24

PPP Suite

The Point-to-Point Protocol (PPP) suite includes the following protocols, in addition to PPP:

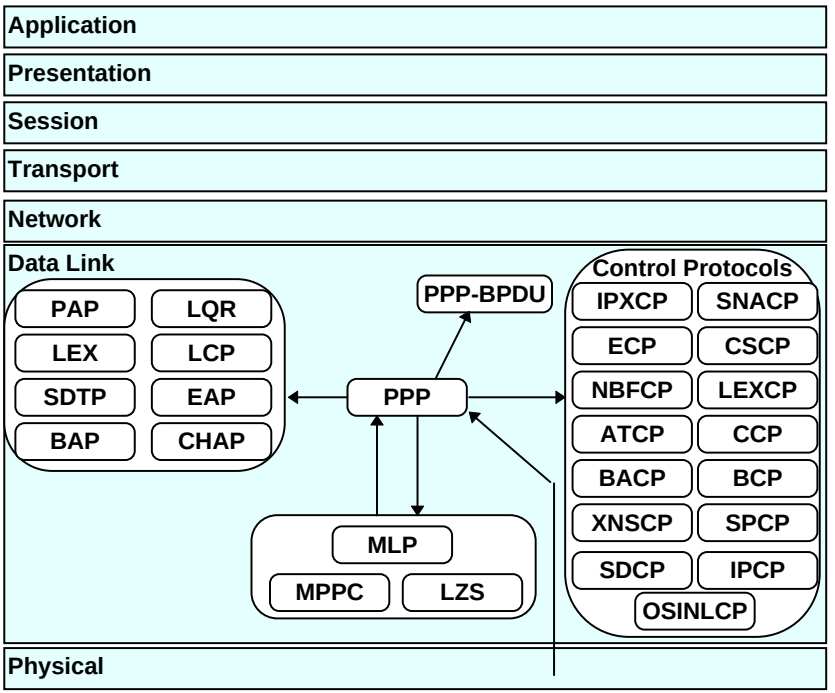
- MLP: Multilink PPP.
- PPP-BPDU: PPP Bridge Protocol Data Unit.
- PPPoE: PPP over Ethernet.
- BAP: Bandwidth Allocation Protocol.
- BSD.
- CHAP: Challenge Handshake Authentication Protocol.
- DESE: Data Encryption Standard Encryption.
- EAP: Extensible Authentication Protocol.
- LCP: Link Control Protocol.
- LEX: LAN Extension Interface Protocol.
- LQR: Link Quality Report.
- PAP: Password Authentication Protocol.

PPP control protocols

- ATCP: AppleTalk Control Protocol.
- BACP: Bandwidth Allocation Control Protocol.
- BCP: Bridging Control Protocol.

- BVCP: PPP Banyan Vines Control Protocol.
- CCP: Compression Control Protocol.
- DNCP: PPP DECnet Phase IV Control Protocol.
- ECP: Encryption Control Protocol.
- IPCP: IP Control Protocol.
- IPv6CP: IPv6 Control Protocol.
- IPXCP: IPX Control Protocol.
- LEXCP: LAN Extension Interface Control Protocol.
- NBFCP: PPP NetBios Frames Control Protocol.
- OSINLCP: OSI Network Layer Control Protocol.
- SDCP: Serial Data Control Protocol.
- SNACP: SNA PPP Control Protocol.

The following diagram shows the PPP suite in relation to the OSI model :



PPP protocol suite in relation to the OSI model

PPP

RFC 1548 http://www.cis.ohio-state.edu/htbin/rfc/rfc1548.html
RFC 1661 http://www.cis.ohio-state.edu/htbin/rfc/rfc1661.html
RFC 1662 http://www.cis.ohio-state.edu/htbin/rfc/rfc1662.html

PPP (Point-to-Point Protocol) is designed for simple links which transport packets between two peers. These links provide full-duplex simultaneous bi-directional operation and are assumed to deliver packets in order. PPP provides a common solution for the easy connection of a wide variety of hosts, bridges and routers.

The structure of the PPP header is shown in the following illustration:

Address	Control	Protocol	Information	FCS
1 byte	1 byte	2 bytes	variable	2 bytes

PPP header structure

Address
HDLC broadcast address. PPP does not assign individual station addresses. The value of this field is always set to FF Hex.

Control
HDLC command for Unnumbered Information (UI) with the Poll/Final bit set to zero. The value of this field is always set to 03 Hex. Frames containing any other value in this field are discarded.

Protocol
Identifies the encapsulated protocol within the Information field of the frame.

Information
Higher-level protocol data.

FCS
Value of the frame checksum calculation. PPP verifies the contents of the FCS field upon receipt of the packet.

MLP (PPP Multilink)

RFC 1717 http://www.cis.ohio-state.edu/htbin/rfc/rfc1717.html
RFC 1990 http://www.cis.ohio-state.edu/htbin/rfc/rfc1990.html

Multilink is based on a PCP option negotiation that permits a system to indicate to its peer that it is capable of combining multiple physical links into a "bundle". Only under exceptional conditions would a given pair of systems require the operation of more than one bundle connecting them.

Multilink is negotiated during the initial LCP option negotiation. A system indicates to its peer that it is willing to do multilink by sending the multilink option as part of the initial LCP option negotiation. This negotiation indicates the following:

1. The system offering the option is capable of combining multiple physical links into one logical link.
2. The system is capable of receiving upper layer PDUs fragmented using the multilink header and reassembling the fragments back into the original PDU for processing.
3. The system is capable of receiving PDUs of size N octets where N is specified as part of the option even if N is larger than the maximum receive unit (MRU) for a single physical link.

Using the PPP Multilink protocol, network protocol packets are first encapsulated (but not framed) according to normal PPP procedures; large packets are broken up into multiple segments sized appropriately for the multiple physical links. A new PPP header consisting of the Multilink protocol identifier, and the Multilink header is inserted before each section. Thus, the first fragment of a multilink packet in PPP will have two headers: one for the fragment, followed by the header for the packet itself.

PPP Multilink fragments are encapsulated using the protocol identifier 0x00-0x3d. Following the protocol identifier is a 4-byte header containing a sequence number, and two 1-byte fields indicating whether the fragment begins or terminates a packet. After negotiation of an additional PPP LCP option, the 4-byte header may be optionally replaced by a 2-byte header with a 12-bit sequence space. Address/Control and Protocol ID compression are assumed to be in effect.

The following is the format for the long sequence number fragment:

PPP Header	Address 0xff								Control 0x03
	PID (H) 0x00								PID (L) 0x3d
MP Header	B	E	0	0	0	0	0	0	Sequence number
	Sequence number (L)								
	Fragment data . . .								
PPP FCS	FCS								

PPP Multilink long sequence number fragment

The following is the format for the short sequence number fragment:

PPP Header	Address 0xff								Control 0x03
	PID (H) 0x00								PID (L) 0x3d
MP Header	B	E	0	0	Sequence Number				
	Fragment Data . . .								
PPP FCS	FCS								

PPP Multilink short sequence number fragment

PID
Protocol ID compression.

B
Beginning fragment bit. A 1-bit field which is set to 1 on the first fragment derived from a PPP packet and set to 0 for all other fragments from the same PPP packet.

E
Ending fragment bit. A 1-bit field which is set to 1 on the last fragment and set to 0 for all other fragments. A fragment may have both the beginning and ending fragment bits set to 1.

Sequence number

24-bit or 12-bit number that is incremented for every fragment transmitted. By default, the sequence field is 24 bits long, but can be negotiated to be only 12 bits with an LCP configuration option.

0

A reserved field between the ending fragment bit and the sequence number. This field is not used at present and must be set to zero. It is 2 bits long when short sequence numbers are used; otherwise it is 6 bits in length.

FCS

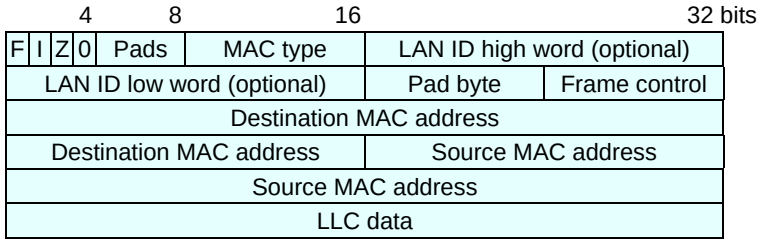
Frame check sequence. This value is inherited from the normal framing mechanism from the member link on which the packet is transmitted.

PPP-BPDU

RFC 1638 http://www.cis.ohio-state.edu/htbin/rfc/rfc1638.html

There exist two basic types of bridges: those that interconnect LANs directly, called Local Bridges, and those that interconnect LANs via an intermediate WAN medium such as a leased line, called Remote Bridges. The PPP-BPDU (Bridge Protocol Data Unit) is used to connect Remote Bridges.

The format of the PPP-BPDU packet is shown in the following illustration:



PPP-BPDU packet structure

F

F flag, set if the LAN FCS field is present.

I

I flag, set if the LAN ID field is present.

Z

Z flag, set if IEEE 802.3 pad must be zero filled to minimum size.

0

0 flag reserved, must be zero.

Pads

Any PPP frame may have padding inserted in the Optional Data Link Layer Padding field. This number tells the receiving system how many pad octets to strip off.

MAC type
Values of the MAC Type field.
1 Bridge-Identification.
2 Line-Identification.
3 MAC-Support.
4 Tinygram-Compression.
5 LAN-Identification.
6 MAC-Address.
7 Spanning-Tree-Protocol.

LAN ID
Optional 32-bit field that identifies the Community of LANs which may be interested in receiving this frame. If the LAN ID flag is not set, then this field is not present and the PDU is four octets shorter.

Frame control
On 802.4, 802.5 and FDDI LANs, there are a few octets preceding the Destination MAC Address, one of which is protected by the FCS. The MAC Type of the frame determines the contents of the Frame Control field. A pad octet is present to provide a 32-bit packet alignment.

Destination MAC address
As defined by the IEEE. The MAC Type field defines the bit ordering.

Source MAC address
As defined by the IEEE. The MAC Type field defines the bit ordering.

LLC data
This is the remainder of the MAC frame which is (or would be were it present) protected by the LAN FCS.

PPPoE

RFC 2516 http://www.cis.ohio-state.edu/htbin/rfc/rfc2516.html

PPPoE is a method for transmitting PPP over Ethernet. It provides the ability to connect a network of hosts over a simple bridging access device to a remote access concentrator. With this model, each host utilizes its own PPP stack and the user is presented with a familiar user interface. Access control, billing and type of service can be done on a per-user, rather than a per-site, basis.

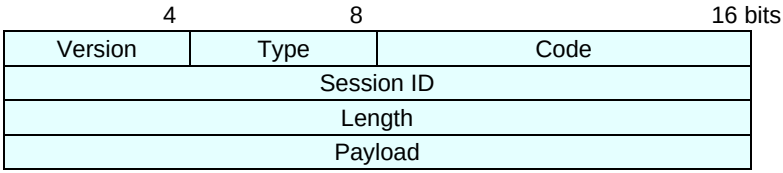
To provide a point-to-point connection over Ethernet, each PPP session must learn the Ethernet address of the remote peer, as well as establish a unique session identifier. PPPoE includes a discovery protocol that performs this.

PPPoE has two distinct stages. There is a discovery stage and a PPP session stage. When a host wishes to initiate a PPPoE session, it must first perform discovery to identify the Ethernet MAC address of the peer and establish a PPPoE SESSION_ID. While PPP defines a peer-to-peer relationship, discovery is inherently a client-server relationship. In the discovery process, a host (the client) discovers an access concentrator (the server). Based on the network topology, there may be more than one access concentrator that the host can communicate with. The discovery stage allows the host to discover all access concentrators and then select one. When discovery completes successfully, both the host and the selected access concentrator have the information they then use to build their point-to-point connection over Ethernet.

The discovery stage remains stateless until a PPP session is established. Once a PPP session is established, both the host and the access concentrator must allocate the resources for a PPP virtual interface.

The EtherType field in the Ethernet frame is set to either 0x8863 for the discovery stage or 0x8864 for the PPP session stage.

The Ethernet payload for PPPoE is as shown in the following illustration:



Ethernet payload for PPPoE

Version
Specifies the version number: 0x1 for the current version of PPPoE (RFC 2516).

Type
Set to 0x1 for the current version of PPPoE (RFC 2516).

Code
Value of the code depends on the packet sent. Values may be as follows:

Packet	***Code***
Discovery stage:	
Active Discovery Initiation (PADI)	0x09
Active Discovery Offer (PADO)	0x07
Active Discovery Request (PADR)	0x19
Active Discovery Session-confirmation (PADS)	0x65
Active Discovery Terminate (PADT)	0xa7
PPP Session Stage	0x00

Session ID
Unsigned value in network byte order which defines a PPP session along with the Ethernet source and destination addresses. 0xffff is reserved for future use.

Length
Length of the PPPoE payload, not including the Ethernet or PPPoE headers.

BAP

RFC 2125 http://www.cis.ohio-state.edu/htbin/rfc/rfc2125.html

The Bandwidth Allocation Protocol (BAP) can be used to manage the number of links in a multi-link bundle. BAP defines datagrams to coordinate adding and removing individual links in a multi-link bundle, as well as specifying which peer is responsible for various decisions regarding managing bandwidth during a multi-link connection.

The format of the BAP packet is shown in the following illustration:

Type	Length	Data
1 byte	1 byte	variable

BAP packet structure

Type

One-octet field which indicates the type of the BAP Datagram Option. This field is binary coded hexadecimal.

1 Call-Request.
2 Call-Response.
3 Callback-Request.
4 Callback-Response.
5 Link-Drop-Query-Request.
6 Link-Drop-Query-Response.
7 Call-Status-Indication.
8 Call-Status-Response.

Length

One-octet field which indicates the length of this BAP option including the Type, Length and Data fields.

Data

Zero or more octets and contains information specific to the BAP option. The format and length of the Data field is determined by the Type and Length fields.

BSD

RFC 1977: http://www.cis.ohio-state.edu/htbin/rfc/rfc1977.html

BSD is the freely and widely distributed UNIX compress command source. It provides the following features:

- Dynamic table clearing when compression becomes less effective.
- Automatic turning off of compression when the overall result is not smaller than the input.
- Dynamic choice of code width within predetermined limits.
- Heavily used for many years in networks, on modem and other point-to-point links to transfer net news.
- An effective code width, requires less than 64 Kbytes of memory on both send and receive.

Before any BSD Compress packets may be communicated, PPP must reach the network layer protocol phase, and the CCP control protocol must reach the opened state. Exactly one BSD Compress datagram is encapsulated in the PPP information field, where the PPP protocol field contains 0xFD, or 0xFB. 0xFD is used above MLP. 0xFB is used below MLP to compress independently on individual links of a multilink bundle. The maximum length of the BSD Compress datagram transmitted over a PPP link is the same as the maximum length of the information field of a PPP encapsulated packet. Only packets with PPP protocol numbers in the range 0x0000 to 0x3FFF and neither 0xFD, nor 0xFB are compressed. Other PPP packets are always sent uncompressed. Control packets are infrequent and should not be compressed for robustness.

CHAP

RFC 1334 http://www.cis.ohio-state.edu/htbin/rfc/rfc1334.html

Challenge Handshake Authentication Protocol (CHAP) is used to periodically verify the identity of the peer using a 3-way handshake. This is done upon initial link establishment and may be repeated any time after the link has been established.

Exactly one CHAP packet is encapsulated in the Information field of a PPP data link layer frame where the protocol field indicates type hex c223. The structure of the CHAP packet is shown in the following illustration:

Code	Identifier	Length	Data . . .
1 byte	1 byte	2 bytes	variable

CHAP packet structure

Code

Identifies the type of CHAP packet. CHAP codes are assigned as follows:

1 Challenge.
2 Response.
3 Success.
4 Failure.

Identifier

Aids in matching challenges, responses and replies.

Length

Length of the CHAP packet including the Code, Identifier, Length and Data fields.

Data

Zero or more octets, the format of which is determined by the Code field. The format of the Challenge and Response data fields is shown in the following illustration:

Value size	Value	Name
1 byte		1 byte

CHAP Challenge and Response data structure

Value size
Indicates the length of the Value field.

Value
Challenge value is a variable stream of octets which must be changed each time a challenge is sent.
Response value is the one-way hash calculated over a stream of octets consisting of the Identifier, followed by the "secret" and the Challenge value.

Name
Identification of the system transmitting the packet.

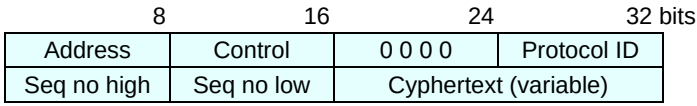
For Success and Failure, the data field contains a variable message field which is implementation dependent.

DESE

RFC 1969: http://www.cis.ohio-state.edu/htbin/rfc/rfc1969.html

The DES (Data Encryption Standard) Encryption algorithm is a well studied, understood and widely implemented encryption algorithm which was designed for efficient implementation in hardware. DESE is an option of ECP which indicates that the issuing implementation is offering to employ the DES encryption for decrypting communications on the link and may be thought of as a request for its peer to encrypt packets in this manner.

The format of the DESE packet is shown in the following illustration:



DESE packet structure

The address and control fields are as described for PPP.

Protocol ID

Values may be 0x53 or 0x55; the latter indicates that cyphertext includes headers for MLP and requires that the Individual Link Encryption Control Protocol has reached the opened state. The leading zero may be absent if the PPP Protocol Field Compression (PFC) option has been negotiated.

Seq no high/low

Sequence numbers are 16-bit numbers which are assigned by the encryptor sequentially starting with 0 (for the first packet transmitted once ECP has reached the open state).

Cyphertext

Generation of encrypted data is described in the DESE standard.

EAP

draft-ietf-pppext-eap-auth-03.txt

The Extensible Authentication Protocol (EAP) is a PPP extension that provides support for additional authentication methods within PPP.

The format of the EAP header is shown in the following illustration:

Code	Identifier	Length	Type	Data
1 byte	1 byte	2 bytes	1 byte	variable

EAP packet structure

Code

Decimal value which indicates the type of EAP packet:

1 Request.
2 Response.
3 Success.
4 Failure.

Identifier

Aids in matching responses with requests.

Length

Indicates the length of the EAP request and response packets including the Code, Identifier, Length, Type, and Data fields. Octets in the packet outside the range of the Length field are treated as Data Link Layer padding and are ignored on reception.

Type

Indicates the EAP type. The following types are supported: KEA validate, KEA public, GTC, OTP, MD5, NQK, Notification, Identity.

LCP

RFC 1570 http://www.cis.ohio-state.edu/htbin/rfc/rfc1570.html
RFC 1661 http://www.cis.ohio-state.edu/htbin/rfc/rfc1661.html

In order to be sufficiently versatile to be portable to a wide variety of environments, PPP provides the Link Control Protocol (LCP) for establishing, configuring and testing the data link connection. LCP is used to automatically agree upon the encapsulation format options, handle varying limits on sizes of packets, detect a looped-back link and other common misconfiguration errors, and terminate the link. Other optional facilities provided are authentication of the identity of its peer on the link, and determination when a link is functioning properly and when it is failing.

The format of the LCP packet is shown in the following illustration:

Code	Identifier	Length	Data
1 byte	1 byte	2 bytes	variable

LCP packet structure

Code

Decimal value which indicates the type of LCP packet:

1 Configure-Request.
2 Configure-Ack.
3 Configure-Nak.
4 Configure-Reject.
5 Terminate-Request.
6 Terminate-Ack.
7 Code-Reject.
8 Protocol-Reject.
9 Echo-Request.
10 Echo-Reply.
11 Discard-Request.
12 Link-Quality Report.

Identifier

Decimal value which aids in matching requests and replies.

Length
Length of the LCP packet, including the Code, Identifier, Length and Data fields.

Data
Variable length field which may contain one or more configuration options. The format of the LCP configuration options is as follows:

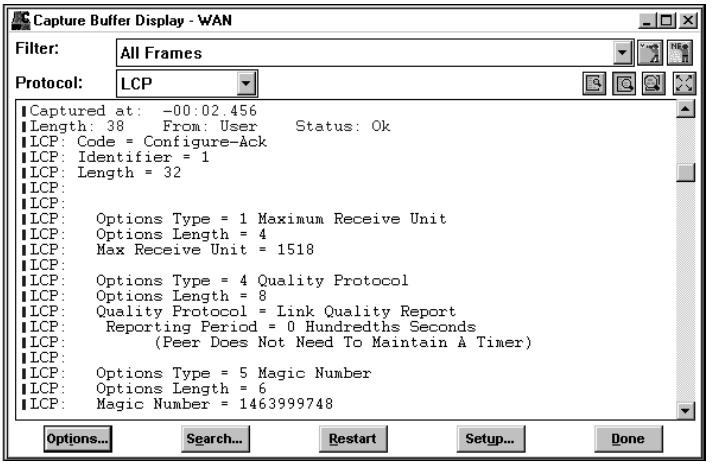
Type	Length	Data

LCP configuration options

Type
One-byte indication of the type of the configuration option.

Length
Length of the configuration option including the Type, Length and Data fields.

Data
Value of the Data field.



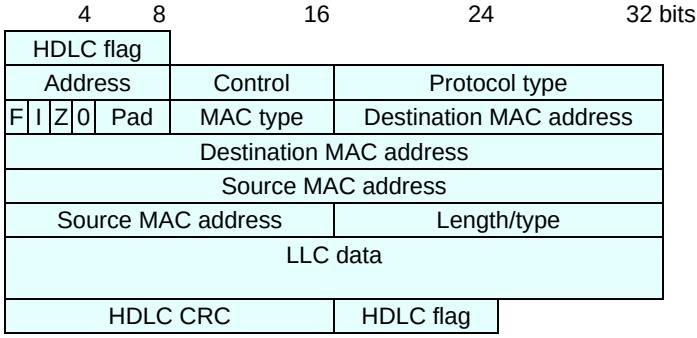
LCP decode

LEX

RFC 1841 http://www.cis.ohio-state.edu/htbin/rfc/rfc1841.html

A LAN extension interface unit is a hardware device installed at remote sites (such as a home office or small branch office) that connects a LAN across a WAN link to a router at a central site. To accommodate this LAN extension interface architecture, a PPP Network Control Protocol was developed: the LAN extension interface protocol, PPP-LEX. The basic functionality of LEX is to encapsulate LAN extension interface control and data packets. Consequently packets can be control packets or data packets. Control packets are described under LEXCP.

The frame format for a PPP-LEX data packet is shown in the following illustration. The MAC frame is transferred except for the FCS field. The LAN extension interface unit computes the FCS for packets transferred to the LAN and strips the FCS for packets destined for the host router.



PPP-LEX data packet structure

HDLC flag
HDLC frame delimiter.

Address
Address field containing broadcast address 0xFF.

Control
Control field containing unnumbered information 0x03.

Protocol type
Contains the IETF-assigned protocol type value. In this case this field will always contain 0x0041 to indicate a data packet.

F
Set if the LAN FCS field is present. Because PPP-LEX data packets do not contain the LAN FCS field, this bit should not be set (field=0).

I
Set if the LAN ID field is present. Because PPP-LEX data packets do not contain the field, this bit should not be set (field=0).

Z
Set if IEEE 802.3 Pad must be zero filled to minimum size.

0
Reserved, must be zero.

Pad
Any PPP frame may have padding inserted in the Optional Data Link Layer Padding field. The value tells the receiving system how many pad octets to strip off. The LAN extension interface protocol does not support the Optional Data Link Layer Padding field, so the value of this field should be zero.

MAC type
This field contains the most up-to-date value of the MAC type as specified in the most recent *Assigned Numbers RFC*. The current value (according to RFC 1841) indicates IEEE 802.3/Ethernet with canonical addresses.

Destination MAC address
6-octet field containing the MAC address of the destination system as defined by IEEE. The MAC type field defines the bit ordering.

Source MAC address
6-octet field containing the MAC address of the source system as defined by IEEE. The MAC Type field defines the bit ordering.

Length / type
Ethernet protocol type. For IEEE 802.3 frames, this is a length field.

LLC data
Remainder of the MAC frame which is (or would be if it were present) protected by the LAN FCS.

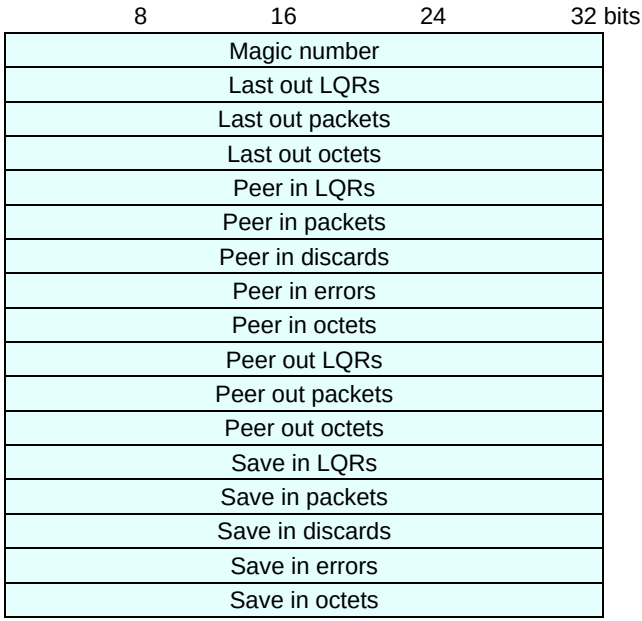
HDLC CRC
16-bit cyclic redundancy check field.

LQR

RFC 1333 http://www.cis.ohio-state.edu/htbin/rfc/rfc1333.html

The Link Quality Report (LQR) protocol specifies the mechanism for link quality monitoring within PPP. Packets are sometimes dropped or corrupted due to line noise, equipment failure, buffer overruns, etc. and it is often desirable to determine when and how often the link drops data. Routers may temporarily allow another route to take precedence, or an implementation may have the option of disconnecting and switching to an alternate link. For these reasons, such a quality monitoring mechanism is necessary.

One LQR packet is encapsulated in the information field of PPP data link layer frames where the protocol field indicates type hex c025. The structure of the LQR packet is shown in the following illustration:



LQR packet structure

Magic number

Aids in detecting links which are in the looped-back condition.

Last out LQRs
Copied from the most recently received Peer Out LQRs on transmission.

Last out packets
Copied from the most recently received Peer Out Packets on transmission.

Last out octets
Copied from the most recently received Peer Out Octets on transmission.

Peer in LQRs
Copied from the most recently received Save In LQRs on transmission. Whenever the Peer In LQRs field is zero, the Last Out fields are indeterminate and the Peer In fields contain the initial values for the peer.

Peer in packets
Copied from the most recently received Save In Packets on transmission.

Peer in discards
Copied from the most recently received Save In Discards on transmission.

Peer in errors
Copied from the most recently received Save In Errors on transmission.

Peer in octets
Copied from the most recently received Save In Octets on transmission.

Peer out LQRs
Copied from the Out LQRs on transmission. This number must include this LQR.

Peer out packets
Copied from the current MIB ifOutUniPackets and ifOutNUniPackets on transmission. This number must include this LQR.

Peer out octets
Copied from the current MIB ifOutOctets on transmission. This number must include this LQR.

The following fields are not actually transmitted over the inbound link. Rather, they are logically appended to the packet by the Rx process.

Save in LQRs
Copied from In LQRs on reception. This number must include this LQR.

Save in packets
Copied from the current MIB ifInUniPackets and ifInNUniPackets on reception. This number must include this LQR.

Save in discards
Copied from the current MIB ifInDiscards on reception. This number must include this LQR.

Save in errors
Copied from the current MIB ifInErrors on reception. This number must include this LQR.

Save in octets
Copied from the current InGoodOctets on reception. This number must include this LQR.

PAP

RFC 1334 http://www.cis.ohio-state.edu/htbin/rfc/rfc1334.html

Password Authentication Protocol (PAP) provides a simple method for the peer to establish its identity using a 2-way handshake. This is done only upon initial link establishment.

The PAP packet is encapsulated in the Information field of a PPP data link layer frame where the protocol field indicates type hex c023. The structure of the PAP packet is shown in the following illustration:

Code	Identifier	Length	Data . . .
1 byte	1 byte	2 bytes	

PAP packet structure

Code

One-octet field which identifies the type of PAP packet. PAP codes are assigned as follows:

1 Authenticate-Request.
2 Authenticate-Ack.
3 Authenticate-Nak.

Identifier

Aids in matching requests and replies.

Length

Indicates the length of the PAP packet including the Code, Identifier, Length and Data fields.

Data

Zero or more octets, the format of which is determined by the Code field. The format of the data field for Authenticate-Request packets is shown below:

Peer-ID length	Peer-ID	Password length	Password
1 byte	variable	1 byte	variable

Data structure for Authenticate-Request packet

Peer-ID length
Length of the Peer-ID field.

Peer-ID
Indicates the name of the peer to be authenticated.

Password length
Length of the Password field.

Password
Indicates the password to be used for authentication.

The format of the data field for Authenticate-Ack and Authenticate-Nak packets is shown below:

Message Length	Message
1 byte	variable

Data structure for Authenticate-Ack and Authenticate-Nak packets

Message length
Length of the Message field.

Message
The contents of the Message field are implementation dependent.

ATCP

RFC 1378 http://www.cis.ohio-state.edu/htbin/rfc/rfc1378.html

The AppleTalk Control Protocol (ATCP) is responsible for configuring, the AppleTalk parameters on both ends of the point-to-point link. ATCP uses the same packet exchange mechanism as the Link Control Protocol (LCP). ATCP packets may not be exchanged until PPP has reached the Network-Layer Protocol phase. ATCP packets received before this phase is reached are discarded.

The format of the ATCP packet is shown in the following illustration:

Code	Identifier	Length	Data
1 byte	1 byte	2 bytes	variable

ATCP packet structure

Code

Decimal value which indicates the type of ATCP packet.

1 Configure-Request.
2 Configure-Ack.
3 Configure-Nak.
4 Configure-Reject.
5 Terminate-Request.
6 Terminate-Ack.
7 Code-Reject.

Identifier

Decimal value which aids in matching requests and replies.

Length

Length of the ATCP packet, including the Code, Identifier, Length and Data fields.

Data

Variable length field which may contain one or more configuration options. The format of the ATCP configuration options is as follows:

Type	Length	Data

ATCP configuration options

Type

One-byte indication of the type of the configuration option.

1 AppleTalk-Address.
2 Routing-Protocol.
3 Suppress-Broadcasts.
4 AT-Compression Protocol.
6 Server-Information.
7 Zone-Information.
8 Default Router-Address.

Length

Length of the configuration option including the Type, Length and Data fields.

Data

Value of the Data field.

BACP

RFC 2125 http://www.cis.ohio-state.edu/htbin/rfc/rfc2125.html

The Bandwidth Allocation Control Protocol (BACP) is the associated control protocol for BAP. BAP can be used to manage the number of links in a multi-link bundle. BAP defines datagrams to coordinate adding and removing individual links in a multi-link bundle, as well as specifying which peer is responsible for which decisions regarding managing bandwidth during a multi-link connection. BACP defines control parameters for the BAP protocol to use.

The format of the BACP packet is shown in the following illustration:

Code	Identifier	Length	Data
1 byte	1 byte	2 bytes	variable

BACP packet structure

Code

Decimal value which indicates the type of BACP packet.

1 Configure-Request.
2 Configure-Ack.
3 Configure-Nak.
4 Configure-Reject.
5 Terminate-Request.
6 Terminate-Ack.
7 Code-Reject.

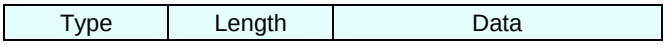
Identifier

Decimal value which aids in matching requests and replies.

Length

Length of the BACP packet, including the Code, Identifier, Length and Data fields.

Data
Variable length field which may contain one or more configuration options. The format of the BACP configuration options is as follows:



BACP configuration options

Type
One-byte indication of the type of the configuration option.
1 Favored-Peer

Length
Length of the configuration option including the Type, Length and Data fields.

Data
Value of the Data field.

BCP

RFC 1638 http://www.cis.ohio-state.edu/htbin/rfc/rfc1638.html

The Bridging Control Protocol (BCP) is responsible for configuring the bridge protocol parameters on both ends of the point-to-point link. BCP uses the same packet exchange mechanism as the Link Control Protocol. BCP packets can not be exchanged until PPP has reached the Network-Layer Protocol phase. BCP packets received before this phase is reached are discarded.

The format of the BCP packet is shown in the following illustration:

Code	Identifier	Length	Data
1 byte	1 byte	2 bytes	variable

BCP packet structure

Code

Decimal value which indicates the type of BCP packet.

1 Configure-Request.
2 Configure-Ack.
3 Configure-Nak.
4 Configure-Reject.
5 Terminate-Request.
6 Terminate-Ack.
7 Code-Reject.

Identifier

Decimal value which aids in matching requests and replies.

Length

Length of the BCP packet, including the Code, Identifier, Length and Data fields.

Data

Variable length field which may contain one or more configuration options. The format of the BCP configuration options is as follows:

Type	Length	Data

BCP configuration options

Type

One-byte indication of the type of the configuration option.

1 Bridge-Identification.
2 Line-Identification.
3 MAC-Support.
4 Tinygram-Compression.
5 LAN-Identification.
6 MAC-Address.
7 Spanning-Tree-Protocol.

Length

Length of the configuration option including the Type, Length and Data fields.

Data

Value of the Data field.