

<[draft-narten-ppp-over-sonet-to-historic-00.txt](#)>

August 7, 1998

PPP Over SONET Applicability Statement for Historic Status<[draft-narten-ppp-over-sonet-to-historic-00.txt](#)>

Status of this Memo

This document is an Internet-Draft. 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."

To view the entire list of current Internet-Drafts, please check the ``1id-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Northern Europe), ftp.nis.garr.it (Southern Europe), munnari.oz.au (Pacific Rim), ftp.ietf.org (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this memo is unlimited.

This Internet Draft expires February, 1999.

1. Introduction

PPP over Sonet [[RFC 1619](#)] uses "HDLC-like framing" as defined in [RFC 1662]. These documents were developed to address a void in existing SONET standards at that time.

T1.X1 is the ANSI accredited T1 Technical Subcommittee responsible for developing SONET standards, including SONET rates, formats and payload mappings. Its output is also brought into the ITU, for example as in ITU-T Recommendation G.707 [[G.707](#)].

At the December, 1997 IETF in Washington DC, a BOF was held to discuss a potential weakness in previous mappings. Specifically, long packets (where "long" can include those carried by PPP) containing a special bit pattern can cause a SONET receiver to declare a "loss of

signal". The minutes of the BOF are included in [Appendix A](#).

T1.X1 has recently defined a mapping for HDLC over SONET [T1.105.02, T1.105] that addresses the potential weakness described above. The work done by T1.X1 on SONET supersedes past efforts that have taken place within the IETF. The recommendation of this applicability statement is that all PPP over SONET implementations use the combination of [RFC 1662](#) plus "Packet Over Sonet/SDH" [POS]. The first document describes how to encapsulate PPP in HDLC; the second document describes how to encapsulate HDLC in SONET. We also recommend that [RFC 1619](#) be reclassified as Historic.

2. References

- [RFC 1619] Simpson, W., "PPP over SONET/SDH", [RFC 1619](#), May 1994.
- [RFC 1662] Simpson, W., "PPP in HDLC-like Framing", [RFC 1662](#), July 1994.
- [T1.105.02] ANSI T1.105.02 "Synchronous Optical Network (SONET) - Payload Mappings"
- [T1.105] ANSI T1.105 "Synchronous Optical Network (SONET) - Basic Description including Multiplex Structure, Rates and Formats".
- [G.707] CCITT Recommendation G.707, "Synchronous Digital Hierarchy Bit Rates", 1998 (??? - XXX).
- [POS] Narten, T., "Packet over Sonet/SDH", [draft-narten-packet-over-sonet-00.txt](#).

3. Author's Address

Thomas Narten
IBM Corporation
3039 Cornwallis Ave.
PO Box 12195 - BRQA/502
Research Triangle Park, NC 27709-2195

Phone: 919-254-7798
EMail: narten@raleigh.ibm.com

APPENDIX A: Minutes of the PPP Over Sonet/SDH Bof

Minutes of the PPP Over Sonet/SDH Bof 10 December 1997

Chair: Mike O'Dell

Scribe: Frank Kastenholtz

Agenda:

- 0. Introduction and Agenda-Bashing
- 1. Statement of the Problem
- 2. Stop gaps for the near-term
- 3. Long-term Solution

Mike O'Dell opened the meeting, presenting the agenda. There was no agenda-bashing.

Peter Lothberg gave a brief talk on what the problem was:

If a SONET receiver detects a "long" string of 0s ("no light") then the receiver will declare a "loss of signal". Of course, someone may, legitimately, wish to send a large number of 0s; this should not cause an LOS.

In order to prevent this, SONET uses a 'scrambler' which attempts to ensure transmission of a sufficient number of 1s so that the receiver will not declare a false LOS. This scrambler works well for rather short data blocks. However, for long blocks (such as PPP Packets), it is possible to devise packets that, after scrambling, show up on the fiber as a long string of 0s (i.e. no light). Of course, it is also possible that such a packet could be generated in the normal course of events (i.e. non-maliciously and non-consciously)

Furthermore, an active attack could be launched against the SONET infrastructure from anyplace in the Internet.

This is not a real operational issue right now. The current error rates on SONET links (including otherwise unexplained LOS) is well within the nominal error rates expected for SONET. However, since this issue was brought up, there have been some reports of attacks being made.

Peter then presented a stop-gap to deal with this issue in the short term.

A new scrambler (the "ATM Scrambler") would be inserted between the HDLC Byte-Stuffing and the normal sonet payload scrambler.

This has the property that it can be done in software in the near term and migrated into hardware later on. The ATM Scrambler is a $1+X^{43}$ scrambler. This scrambler is well known and easy to implement, is believed to have a sufficient number of states to protect the SONET networks. On the down-side, it does introduce error multiplication (because of the feed-back into the shift-register) and may interact with the CRC-16 of PPP Packets in ways to miss some errors.

The error multiplication was deemed not to be a significant issue since once a packet was "errored", it really wasn't all that big a deal if it had n bits of error or $2n$ bits or...

The interaction with CRC-16 can be alleviated in several ways:

- Packet header consistency checks (eg, the packet length and the IP header's Total Length field must be consistent)
- Transport error detection
- Use of CRC-32

Karl Fox suggested that the PPP/SONET specification be modified to require that CRC32 be used all the time (i.e. not be negotiated, just "always used").

Mike O'Dell then offered as a long-term solution that ANSI T1.X1 do an addendum to T1.105.02 (?) to specify "HDLC Over Sonet". T1.X1 would formally adopt the ATM scrambler as a part of this specification.

General Discussion then ensued.

Bill Simpson recommended that the solution be simple byte stuffing in order to prevent the known bad bit patterns from appearing.

There were many comments back and forth on the relative merits of byte stuffing vs $1+X^{43}$ scrambling (which was easier, which was more robust, implementing in hardware vs software, and so forth).

A "sense-of-the-BOF" was taken via the time-honored IETF Hum:

- The IETF should pursue the $1+X^{43}$ scrambler solution offered by Peter Lothberg for the near-term.
- The byte-stuffing technique would not be pursued.
- The long-term solution would be an ANSI T1.X1 "HDLC-over-SONET Mapping" which would use the $1+X^{43}$ scrambler.

Discussion then turned to the SONET Payload C2 byte. Should a new code be allocated for the "new" (i.e. with $1+X^{43}$ scrambling) payload format?

- The current standard ([RFC1619](#)) specifies an experimental code

- point of 207 (0xcf) for the C2 byte.
- A new code-point would allow receivers to differentiate new- from old-format payloads.
 - It was not clear whether all implementations generated or checked the C2-byte code-point, so putting in a new code point might not be a real win.

The recommendation of the BOF was that the IETF would not pursue a new code-point and would use the old one for "new-format" packets. However, if and when ANSI T1.X1 recommends a code-point for their mapping, that number should be adopted "with all deliberate haste."

ANSI T1X1 expects to make a decision on this in January 1998. Their results will be announced to the PPP Mailing List.

Peter Lothberg then made a brief presentation for "Bytes Over Sonet", a proposed new standard, with the following suggested features:

- No byte stuffing, length-fields would be used.
- A Combined scrambler/CRC
- Combined multiple STM-1s
- Some kind of TDM

A mailing list will be set up: bos@stupi.se. Subscription requests should be sent to bos-request@stupi.se

