

## TCP Option to Denote Packet Mood

### Abstract

This document proposes a new TCP option to denote packet mood.

### Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This is a contribution to the RFC Series, independently of any other RFC stream. The RFC Editor has chosen to publish this document at its discretion and makes no statement about its value for implementation or deployment. Documents approved for publication by the RFC Editor are not a candidate for any level of Internet Standard; see [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/rfc5841>.

### Copyright Notice

Copyright (c) 2010 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.

## 1. Introduction

In an attempt to anthropomorphize the bit streams on countless physical layer networks throughout the world, we propose a TCP option to express packet mood [DSM-IV].

Packets cannot feel. They are created for the purpose of moving data from one system to another. However, it is clear that in specific situations some measure of emotion can be inferred or added. For instance, a packet that is retransmitted to resend data for a packet for which no ACK was received could be described as an 'angry' packet, or a 'frustrated' packet (if it is not the first retransmission for instance). So how can these kinds of feelings be conveyed in the packets themselves. This can be addressed by adding TCP Options [RFC793] to the TCP header, using ASCII characters that encode commonly used "emoticons" to convey packet mood.

### 1.1. Terminology

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [RFC2119].

## 2. Syntax

A TCP Option has a 1-byte kind field, followed by a 1-byte length field [RFC793]. It is proposed that option 25 (released 2000-12-18) be used to define packet mood. This option would have a length value of 4 or 5 bytes. All the simple emotions described as expressible via this mechanism can be displayed with two or three 7-bit, ASCII-encoded characters. Multiple mood options may appear in a TCP header, so as to express more complex moods than those defined here (for instance if a packet were happy and surprised).

#### TCP Header Format

Kind	Length	Meaning
----	-----	-----
25	Variable	Packet Mood

In more detail:

```

+-----+-----+-----+-----+
|00011001|00000100|00111010|00101001|
+-----+-----+-----+-----+
Kind=25 Length=4 ASCII : ASCII )

+-----+-----+-----+-----+-----+
|00011001|00000101|00111110|00111010|01000000|
+-----+-----+-----+-----+-----+
Kind=25 Length=5 ASCII > ACSII : ASCII @

```

### 3. Simple Emotional Representation

It is proposed that common emoticons be used to denote packet mood. Packets do not "feel" per se. The emotions they could be tagged with are a reflection of the user mood expressed through packets.

So the humanity expressed in a packet would be entirely sourced from humans.

To this end, it is proposed that simple emotions be used convey mood as follows.

ASCII	Mood
=====	=====
:)	Happy
:(	Sad
:D	Amused
%()	Confused
:o	Bored
:O	Surprised
:P	Silly
:@	Frustrated
>:@	Angry
:	Apathetic
;)	Sneaky
>:)	Evil

## Proposed ASCII character encoding

Binary	Dec	Hex	Character
=====	===	===	=====
010 0101	37	25	%
010 1000	40	28	(
010 1001	41	29	)
011 1010	58	3A	:
011 1011	59	3B	;
011 1110	62	3E	>
100 0000	64	40	@
100 0100	68	44	D
100 1111	79	4F	O
101 0000	80	50	P
110 1111	111	6F	o
111 1100	124	7C	

For the purposes of this RFC, 7-bit ASCII encoding is sufficient for representing emoticons. The ASCII characters will be sent in 8-bit bytes with the leading bit always set to 0.

#### 4. Use Cases

There are two ways to denote packet mood. One is to infer the mood based on an event in the TCP session. The other is to derive mood from a higher-order action at a higher layer (subject matter of payload for instance).

For packets where the 'mood' is inferred from activity within the TCP session, the 'mood' MUST be set by the host that is watching for the trigger event. If a client sends a frame and receives no ACK, then the retransmitted frame MAY contain the TCP OPTION header with a mood set.

Any packet that exhibits behavior that allows for mood to be inferred SHOULD add the TCP OPTION to the packets with the implied mood.

Applications can take advantage of the defined moods by expressing them in the packets. This can be done in the SYN packet sent from the client. All packets in the session can be then tagged with the mood set in the SYN packet, but this would have a per-packet performance cost (see [Section 5](#), "Performance Considerations").

Each application MUST define the preconditions for marking packets as happy, sad, bored, confused, angry, apathetic, and so on. This is a framework for defining how such moods can be expressed, but it is up to the developers to determine when to apply these encoded labels.

#### 4.1. Happy Packets

Healthy packets are happy packets you could say. If the ACK packets return within <10 ms end-to-end from a sender's stack to a receiver's stack and back again, this would reflect high-speed bidirectional capability, and if no retransmits are required and all ACKs are received, all subsequent packets in that session SHOULD be marked as 'happy'.

No loss, low-latency packets also makes for happy users. So the packet would be reflecting the end-user experience.

#### 4.2. Sad Packets

If retransmission rates achieve greater than 20% of all packets sent in a session, it is fair to say the session can be in mourning for all of the good packets lost in the senseless wasteland of the wild Internet.

This should not be confused with retransmitted packets marked as 'angry' since this tag would apply to all frames in the session numbed by the staggering loss of packet life.

#### 4.3. Amused Packets

Any packet that is carrying a text joke SHOULD be marked as 'amused'.

Example:

```
1: Knock Knock
2: Who's there?
1: Impatient chicken
2: Impatient chi...
1: BAWK!!!!
```

If such a joke is in the packet payload then, honestly, how can you not be amused by one of the only knock-knock jokes that survives the 3rd grade?

#### 4.4. Confused Packets

When is a packet confused? There are network elements that perform per-packet load balancing, and if there are asymmetries in the latencies between end-to-end paths, out-of-order packet delivery can occur.

When a receiver host gets out-of-order packets, it SHOULD mark TCP ACK packets sent back to the sender as confused.

The same can be said for packets that are sent to incorrect VLAN segments or are misdirected. The receivers might be aware that the packet is confused, but there is no way to know at ingress if that will be the fate of the frame.

That being said, application developers SHOULD mark packets as confused if the payload contains complex philosophical questions that make one ponder the meaning of life and one's place in the universe.

#### 4.5. Bored Packets

Packets carrying accounting data with debits, credits, and so on MUST be marked as 'bored'.

It could be said that many people consider RFCs boring. Packets containing RFC text MAY be marked as 'bored'.

Packets with phone book listings MUST be marked 'bored'.

Packets containing legal disclaimers and anything in Latin SHOULD be marked 'bored'.

#### 4.6. Surprised Packets

Who doesn't love when the out-of-order packets in your session surprise you while waiting in a congested queue for 20 ms?

Packets do not have birthdays, so packets can be marked as surprised when they encounter unexpected error conditions.

So when ICMP destination unreachable messages are received (perhaps due to a routing loop or congestion discards), all subsequent packets in that session SHOULD be marked as surprised.

#### 4.7. Silly Packets

Not all packets are sent as part of a session. Random keepalives during a TCP session MAY be set up as a repartee between systems connected as client and server. Such random and even playful interchanges SHOULD be marked as silly.

#### 4.8. Frustrated Packets

Packets that are retransmitted more than once SHOULD be marked as frustrated.

#### 4.9. Angry Packets

Packets that are retransmitted SHOULD be marked as angry.

#### 4.10. Apathetic Packets

When sending a RST packet to a connected system, the packet should be marked as apathetic so that the receiver knows that your system does not care what happens after that.

#### 4.11. Sneaky Packets

When a packet is used in a particularly clever way, it SHOULD be marked as sneaky. What is "clever" is rather subjective, so it would be prudent to get a few opinions about a particular use to make sure that it is clever.

#### 4.12. Evil Packets

It is hard for a TCP packet to discern higher moral quandaries like the meaning of life or what exactly defines 'evil' and from whose perspective such a characterization is being made. However, developers of TCP-based applications MAY choose to see some activities as evil when viewed through their particular lens of the world. At that point, they SHOULD mark packets as evil.

Some organizations are prohibited from using this mood by mission statement. This would also prohibit using the security flag in the IP header described in [RFC3514] for the same reasons.

### 5. Performance Considerations

Adding extensions to the TCP header has a cost. Using TCP extensions with the ASCII-encoded mood of the packet would detract from the available MSS usable for data payload. If the TCP header is more than 20 bytes, then the extra bytes would be unavailable for use in the payload of the frame.

This added per-packet overhead should be considered when using packet mood extensions.

### 6. Security Considerations

The TCP checksum, as a 16-bit value, could be mistaken if ASCII characters with the same number of zeros and ones were substituted out. A happy ":" could be replaced with a frown by a malicious attacker, by using a winking eye ";(". This could misrepresent the intended mood of the sender to the receiver.

## 7. Related Work

This document does not seek to build a sentient network stack. However, this framework could be used to express the emotions of a sentient stack. If that were to happen, a new technical job class of network psychologists could be created. Who doesn't like new jobs? :)

## 8. IANA Considerations

If this work is standardized, IANA is requested to officially assign value 25 as described in [Section 3](#). Additional moods and emoticon representations would require IESG approval or standards action [[RFC5226](#)].

## 9. Informative References

- [DSM-IV] "Diagnostic and Statistical Manual of Mental Disorders (DSM)", <http://www.psychiatryonline.com/resourceTOC.aspx?resourceID=1>.
- [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.
- [RFC5226] Narten, T. and H. Alvestrand, "Guidelines for Writing an IANA Considerations Section in RFCs", [BCP 26](#), [RFC 5226](#), May 2008.
- [RFC3514] Bellovin, S., "The Security Flag in the IPv4 Header", [RFC 3514](#), April 1 2003.



## Authors' Addresses

Richard Hay  
Google  
1600 Amphitheatre Pkwy  
Mountain View, CA 94043  
EMail: rhay@google.com

Warren Turkal  
Google  
1600 Amphitheatre Pkwy  
Mountain View, CA 94043  
EMail: turkal@google.com