PPP

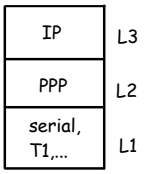
This lecture describes the Point-to-Point Protocol, a link-layer (Layer 2) protocol that is often used to encapsulate IP packets when they are transmitted over data links that were not designed for packet transmission. This includes serial ("RS-232"), T-carrier (e.g. T1), and SONET links.
After this lecture you should be able to: encapsulate a packet using PPP framing including adding and removing escape characters, generate a PPP frame for an IP packet, and decide whether a configuration item would be negotiated by the LCP, NCP or neither.

Introduction

Many data links transmit continuous sequences of bits or characters and have no way to mark the start or end of a packet. This includes RS-232 serial interfaces and related links such as dial-up modem connections and links that were originally designed to carry TDM PCM voice data such as T1 and SONET/SDH links.

In order to transmit groups of bytes such as an an IP packet[1] over such a link, the data link layer has to, at a minimum, be able to separate bits or bytes into frames. Other features such as support for indicating the network-layer (L3) protocol being encapsulated and error detection can greatly increase the usefulness and efficiency of the link layer protocol.

PPP is a link-layer (L2) protocol that provides these features over point-to-point *full-duplex byte-oriented* physical layers.



Exercise 1: Can PPP be used over a serial link configured for 9600,7,N,1 (9600 bps, 7 data bits, no parity, 1 stop bit)? Can it be used to encapsulate IP frames for broadcast to multiple users?

PPP has many optional extensions including authentication, encryption, compression, and configuration of the network-layer parameters (i.e. replacing DHCP).

PPP is sometimes also used over packet-based links such as Ethernet (PPPoE) and ATM (PPPoA) because of these additional services.

[1] The convention is to use the term "frame" for Layer 2 and "packet" for Layer 3.

PPP and it's many optional features are defined in several IETF RFC's including RFC1661 and RFC1662.

PPP Encapsulation

PPP uses a simple subset of HDLC to encapsulate frames.

PPP delimits frames by using a "flag" character, 0x7e. An "escape" character, 0x7d, is also defined and is used as the first byte of a two-byte sequence that is replaced by a single byte at the receiving end. In addition to escaping the flag and escape characters, other characters (e.g. control characters) can also be escaped if they cannot be transmitted transparently over the channel. This is done by XOR'ing the byte with 0x20. For example, the flag character is transmitted as the pair of characters 0x7d 0x5e.

Exercise 2: What sequence of characters is transmitted when an escape character appears in the frame? What range of characters is transmitted when escaping unprintable ASCII characters (those between 0x00 and 0x2f)?

PPP Framing

The following ASCII diagram, taken from the PPP RFC, shows the framing of a PPP packet:

```
+----------+----------+----------+
|   Flag   | Address  | Control  |
| 01111110 | 11111111 | 00000011 |
+----------+----------+----------+
+----------+-------------+---------+
| Protocol | Information | Padding |
| 8/16 bits|      *      |    *    |
+----------+-------------+---------+
+----------+----------+-----------------
|   FCS    |   Flag   | Inter-frame Fill
|16/32 bits| 01111110 | or next Address
+----------+----------+-----------------
```

The address and control fields are fixed to the values above (0xff and 0x03) and are included for

compatibility with HDLC. The protocol field specifies the L3 protocol. It is 8 bits or 16 bits if the LS bit of the first byte is zero. For example, 0x21 is IPv4 while 0xc021 is used for link control protocol (LCP, used by PPP). The Frame Check Sequence (FCS) is a 16-bit (optionally 32-bit) CRC. Padding is optional and requires that the L3 protocol have a way to distinguish between data and padding.

Exercise 3: What are the first four bytes of a PPP-encapsulated IP frame? What bytes would be transmitted for an IP address field with value 127.126.0.1? If the IP frame was 60 bytes long, no bytes needed to be escaped and the default PPP link options were being used, what would be the length of the PPP frame? Can an encapsulated IP frame distinguish between data and padding?

Network and Link Layers

The Link Layer (L2) is responsible for transferring frames over the local network. A common link layer is Ethernet. The Network Layer (L3) is responsible for transporting frames between local networks. IP is the most common network layer.

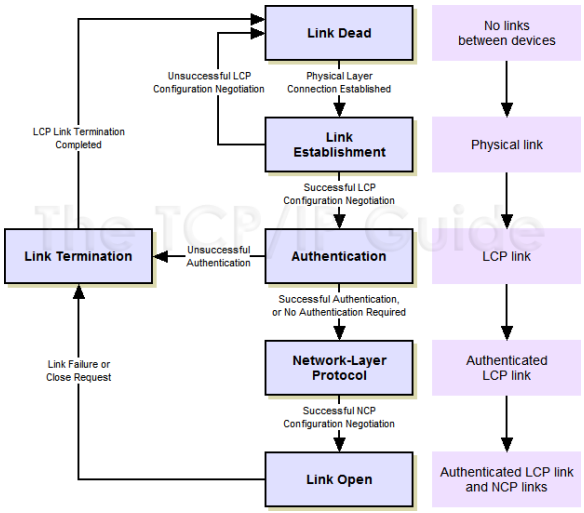
PPP includes two control protocols, LCP and NCP that are used to configure many aspects of the link and network layers. Since PPP is the link layer protocol, LCP deals with configuration of PPP. NCP can be used to configure various L3 protocols, but is mainly used for configuration of the IP protocol.

Link Control Protocol

The LCP defines a state machine that controls the operation of PPP. A PPP connection can be in various states as shown in the diagram below[2]. Transitions between states happen as a result of LCP or NCP events (e.g. authentication success or timer timeout).

Some of the features that can be negotiated by the LCP include:

- authentication using various protocols (PAP, CHAP, EAP)
- address/control and protocol field compression
- maximum frame length (MTU, MRU)
- characters that require escaping
- many other options including multi-link PPP (MLPPP), an option that allows multiple PPP links to be combined ("bonded") into one higher-speed L3 link



Network Control Protocol

Each L3 protocol has its own PPP protocol that is used to negotiate L3 features. For IP this is IPCP (IP Control Protocol) defined in RFC 1332. The main purpose of IPCP is to set up IP addresses.

Exercise 4: Would LCP or NCP be used to negotiate compression (e.g. zip)? To configure a DNS server? To set the baud rate on the serial interface?

Alternative Encapsulation Protocols

There are other L2 encapsulation protocols. HDLC (High-Level Data Link Control) is the basis for many other framing protocols including PPP and Frame Relay.

HDLC is often used to provide framing on links that do not have byte framing. This is done by using bit stuffing (a stuffed 0 bit after 5 consecutive 1 bits) rather than the byte stuffing described above.

Frame Relay is another variant of HDLC which is (was) primarily used by telephone companies to provide data service over various types of TDM links.

The choice of L2 encapsulation protocol is primarily determined by the features required (e.g. authentication, L3 configuration) and by which encapsulation protocols are supported by the peer device (typically a router).

[2] From this web page