Feb. 20, 2012.

[Ed11] Eddy, W., "Additional TCP Experimental-Use Options", work in progress, draft-eddy-tcpm-addl-exp-options-00, Aug. 16, 2011.

[IANA] IANA web pages, http://www.iana.org/

[RFC951] Croft, B., J. Gilmore, "BOOTSTRAP PROTOCOL (BOOTP)", RFC 951, Sept. 1985.

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

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

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

[Si11] Simpson, W., "TCP Cookie Transactions (TCPCT) Sockets Application Program Interface (API)", work in progress, draft-simpson-tcpct-api-04, Apr. 7, 2011.

7. 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 Sharf provided detailed feedback.

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

Authors' Addresses

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