                    PPP Gandalf FZA Compression Protocol

Status of this Memo

   This memo provides information for the Internet community. It does
   not specify an Internet standard. Distribution of this memo is
   unlimited.

Abstract

   The Point-to-Point Protocol (PPP) [1] provides a standard method for
   transporting multi-protocol datagrams over point-to-point links.

   The PPP Compression Control Protocol [2] provides a method to
   negotiate and utilize compression protocols over PPP encapsulated
   links.

   This document describes the use of the Gandalf FZA data compression
   algorithm [3] for compressing PPP encapsulated packets.

Table of Contents

Barbir, Carr & Simpson
[Page i]

1. Introduction

   FZA is a high performance LZ [4] derivative that maximizes
   compression at the expense of memory and CPU. Compression
   performance can be adjusted based on CPU and memory available.

   Multiple PPP packets can be combined in a single compressed
frame, or
   a single PPP packet can be spread across multiple frames.

1.1. Licensing

   Source and object licenses are available on a non-discriminatory
   basis for either a royalty or fixed price arrangement. Patent
   indemnity is included with the license.

2. FZA Packets

   Before any FZA packets may be communicated, PPP must reach the
   Network-Layer Protocol phase.

   When the Compression Control Protocol (CCP) has reached the
Opened
   state, and FZA is negotiated as the primary compression
algorithm,
   the PPP Protocol field indicates type hex 00FB (link compressed
   datagram), or type hex 00FD (compressed datagram).

   The maximum length of the FZA datagram transmitted over a PPP
link is
   the same as the maximum length of the Information field of a PPP
   encapsulated packet.

   Padding

The FZA packets require the negotiation of the Self-
Describing-
        Padding Configuration Option [5] at LCP Link Establishment.

   Reliability and Sequencing

        The FZA algorithm expects a reliable link, as described in
"PPP
        Reliable Transmission" [6].

        FZA expects the packets to be delivered in sequence.

   Data Expansion

        The maximum expansion of Gandalf FZA is 2:1. However, typical
        expansion on pre-compressed data is 1.01:1. Expanded data is
sent
        to maintain the integrity of the compression history.

        When the expansion exceeds the size of the peer's Maximum
Receive
        Unit for the link, the expanded packet is sent in multiple PPP
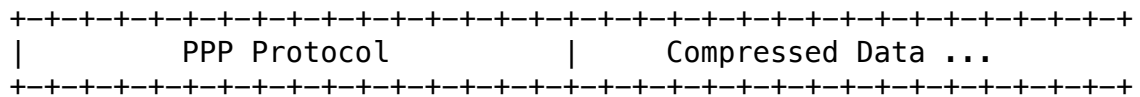   frames. The compressed data contains an indication of the end
of
        the original packet.

2.1. Packet Format

   A summary of the Gandalf FZA packet format is shown below. The
   fields are transmitted from left to right.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          PPP Protocol         |     Compressed Data ...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

   PPP Protocol

      One or two octets. The PPP Protocol field is described in the
      Point-to-Point Protocol Encapsulation [1].

      Type 00FD is used when the PPP multilink protocol is not used,
      and/or "inside" a multilink bundle. Type 00FB is used
"outside"
      multilink, to compress independently on individual links of a
      multilink bundle. This value MAY be compressed when LCP
      Protocol-Field-Compression is negotiated.

   Compressed Data

      One or more octets. The compressed PPP encapsulated
packet(s).

      Prior to compression, the uncompressed data begins with the
      original PPP Protocol number. This value MAY be compressed
when
      LCP Protocol-Field-Compression is negotiated.

      The original Protocol number is followed by the original
      Information field. The length of the original Information
field
      before compression MUST NOT exceed the link Maximum Receive
Unit
      (MRU).

      PPP Link Control Protocol packets MUST NOT be sent within
      compressed data.

3. Configuration Option Format

   Description

      The CCP Gandalf-FZA Configuration Option negotiates the use of
      Gandalf FZA on the link. By default or ultimate disagreement,
no
      compression is used.

   A summary of the Gandalf-FZA Configuration Option format is shown
   below. The fields are transmitted from left to right.

      +-+
      | Type | Length | History | Version ...
      +-+

   Type

      19

   Length

      >= 3

   History

      One octet. The History field specifies the maximum size of
the
      compression history in powers of 2. Valid values range from
12 to
      15.

      The peer is not required to send as many histories as the
      implementation indicates that it can accept.

   Version

      Zero or more octets of additional configuration information.
Any
      implementation that does not implement this information MUST
send
      a Configure-Nak without this field.

      The Version field is not present for FZA.

      The Version field is a single octet containing the value 1 for
      FZA+.

Security Considerations

   Security issues are not discussed in this memo.

Acknowledgements

   FZA was developed by David Carr while at Gandalf Data Limited.

   FZA+ was an improvement by Abbie Barbir.

   Editting and formatting by William Simpson.

References

   [1]   Simpson, W., Editor, "The Point-to-Point Protocol (PPP)",
STD
          51, RFC 1661, DayDreamer, July 1994.

   [2]   Rand, D., "The PPP Compression Control Protocol (CCP)", RFC
          1962, Novell, June 1996.

   [3]   Barbir, A., "A New Fast Approximate Arithmetic Coder",
          Proceedings of IEEE 28th SouthEastern Symposium on Systems
          Theory (SSST), Baton Rouge, Louisiana, pages 482-486, April
          1996.

   [4]   Lempel, A. and Ziv, J., "A Universal Algorithm for
Sequential
          Data Compression", IEEE Transactions On Information Theory,
          Vol. IT-23, No. 3, May 1977.

   [5]   Simpson, W., Editor, "PPP LCP Extensions", RFC 1570,
          DayDreamer, January 1994.

   [6]   Rand, D., "PPP Reliable Transmission", RFC 1663, Novell,
July
          1994.

Contacts

Licensing queries should be directed to:

Michael Williams
Director of Business Development
Gandalf Data Limited
130 Colonnade Road South
Napean, Ontario, Canada K2E 7M4
(613) 274-6500 ext 6575

   Comments should be submitted to the ietf-ppp@merit.edu mailing
list.

   This document was reviewed by the Point-to-Point Protocol Working
   Group of the Internet Engineering Task Force (IETF).

   The working group can be contacted via the current chair:

      Karl Fox
      Ascend Communications
      3518 Riverside Drive, Suite 101
      Columbus, Ohio 43221

         karl@MorningStar.com
         karl@Ascend.com

   Questions about this memo can also be directed to:

      Abdulkader Barbir
      Gandalf Data Limited
      130 Colonnade Road South
      Napean, Ontario, Canada K2E 7M4
      (613) 274-6500 ext 8550

         abarbir@gandalf.ca

   Questions about this memo should not be directed to:

      Dave Carr

Newbridge Networks Corporation
600 March Road
P.O. Box 13600
Kanata, Ontario, Canada, K2K 2E6

          dcarr@newbridge.com

William Allen Simpson
DayDreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071

          wsimpson@UMich.edu
          wsimpson@GreenDragon.com (preferred)