NDT-Framing

Status of this Memo

   This document is an Internet-Draft and is NOT offered in accordance
   with Section 10 of RFC2026. The author does not provide the IETF with
   any rights other than to publish it as 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 made obsolete 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."
   The list of current Internet-Drafts can be accessed at http://
   www.ietf.org/ietf/1id-abstracts.txt
   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html.

Abstract

   During the last year several attempts have been made to provide a
   generalized framing mechanism for protocols using TCP. The main pur-
   pose of those mechanisms is to enable hardware network adapter build-
   ers to discover fast the next Upper Layer Protocol Data Unit boundary
   and thus minimize the amount of memory required on the network adapt-
   ers in order to handle packet drop or reordering in the network.

   Within IETF some work is being done for both generic framing mecha-
   nisms or framing mechanisms embedded in other protocols.
   For some generic approaches see [ULPF] - a draft that is being worked
   at within the Transport Working Group of IETF (TSWG).
   For some protocol specific mechanisms see [iSCSI] a draft being worked
   at within the IP Storage Working Group of IETF (IPS).
   For both - to get an understanding of the issues involved the reader
   is advised to browse in the archives of the mailing lists of the IPS
   and TSWG looking for framing related debates and summary of meetings.
   Some discussions are also held within the Remote DMA community that is
   not a formal working group of the IETF but is actively pursuing a
   generic Direct Data Placement mechanism (a steering mechanism) for
   several transport protocols assuming that a framing mechanism is in
   place.

Acknowledgements

   To my wife, Turi, that once observed that even when quoted random sen-
   tences you can recognize an author.

1. Introduction

   Technically the framing mechanisms pursued fall in one of the follow-
   ing categories:

      - framing indicators placed at predetermined positions in the
        data stream (pointers to the next ULP Data Unit (ULPDU) start
      - framing patterns that precede (or succeed) every frame

   In the framing pattern category two different approaches have been
   proposed:

      - using a framing pattern and transforming the data stream so
        that the pattern will not appear anywhere except as framing on
        the wire; this approach requires at least the sender to touch
        every byte of the ULPDU
      - using a framming pattern, combining it with the ULPDU length
        and making sure that ULPDU starts at the beginning of a TCP
        frame; this approach does not require the sender of receiver
        to touch every byte but implies a slightly changed TCP imple-
        mentation and the absence of middle boxes that split or com-
        bine TCP segments.

   The first approach is a modern version of the old byte/bit stuffing
   techniques, with a twist - the overhead is constant (a scheme similar
   to [COBS]). Its main drawback is that it requires the sender to exam-
   ine (touch) every piece of payload.

   The second approach - as presented until now (see [F1ULP]) uses a
   framing pattern selected for "the life" of the connection and an addi-
   tional piece of user data (the payload length) as a safeguard against
   a data pattern looking like a framing pattern and appearing at the
   beginning of a frame, being mistaken for a framing pattern.

   With all the intuitive attractiveness of this technique (very simple
   and low overhead) my concern with this techniques is related to the
   following "line-of-reasoning":

       - this is a "statistical approach based on a very low probabil-
         ity of appearance of the framing pattern in the data payload
         (and even lower at a specific position) BUT
       - no statistical statement can be made about the probability of
         mistaking a regular data pattern as being a framing pattern (+
         length).

   A closer examination will reveal that since we are matching a random-
   constant to a very long data stream (potential infinite) this proba-
   bility can get uncomfortably close to 1. The weakens of the technique
   is that the error probability is proportional to the stream length.

   This draft is an attempt to build an "no-data-touch" scheme that:

       - Has an extremely low probability for normal data to be mis-
         taken as framing and thus can be used even in the absence of a
         protocol stack that enforces alignment (probability that could
         be made substantially lower than the probability of a data
         error passing undetected)
       - The probability is constant and independent of the stream
         length

2. Scheme development

   In order to reduce the probability of a regular data being mistaken
   for a framing pattern and keep this probability constant - we will
   introduce a "salt" (a random number) and a digest in the framing
   header. The framing header will thus include a framing-pattern, a salt
   and a digest (e.g. CRC) as outlined bellow:

```
    Byte /     0         |       1         |       2         |       3         |
        /                |                 |                 |                 |
       |7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|
        +--------------+--------------+--------------+--------------+
     0| Framing Pattern                                              |
        +--------------+--------------+--------------+--------------+
     4| Salt                                                         |
        +--------------+--------------+--------------+--------------+
     8/ Header-Digest (at least 4 bytes)                             /
     +/                                                              /
        +--------------+--------------+--------------+--------------+
     x/ ULPDU                                                        /
     +/                                                              /
        +--------------+--------------+--------------+--------------+
```

Provided that the digest algorithm is carefully selected (is a good
randomizer) for a 4 byte digest the probability of such a sequence
appearing in the ULPDU payload can be estimated as $(2^{-64})*L$ where L
is the payload length.

The same probability holds for the entire life of the stream as we
insert in the framing header a random element that is different for
every frame.

Please observe that in the previously discussed framing Non-Data-
Touch (NDT) schemes employed using only a framing pattern the proba-
bility of the framing pattern appearing in the stream is increasing as
the stream length increases as the same pattern is used for the whole
stream (using the length does not improve on this as length is fairly
regular).

The above scheme can be extended to include elements that safeguard
against misuse by introducing "virtual fields" that are never sent on
the wire - a shared (session) secret key, a length a sequence number
etc.. Those are not needed if the channel is secured by some other
means (TLS, IPsec).

A shared secret key can be obtained by the communicating parties at
the ULP communication channel establishment (e.g., by a Diffie-Helman
exchange) and is included in the digest calculation.

The format of the framing header remains the same but the digest
includes now a shared secret of length and content known to the commu-
nicating parties as illustrated bellow:

```
    Byte /      0          |       1          |       2          |       3          |
       /                   |                  |                  |                  |
       |7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|
       +--------------+--------------+--------------+--------------+
     0| Framing Pattern                                           |
       +--------------+--------------+--------------+--------------+
     4| Salt                                                      |
       +--------------+--------------+--------------+--------------+
      | Shared Secret and/or other elements (not on the wire)     |
       +--------------+--------------+--------------+--------------+
     8/ Header-Digest (at least 4 bytes)                          /
     +/                                                           /
       +--------------+--------------+--------------+--------------+
     x/ ULPDU                             /                       /
     +/                                                           /
       +--------------+--------------+--------------+--------------+
```
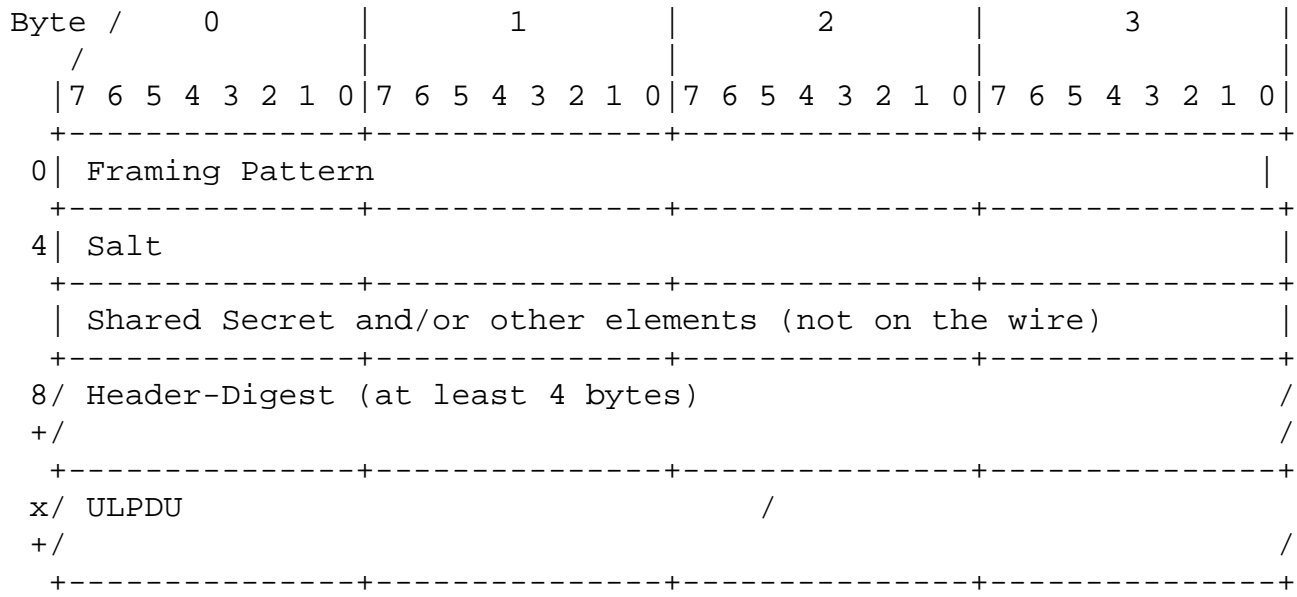
3. Optimizations

If the ULP header includes a digest the framing header may be included
in the ULP header and the digest cover both. The format of the com-
bined header is illustrated bellow:

```
Byte /      0        |        1        |        2        |        3        |
   /                 |                 |                 |                 |
  |7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|7 6 5 4 3 2 1 0|
   +--------------+--------------+--------------+--------------+
 0| Framing Pattern                                              |
   +--------------+--------------+--------------+--------------+
 4| Salt                                                         |
   +--------------+--------------+--------------+--------------+
  | Shared Secret and other (not on the wire)                    |
   +--------------+--------------+--------------+--------------+
 8/ ULP Header                                                   /
 +/                                                              /
   +--------------+--------------+--------------+--------------+
 x/ Header-Digest (at least 4 bytes)                            /
 +/                                                              /
   +--------------+--------------+--------------+--------------+
 y/ rest-of-ULPDU                                               /
 +/                                                              /
   +--------------+--------------+--------------+--------------+
```
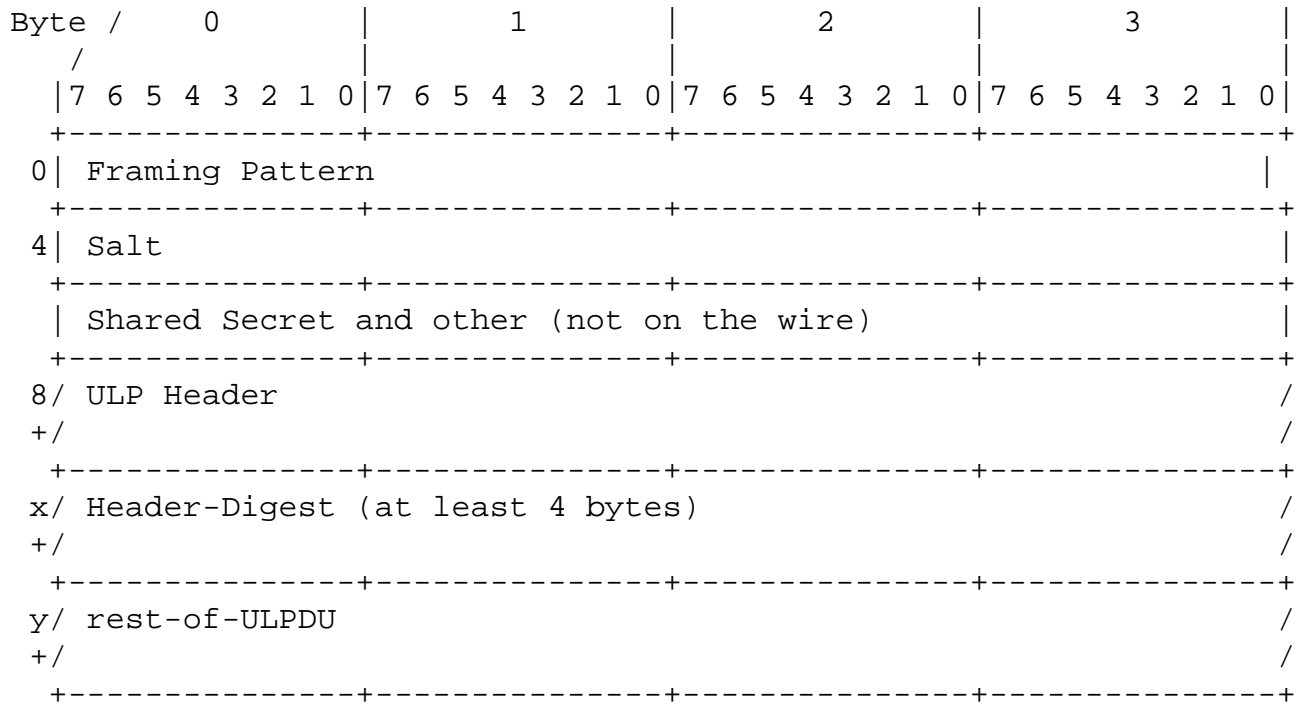
4. Conclusions

In this draft we suggest a NDT framing scheme with a good (low) prob-
ability of "misdetection" - a scheme that is at statistically better
than the previously proposed NDT schemes in that it holds for streams
of indeterminate lengths. It can be used to built a separate framing
header or incorporated in existing ULP headers.

References and Bibliography

        [COBS] S. Cheshire and M. Baker, Consistent Overhead Byte Stuff-
          ing, IEEE Transactions on Networking, April 1999.
        [iSCSI] J. Satran & all, http://www.ietf.org/internet-drafts/
          draft-ietf-ips-iscsi-10.pdf
        [F1ULP] S. Bailey & all, http://www.ietf.org/internet-drafts/
          draft-ietf-tsvwg-tcp-ulp-frame-01.txt

Authors' Addresses

        Julian Satran
        IBM, Haifa Research Lab
        Mount Carmel
        Haifa 31905, Israel
        Phone +972.4.829.6264
        E-mail: Julian_Satran@vnet.ibm.com