Network Working Group
INTERNET DRAFT

Nishit Vasavada
Amber Networks, Inc.

July 2001

ESP: Encapsulation Services Protocol
<draft-vasavada-pwe3-esp-00.txt>

Status of this Memo

Abstract

This document describes "Encapsulation Services (ES)" Protocol (ESP) - the signaling and real-time transport protocol, suitable for emulation of layer 1 and layer 2 circuits (e.g. FR, TDM, ATM and private leased lines) over IP transport networks.

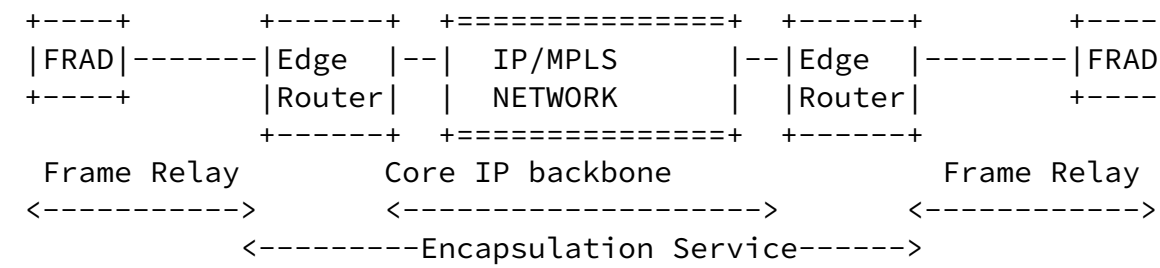
ESP provides end-to-end signaling mechanism suitable to establish ESP tunnels and sessions over the underlying tunneling protocol. We currently use Layer Two Tunneling Protocol (L2TP) as defined in [RFC2661]. Other tunneling protocol may be extended in future to support ESP.

Each emulated layer 1/2 circuit is encapsulated in an ES session inside an ES tunnel. The ES session parameters comply with the parameters of the parent ES tunnel.

1. Introduction

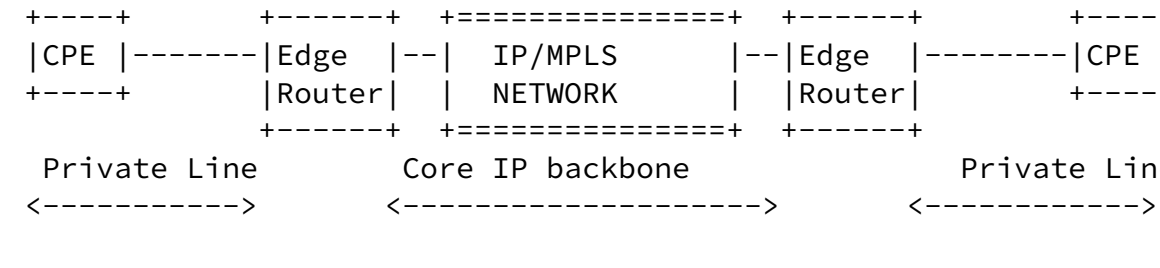
With service providers moving towards all new IP backbone networks,

there is a need to carry access technologies (e.g., Frame Relay, ATM) over the IP backbone network. Most of these legacy access methods are based on Layer 2 technologies. However, IP traditionally being a Layer 3 protocol, the challenge is to carry Layer 2 traffic over this IP backbone.



One such example is shown in the above figure where two Frame Relay Access Devices (FRADs) are connected through an IP backbone. Private Frame Relay networks can also be attached to each other in similar fashion. ESP running on the edge router carries the Frame relay link across the IP/MPLS backbone network to the remote end in a transparent way. ESP emulates the characteristics specific to Frame Relay over the IP network and thus the FRADs at both ends think as if they are connected either directly or through another Frame Relay network. Such emulation method includes the signaling procedure needed to establish the ES tunnels and sessions and then the control protocol needed for the link quality monitoring and also the Frame Relay specific control information propagation mechanisms (such as LMI).

Today's enterprises also depend heavily over private lines, which span across the globe. These private lines carry pure bit-streams, which are never looked into by intermediate network nodes. Hence, carrying these circuits across IP network requires a "Circuit Emulation Service", in particular, "IP Circuit Emulation Service". This is similar to ATM CES which is in use for several years now.



<-------Encapsulation Service----->

In this case, Private Circuits are connected to the Edge Router. ES, running on the edge Router performs "Circuit Emulation" over IP network and carries the Private lines to remote end. In this case, the emulation is more of a "Layer 1" circuit emulation. The control protocol here needs to include mechanism to propagate layer 1 circuit characteristics, such as Idle suppression and Alarm suppression.

Note that in case of Layer-1 private line emulation, the protocol does not need both ends to have similar physical characteristics. On one side, the customer can connect one DS1 of a channelized DS3 link, whereas the customer on the other end can connect through a single DS1 line. Yet information about idle-suppression and alarm status etc. are semantically carried to the remote end.

1.1. Outline of rest of the document:

Section 2 defines the requirements of the protocol. Section 3 shows a reference model using L2TP as the lower layer transport. Rest of

the document is within the context of this reference model. Section 4 provides an overview of the protocol, while Section 5 describes the protocol operation in detail. Section 6 talks about the ES control message formats. Section 7 goes into issues specific to the access link being encapsulated. Appendix A discusses the signaling with L2TP.

2. Requirements of the Protocol

2.1. Connection Identifiers

One important inherent feature of traditional Layer 2 technologies are their connection-oriented nature. But, IP being traditionally connectionless protocol, there is a need to preserve the connection identifiers of the Layer 2 technologies across the IP network. More specifically, for Frame Relay, the (de)multiplexing based on Data Link Connection Identifier (DLCI) must be preserved at the remote end.

2.2. Alarms

Each access technology has some kind of "Alarm" built into the specification of the technology. Such information must be preserved

and carried across the IP network for successful emulation of the same. Frame Relay, for example uses "Link Management Interface (LMI)" to carry such information. ES must have mechanism to propagate such information between two ends.

2.3. Quality of Service

QoS issues are not addressed in this draft.

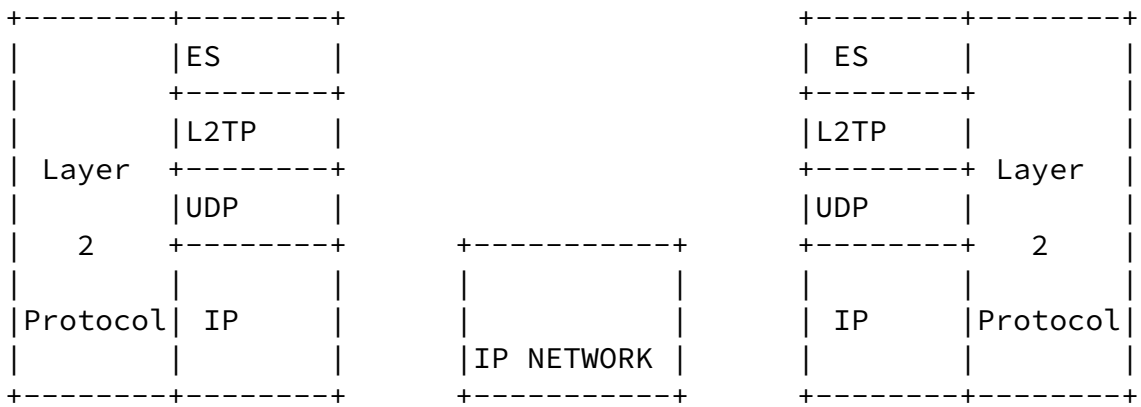
2.4. Other requirements

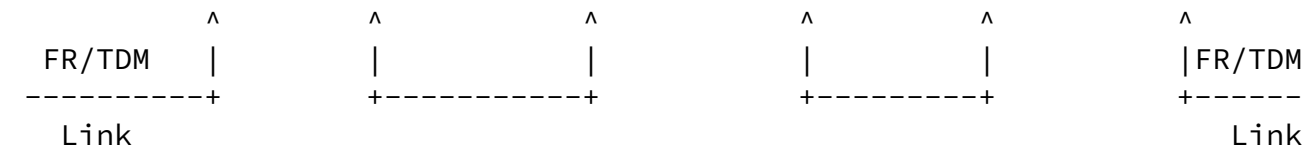
Another requirement for this service is the mechanism for privacy, secrecy, encryption, authentication etc. The mechanism to emulate the private lines must also provide facilities to realize the privacy of user data being transported over the public IP network.

3. Reference Model

An emerging area in the evolution of IP network is the area of "Tunneling Protocols". More specifically, RFC 2661 describes a "Layer 2 Tunneling Protocol" (L2TP) as a means to tunnel PPP traffic over various network clouds. It is proposed to use L2TP as a generic tunneling mechanism and build an "Encapsulation Services" protocol to meet the requirements described in Section 2.

A reference model which can be used for ESP is shown below. However, it is possible to use ES over any other tunneling mechanism in an IP/MPLS network. Future work will focus on these issues.





As shown in the diagram above, the access link is terminated and is fed to ESP. An encapsulation scheme is defined (described later) for this Layer 2 traffic. That, in turn, is encapsulated in L2TP/UDP/IP stack and the resulting IP packet is sent across the IP network to the remote end.

At the remote side, the ES/L2TP/UDP/IP encapsulation is removed and the resulting packet is sent on the access link.

4. Encapsulation Services Protocol Overview

4.1. ES components

Encapsulation Services Protocol (ESP) consists of the following components:
- ES tunnel signaling (only with lower layer - not on the wire)
- ES session signaling
- ES Data transport protocol
- ES Keep Alive Protocol
- ES Alarm propagation protocol
- ES Quality Report Protocol

4.2. Protocol Specification

ESP has two components - ES Tunnel and ES Session.

4.2.1. ES Session

An "ES Session" carries a single layer 1 or 2 connection through IP network.
- In case of Frame Relay, a single DLCI corresponds to an ES session.
- In case of ATM, a single ATM VPI/VCI is carried in an ES session.
- In case of Layer 1 (TDM) circuits, an ES session carries the entire circuit.

The primary objective of an ES session is to "emulate" the characteristics of the circuit through the IP network. An ES session performs the following:

- Performs necessary signaling for connecting the two circuits at the two ends of the network (with the help of underlying transport)
- Exchanges and negotiates the circuit-related traffic parameters between the two end points
- Carries "Service Data Units" (SDUs) for the circuit
- Propagates the "Alarm" and "OAM" information through the IP network to the remote end
- Maintains the connectivity of the circuit through the IP network by appropriate "keep-alive" mechanism
- Monitors, prepares and generates "Quality Report" for the session in IP network

4.2.2. ES Tunnel

An "ES Tunnel" is an aggregation of "ES sessions" which share the following:

- All sessions are between the same pair of IP hosts (tunnel/session end points)
- All sessions are for identical access technology (FR/ATM/TDM)
- All sessions require similar service level treatment in the IP network

The following 4-tuple identifies an ES tunnel:

- IP address of tunnel's remote endpoint: This would typically be a loop-back interface IP address of the remote host, but any other value MAY be provisioned. For the purpose of accepting an ES tunnel, this provisioned value is checked against the IP address of incoming request.

- Access link type (FR/ATM/TDM/any other defined in future): This is an ASCII string - current recognized values are "FR", "ATM" and "CES" (for TDM circuits). Future work may involve standardizing values for various access link types.

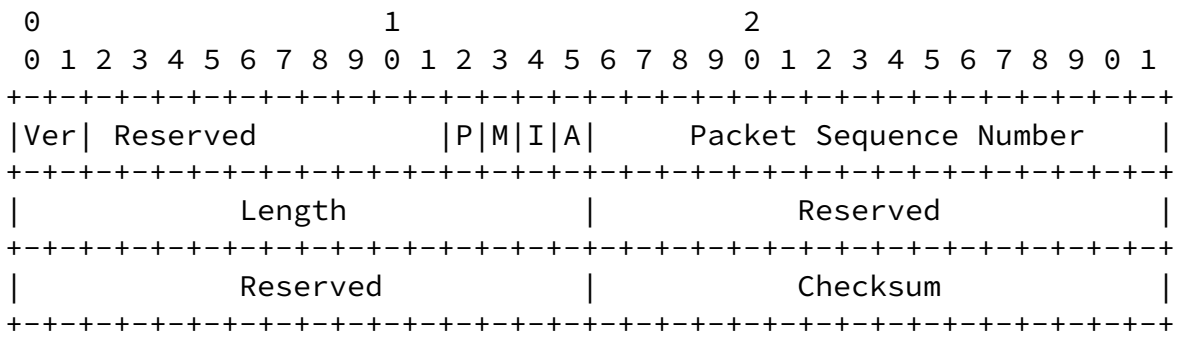
- Service attributes: Service attributes are represented by an ASCII string mutually agreed on by the two ends in an "out-of-band" way or administratively. This parameter maps to a specific set of attributes of the service. This may include, but is not restricted to the level of service, customer information, or any other attributes which may differentiate the traffic in this tunnel from that in any other tunnel. An example of this string is "customer_a_gold".

- DS value: ES conveys this value to the lower layer transport. Currently this applies to L2TP, in the context of [L2TPDS]. The value is used both for the control packets (CCDS AVP as defined in [L2TPDS]) and session packet (SDS AVP is defined in [L2TPDS]).

4.3. ES header format and encapsulation

A generic header format is proposed to facilitate data transport of all kinds of Layer 2 protocols. The header is on the similar lines of the RTP header with extensions/modifications for Encapsulation

Services. There is a 16-bit IP like checksum for the ES header to ensure correctness of sequence number and other header fields.



Flags:

- Ver: Version number, set to 1
- Reserved: Set to 0 on transmit, ignored on receipt
- P - for PCRC: Action on the payload CRC error. Drop the packet if PCRC bit is 1, do not drop if the bit is 0. Latter case is used for CES.
- H = Idle bit mask present (currently set to 0)
- I = Idle on/off (currently set to 0). If set to one, it indicates that an AIS was received from local subscriber link, and the far side should play idle pattern on the DS3 line.
- A = Alarm state on/off (currently set to 0)
- Packet sequence number: Incremented for every ES packet for a given connection. Using sequence number is optional. If Sequence numbers are not used, then this is set to 0. However, they are needed to reorder packets received out of order. The sequence number starts from and wraps to 1, not 0 (since 0 has special meaning).
- Length: Length of the payload
- Checksum: Used for the header and is 16 bit IP style checksum. If the checksum is wrong, the packet will be dropped.
- CRC (32 bits) for the entire header and payload (to take care of FR/ATM encapsulation). CRC is not used for ES control PDUs. ES control PDUs compute a checksum over the ES control portion of the PDU as described in individual control message descriptions.

5. ES Protocol Operation

Like most other protocols, ESP is split in two parts - Control Plane and Data Plane.

5.1. ESP Control Plane

Setting up a tunnel (if needed), sessions within and the maintenance messages are within the Control Plane. There is no ES level message for setting up tunnel (see 5.1 for more on tunnel set-up) - lower layer transport signaling is used for this purpose. All the ES session messages have the following format:

12 byte ES Header + ES Control Message

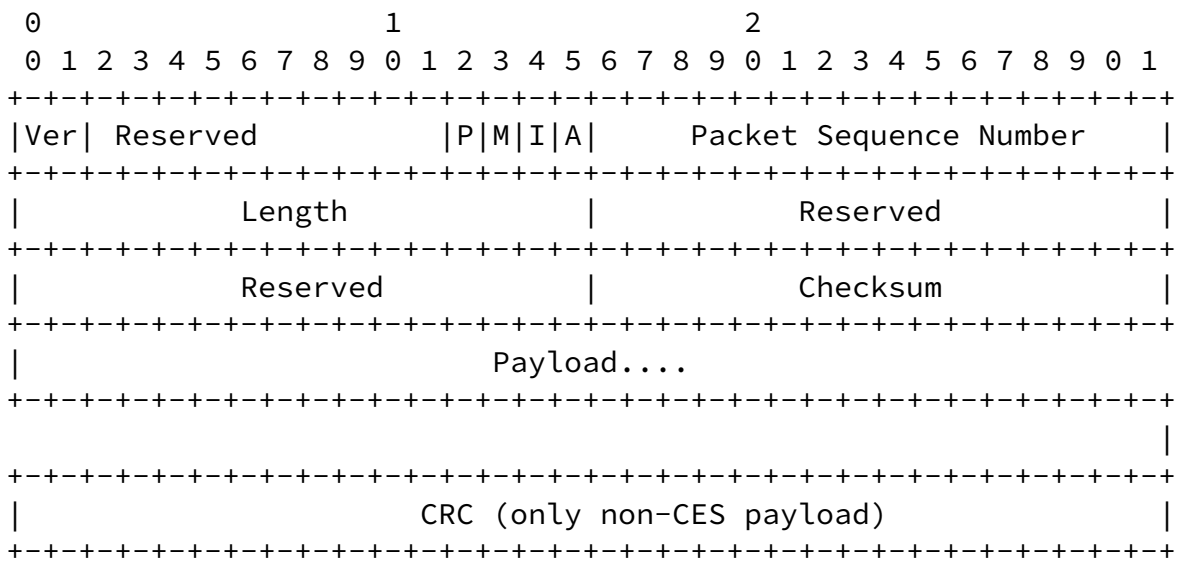
ES header is defined in section 4.3. ES control messages are defined in section 6.4. A 16 bit IP style checksum ensures integrity of

control plane PDUs.

5.2. ESP Data Plane

Data plane carries the L1/L2 payload to the far end of the tunnel. The ESP data plane PDUs have the following format:

12 byte ES Header + Encapsulated Payload [+ CRC]



The payload is encapsulated and a CRC is added for FR and ATM cases.

The access link type specific discussion on payload encapsulation is in Section 7.

The CRC is a 32 bit field at the end of the ES data PDU, and is computed over the entire ES header and payload.

Rest of the document will mainly focus on the control plane.

5.3. ES Tunnel Signaling

5.3.1. ES Tunnel setup

When a network operator intends to carry certain Layer 1 or 2 traffic to a certain remote IP host, the operator sets up an ES tunnel first. An ES tunnel is a logical tunnel. It utilizes the tunnel set up by the underlying transport protocol, e.g. an L2TP tunnel. No ES control packets are exchanged for setting up an ES tunnel.

ES requests the underlying transport layer to setup a tunnel. The distribution of provisioning and tunnel acceptance work is implementation dependent. More information on this can be found in Appendix A.

The Service Access Point between ES and L2TP is left to the implementation. A good value to use would be the local tunnel ID assigned by L2TP.

On receiving an indication from ES to set up a tunnel, the L2TP on local side initiates the tunnel set-up. This follows the normal

set-up procedure as outlined in [RFC2661] with following constraints:

- exchange ESP as a service type in the service capabilities AVP, as defined in [L2TPST]. The M-bit is set for this AVP to ensure the peer supports ESP.
- use the string consisting of access_type.service_attributes as the hostname for the host name AVP
- DS value for the CCDS AVP as defined in [L2TPDS].

The remote side L2TP either processes the request at L2TP layer, or passes on this request to ES providing the same <4-tuple> about the originating side ES. This depends on the implementation as outlined in Appendix A.

5.3.2. ES Tunnel Tear-down

This includes the following cases -

- Based on configured policies, ES determines that an incoming tunnel should not be accepted
- User intends to tear down an existing tunnel.

Either way, ES notifies the underlying L2TP to release the tunnel. While doing so, it specifies the SAP agreed up on when setting up the tunnel (please see above). An appropriate L2TP error code SHOULD be passed to L2TP, which L2TP can use while tearing down the tunnel.

5.4. ES Session Signaling

When user wants to connect a specific layer-1 or layer-2 circuit (referred to as "access circuits" in the rest of the document) with a remote circuit (layer-1 or layer-2 respectively), ES uses the session signaling mechanisms for this purpose. An ES session is part of an ES tunnel. For ES session to be established, a corresponding tunnel MUST already exist.

The session terminates on the specific access circuit on the remote side. User on one side requests ES to setup an ES session between the two access circuits. ES specifies the local and remote side "access circuit identifiers" to L2TP while requesting to set up a session in the tunnel. These circuit identifiers depend on the specific Layer-1 or Layer-2 protocol that is being carried as listed below:

- Access port/line number and DLCI value for Frame Relay
- Access port/line number and VPI/VCI values for ATM
- Access port/line number for TDM circuits coming into the box

On the remote end, L2TP supplies the same set of identifiers to ES. ES decides whether to accept or reject such an incoming call, depending upon actual provisioning of the access circuit or some other local policy.

If it wants to accept the call, it indicates so to L2TP. Otherwise, it indicates to L2TP to reject the call and indicates the "Cause and Diagnostic" code for the same.

5.4.1. ES Session setup

The ESP session is transported by L2TP session PDUs. To set up an ESP session, an L2TP session needs to be established first. L2TP uses following attributes passed by ES while setting up the session.

5.4.1.1. Signaling parameter: End-Identifiers

The end identifiers are encoded in the End-Identifier AVP of L2TP session set-up message. This field is a sequence of ASCII octets. ES encodes this in a format (defined later) and supplies to L2TP. The ES on the remote side decodes this to extract the appropriate access circuit identifiers for the session.

The way the remote circuit ID is encoded depends on the access type. For Frame Relay, both interface and DLCI information is needed. For TDM circuits, only interface number is communicated.

5.4.1.2. Signaling parameter: Service Type

L2TP uses ES as the service type in the Service Type AVP while setting up the session. If there is any problem in using ES as the service type for the session, the session is torn down.

5.4.1.3 Signaling parameter: DS Value

DS value is same as the one used in tunnel set-up. L2TP uses it for SDS AVP while setting up the session.

Once L2TP session is set up, ESP sets up a session.

5.4.1.2. Traffic parameter negotiation

After L2TP session gets established, ES on both sides exchanges the access side traffic parameters with the other side. These parameters are specific to the particular access link, carried in the session:

For Frame Relay circuits:
- Committed Information Rate (CIR)
- Committed Burst (Bc)
- Excess Burst (Be)

For TDM circuits:
- Packet Size
- Jitter
- Minimum queue depth

For ATM VCs: [TBD]

With the help of these signaling procedures, ES on both sides performs necessary checks, such that ingress parameters on one side matches the egress parameters on the remote side and vice-versa.

In case of a failure in such checks, ES drops the sessions with an appropriate cause and diagnostics code.

5.4.3. Session Tear-down

When user no longer needs the session, ES exchanges a "Terminate PDU" with the remote peer and then brings down the underlying ES session.

5.4.4. Cause and diagnostic codes

Currently ES follows the well-established codes for L2TP. Future work may involve setting up special codes for ES use.

5.5. Alarm Propagation

ES provides a mechanism to propagate access side "alarm information" to remote side. This ensures that the user at the CPE sees a virtual end-to-end layer-1/layer-2 link extending from its premise to the remote premise.

The exact semantics of the alarm is access-technology dependent. The specific alarms for Frame Relay, TDM circuits and ATM VCs are described below:

5.5.1. Frame Relay

Local Management Interface (LMI) is used to perform link management on LMI links. LMI has mechanisms to convey status of individual DLCI (known as "A-bit status") on a frame relay link. ES terminates the LMI protocol in the box and translates the information into ES control PDU. The far side re-translates the ES control PDU back into LMI message. Thus, both the ES end-points need to participate in LMI protocol. Thus, the A-bit information gets propagated end-to-end through the IP network in a transparent manner.

A particular implementation can choose the frequency of such information exchange. It can be tied to the receipt of A-bit status messages on one end on the link or can be performed in certain periodic interval. Configuration or signaling of this periodic interval will be covered in a future version.

Alarms can also be propagated in bulk so that multiple individual alarms do not have to be transported across the IP network.

5.5.2. TDM circuits

The alarm information for a TDM circuit depends on the type of line being emulated, e.g.: DS1 or DS3. Whenever a line (DS1 or DS3) goes into alarm, ES sends a message to the remote ES specifying the session corresponding to the line. The remote side ES plays the appropriate alarm-bit-pattern for the appropriate interface.

5.5.3. ATM VCC

[TBD]

5.6. Idle Suppression Protocol

When there is no traffic on a Layer-1 circuit, the line can be carrying idle-pattern. In such a case, ES notifies the remote side about the same. The remote device starts playing idle pattern on the line.

5.7. Quality Report

ES provides a mechanism to monitor and provide feedback in terms of "Link Quality Report" on a per-session basis. Such information is collected/measured on the egress side of the session and is fed back to the ingress side. Ingress side ES can take intelligent decisions based on such information for building resiliency and robustness to the service. It is optional on part of the egress equipment to perform such measurement. It is also optional on part of the ingress equipment to take any intelligent decision based on such information. However, if egress equipment does measure and propagate the information to the ingress side, then the ingress side equipment MUST make it accessible to user.

The parameters included in such monitoring and measurements are:

- Packet Loss: It may not be possible to measure packet loss without using the sequence numbers for each packet.
- Round Trip Delay (RTD): ES provides a special loop-back IP packet which when implemented must be looped-back to the source with a timestamp. This can be used to measure the round trip delay for a packet on a session.

- Bandwidth: The instantaneous bandwidth observed on the egress side can be measured for a Frame relay circuit.
- Jitter: The instantaneous jitter experienced on the egress side for a TDM circuit emulation, can be fed back to the ingress side.

More work in this area will be done in near future.

5.8. Keep Alive Protocol

ES relies on L2TP's "Hello Protocol" to maintain the heart beat of the tunnel.

6. ES Message formats

There are two classes of ES messages:

- One class contains three of the ES parameters that are signaled using L2TP mechanisms: Service Capabilities, Service Type, DS and End-Identifier
- The second class contains ES messages that are exchanged between two ES peers as ES control packets (L2TP payload packets).

These messages are described below:

6.1. Service Capabilities and Service Type AVP (Signaled through L2TP)

The exchange of this information is beyond the scope of this

document. Please refer to [L2TPES].

6.2. DS value (Signaled through L2TP)

The exchange of this information is beyond the scope of this document. Please refer to [L2TPDS].

6.3. Remote end identifier (Signaled through L2TP AVP End-Identifier)

[L2TPES] defines the format of the End-Identifier AVP, which is used to identify the local and remote access connection identifiers at either end of the L2TP tunnel.

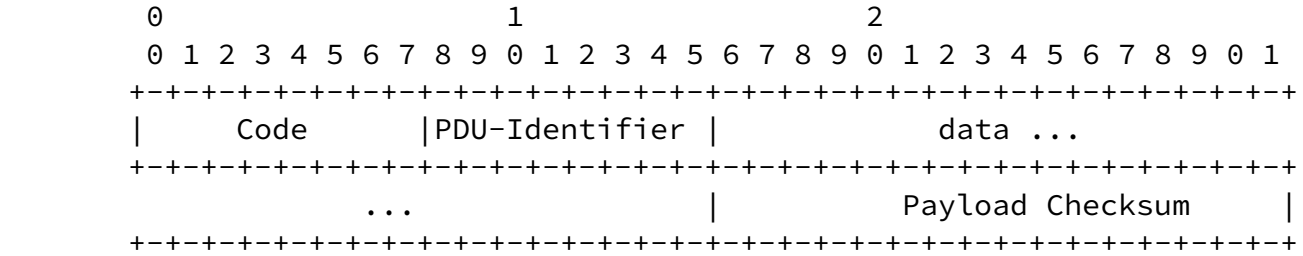
6.4. ES Control messages

The ES control messages are modeled in the same way as the PPP LCP control packets. There are four classes of control messages:

- Session Configuration messages: These are used to exchange the access side connection parameters.
- Session Terminate messages: These are exchanges to gracefully tear down the ES session (before actually tearing down the corresponding L2TP session).
- Access Alarm propagation messages: These messages are used to propagate access side alarm status information. Alarms from multiple sessions can be clubbed together and the message can be sent on a tunnel basis.
- Link Quality Report messages: These are exchanged to report link quality reports.

Every ES message is "request-response" based. For every message, the remote ES peer sends back an acknowledgement to the sending ES peer. For sending multiple messages on the same session, ES uses a sliding window protocol with window size "1". The sender has to make the transmission of every packet reliable by starting a timer and awaiting an acknowledgement. Each message contains a "PDU-Identifier" which must be sent back in the response message by the remote peer. The sender must match the "PDU-Identifier" in the response message to that in the last sent message to validate a response message.

A summary of ES control message is shown below:



Code: The code field is one octet. It identifies the type of message that is encoded. When an unknown code field is received, a "Code Reject" message is generated.

The possible values of the "Code" field are:

0x0001: Configure Request
0x0002: Configure Acknowledgement
0x0003: Configure NAK (Negative Acknowledgement)
0x0004: Configure Reject
0x0005: Reserved

0x0006: Reserved
0x0007: Reserved
0x0008: Reserved
0x0009: Quality Report Message
0x000a: Terminate Request
0x000b: Terminate Acknowledgement
0x000c: Bulk Alarm Message
0x000d: Bulk Alarm Acknowledgement

PDU-Identifier: It is a one octet field. It is used to identify a response message with a request message sent out earlier. When a message with invalid identifier is received, it is silently discarded.

The PDU-Identifier must be changed whenever a new Configure Request is generated and sent to the remote side and whenever a valid reply has been received from the remote peer. When a Configure-Request is retransmitted because of timeout, the PDU-Identifier value MAY be changed. When responding to a request, the PDU-Identifier MUT match the value contained in the request being responded to.

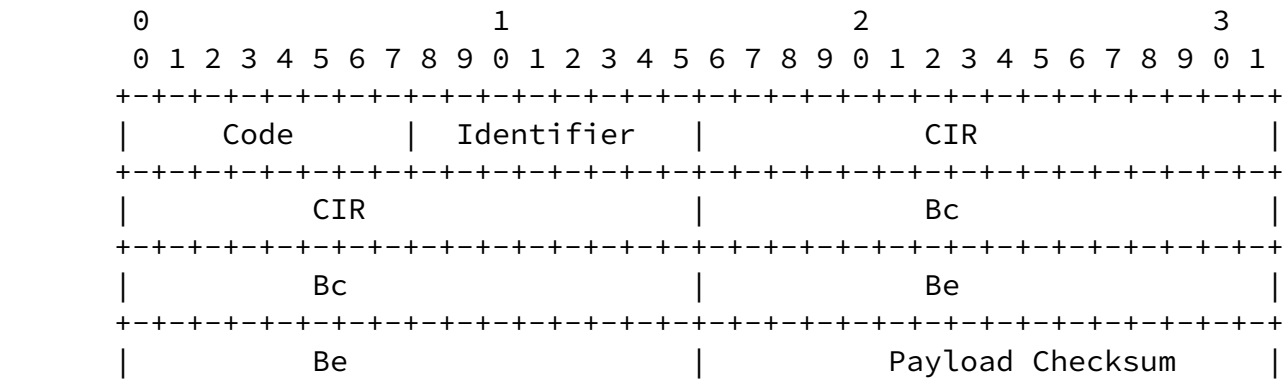
Data: The data field may contain fixed or variable number of octets, depending upon the type of message encoded (indicated by "code" field) as well as the access type (FR, TDM...). For the messages that can contain variable number of octets, the length is obtained from the header of the ES PDU (described in Section: 4.3).

Payload checksum: This is a 16 byte IP style checksum on the payload beginning the "Code" field.

6.4.1. Configure Request (ConfReq)

The ES Configure Request message contains the access side connection parameters. For Frame Relay, the parameters encoded are:
- CIR in "Kilo-Bits-per-second" (4 octets)
- Bc in "number of Bytes" (4 octets)
- Be in "number of Bytes" (4 octets)

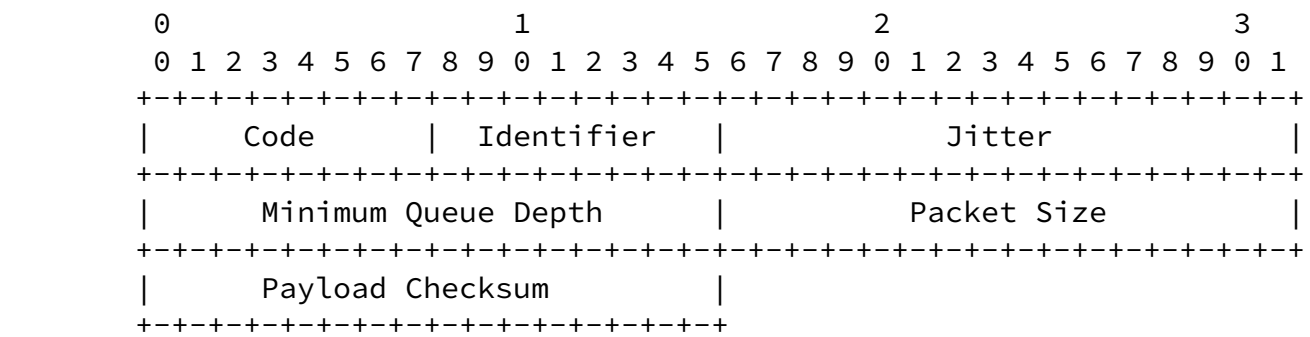
The Configure Request message format for Frame Relay is shown below:



Code: 1 for ES Configure-Request
Identifier and FR related fields: As described above

For CES sessions, the parameters encoded are:
- Jitter in "10s of micro-seconds" (2 octets)
- Minimum Q-depth in "10s of micro-seconds" (2 octets)
- Packet Size

The Configure Request message format for CES sessions is shown below:



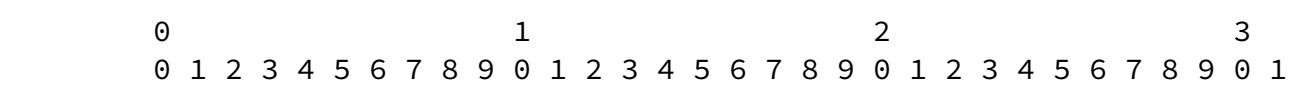
Code: 1 for ES Configure-Request

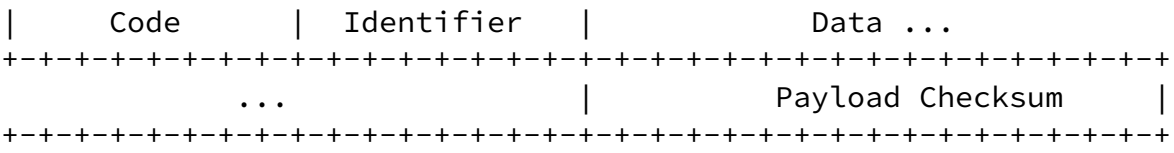
Identifier and CES related fields: As described above

Upon receipt, the ES peer verifies that the requested parameters are supported, and the proposed values are acceptable. If any parameter is not acceptable, the parameter is included in the Configure Reject message and returned to the sender of Configure Request message. If a parameter is supported, but if the value is not acceptable, a Configure NAK is sent to the sender of the Configure Request message. The Configure NAK includes the parameters whose values proposed in Configure Request is/are not acceptable, and offers an alternate value for these parameters.

6.4.2. Configure Acknowledgement (ConfAck)

The receiver of a ConfReq message MUST send back a ConfAck to the sender, if the parameters present in the message were agreeable with. The format of the message is shown below:





Code: 2 for Configure Acknowledgement

Identifier: As described above, this field is copied from the PDU Identifier field in the Configure Request which this message is acknowledging.

Data: This field is a byte by byte copy of the data portion of the

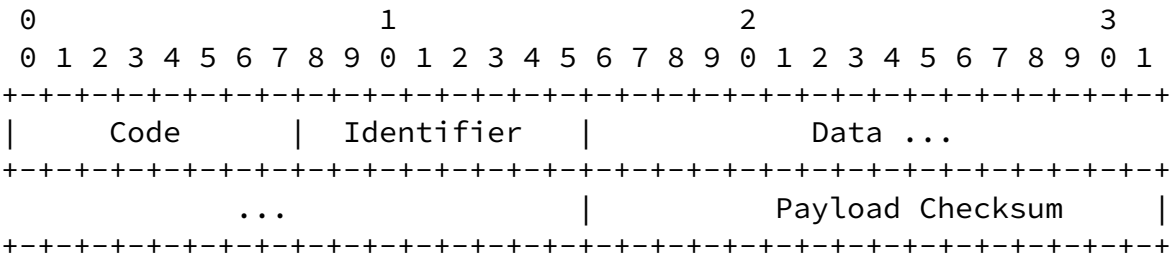
Configure Request which this message is acknowledging.

Upon receipt, the recipient of ConfAck MUST find the ConfReq it sent based on the PDU Identifier in the received ConfReq. If no such ConfReq was sent, the ConfAck is silently discarded. If the ConfReq was found, the data portion of the sent ConfReq and received ConfAck are compared. If they are identical, ES session goes in up state. However, if they are not identical, another ConfReq is sent to the peer.

Data: Byte to byte copy of Configure-Request

6.4.3. Configure NAK (ConfNAK)

If the values present in the Configure Request message are not agreeable with, a "Configure NAK (Negative Acknowledgement)" message must be sent back to sender of ConfReq. The format of this message is shown below:



Code: 3 for ConfNAK

Identifier: As described above, this field is copied from the PDU Identifier field in the Configure Request which this message is

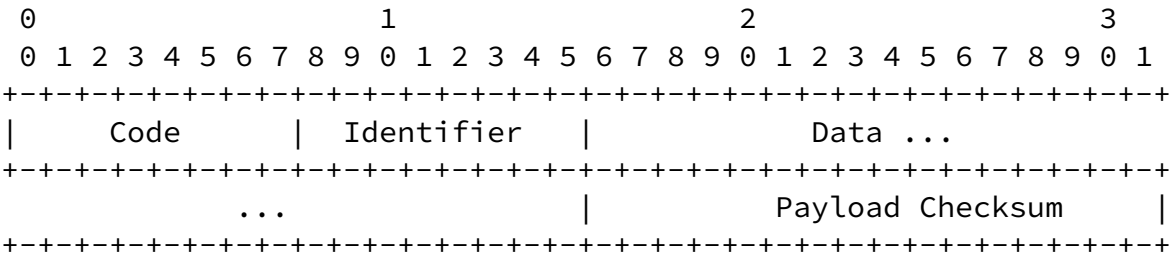
NAKing.

Data: This field contains the parameters whose values in ConfReq were not agreeable to the recipient. The sender of ConfNAK proposes new values for these parameters and sends the message back to the sender of ConfReq.

Upon receipt, the recipient of ConfNAK MUST check the proposed values against its own configuration. If the values proposed in ConfNAK are acceptable, a new Configure Request message SHOULD be sent with all the parameters (including the ones which were not NAKed by the other side). The values of the parameters NAKed will be the values proposed in ConfNAK, or any other value per the configuration.

6.4.4. Configure Reject

If any/all parameter(s) present in the Configure Request message are not acceptable to the receiving side based on the configuration, a "Configure Reject" message back to sender of ConfReq. The format of this message is shown below:



Code: 4 for ConfRej

Identifier: As described above, this field is copied from the PDU Identifier field in the Configure Request which this message is NAKing.

Data: This field contains the parameters in ConfReq which are not supported by the receiving side.

Upon receipt, the recipient of ConfRej MUST check the parameters not supported by the other side. If the local configuration permits

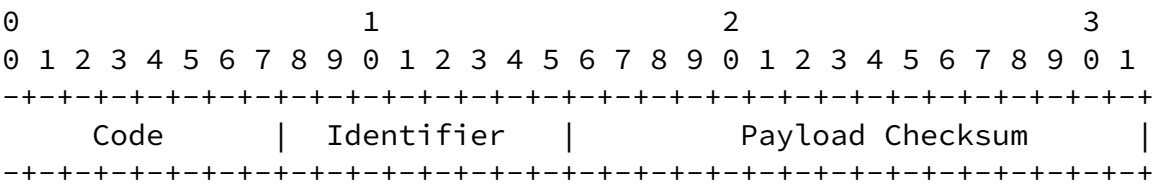
carrying out the session without those parameters, a new Configure Request MUST be sent without the parameters mentioned in Configure Reject. If the local configuration does not permit setting up a session without these parameters, the local side MUST send a Terminate Request to tear down the session.

6.4.5. ES Quality Report

The need of the Quality Report has been described in section 5.7. This area has been marked for future work.

6.4.6. ES Terminate Request

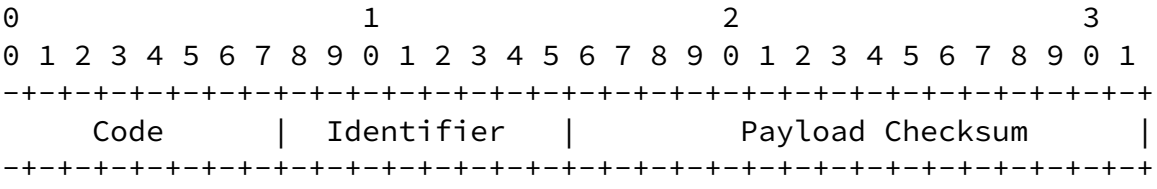
ESP uses this message to terminate an existing session. No additional information is exchanged in this message. The format of this message is shown below:



Code: 10 for TermReq

6.4.7. ES Terminate Ack

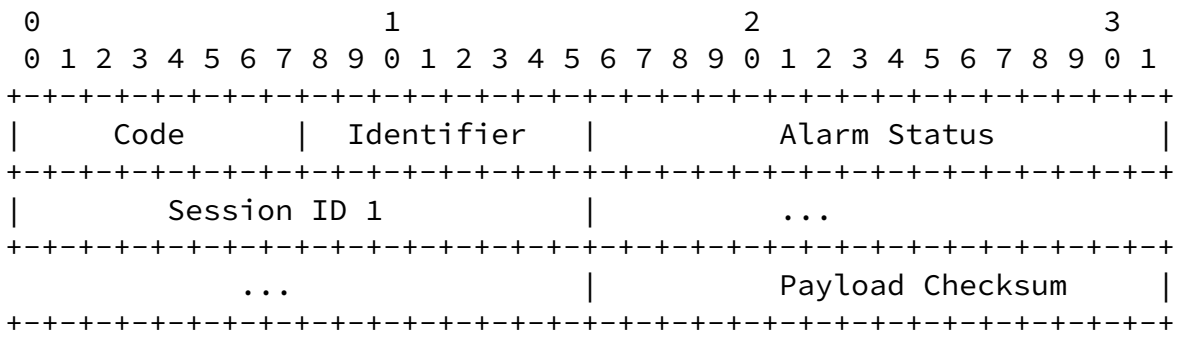
ESP uses this message to acknowledge a terminate request received from the peer. No additional information is exchanged in this message. After sending this message, the sending side SHOULD clean up the ESP session and inform the lower layer about the session termination. Similarly, the receiving side SHOULD clean up the ESP session and inform the lower layer about the session termination. The format of this message is shown below:



Code: 11 for TermAck

6.4.8. ES Bulk Alarm Message

This message is used to aggregate alarms on multiple ES sessions. The format of this message is shown below:



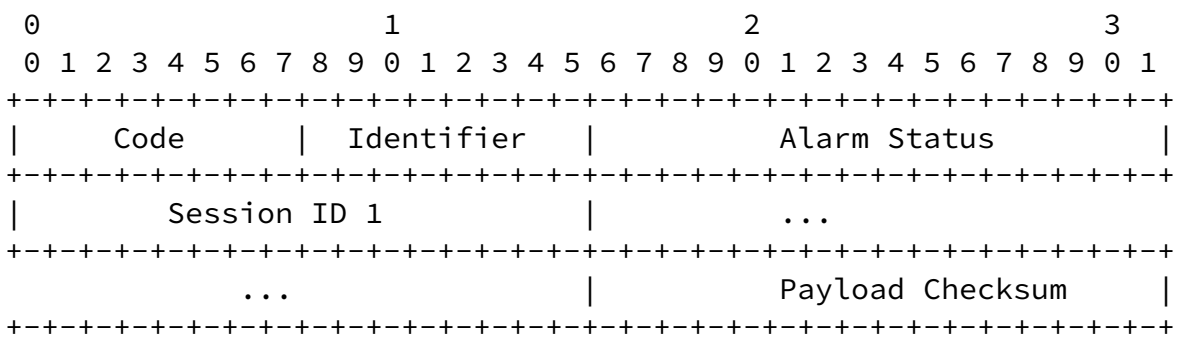
Code: 12 for Bulk Alarm Message

Alarm Status: Two bytes - reflects the status of the alarm for the sessions mentioned in this message

Session ID (one or more): The alarm status shows in the "Alarm Status" field reflects the status of sessions whose IDs are mentioned here.

6.4.9. ES Bulk Alarm Ack

This message is used to acknowledge Bulk Alarm Message. The format of this message is shown below:



Code: 13 for Bulk Alarm Response

Alarm Status: Two bytes - carries the same value as the one in Alarm Status field of the Bulk Alarm Message being acknowledged

Session ID (one or more): These fields are copied from the Bulk Alarm Message being acknowledged

7. Layer 2 related issues

7.1. Layer 2 type independent issues

7.1.1. Out of order packets

Due to the inherent connection-less nature of IP networks, packets may arrive out of order at the end of the ES tunnel. If packet buffering and reordering is desired, sequence numbers SHOULD be utilized in the ES header.

7.1.2. Packet integrity

A 16 bit checksum over the ES header, and a 32 bit CRC over the entire ES header and payload ensures packet integrity and error detection.

7.1.3. Connection Multiplexing

This is provided by carrying multiple ES sessions on top of L2TP sessions in a single L2TP tunnel.

7.2. FR specific issues

7.2.1. CIR, Be and Bc

Although not needed for FR PVC, ES allows these parameters to be signaled to the far side.

7.2.2. FECN

Both on the ingress and egress of the IP network, if there is any congestion in the direction of flow of traffic, the FECN bit MAY be set. If the FECN bit is already set, it SHOULD not be changed.

7.2.3. BECN

BECN bit is not changed in the current version. Future work may involve setting it if necessary.

7.2.4. D/E

The D/E bit MAY be set at the ingress (when FR is encapsulated into ES) of the network if the traffic does not conform to the negotiated CIR/Be/Bc. If the D/E bit is already set, it SHOULD not be changed at the ingress of the IP network. The D/E bit SHOULD not changed at the egress of the IP network.

7.2.5. Encapsulation

The entire FR packet including the header is encapsulated. CRC16 is not included.

7.3. TDM circuits specific issues

TDM frames are encapsulated in entirety.

8. Security Considerations

All the underlying L2TP Security considerations remain, though no 'new' ones are introduced?

9. IANA Considerations

To be completed.

10. Intellectual Property Considerations

Amber Networks may seek patent or other intellectual property protection for some of all of the technologies disclosed in this document. If any standards arising from this document are or become protected by one or more patents assigned to Amber Networks, Amber intends to disclose those patents and license them on reasonable and non-discriminatory terms.

11. Acknowledgments

Many thanks to Himansu Sahu, Stanley Fong, Harisankar Mallath, Ravi Bhat, Imtiyaj Kaji and Chi Fai Ho for their help in reviewing this draft.

12. References

[RFC2661] Townsley, et. al., "Layer Two Tunneling Protocol "L2TP"", RFC 2661, February 1999.

[RFC1661] Simpson, W., "The Point-to-Point Protocol (PPP)", STD 51, RFC 1661, July 1994.

[RFC2119] Bradner S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[L2TPST]. McPherson D., Nanji S., "L2TP Service Type", Work in Progress, April 2001.

[L2TPDS] Calhoun P., et. al., "L2TP Differentiated Services Extension", draft-ietf-l2tpext-ds-03.txt, Work in Progress, March 2001.

[L2TPES] Vasavada N., "Encapsulation Services Protocol Service Type for L2TP", draft-vasavada-l2tpext-es-svctype-00.txt, Work in Progress, June 2001.

13. Author's Address

Nishit Vasavada
Amber Networks, Inc.
48664 Milmont Drive
Fremont, CA 94538
Phone: +1 510.687.5200
Email: nishit@ambernetworks.com

Appendix A: ES Tunnel Signaling

The document only defines what identifies an ES tunnel. The requirements are for a host, but the document does not restrict the implementation as to which layer (ESP or the transport layer under it - L2TP for now) the provisioning and signaling occurs, specially since ES tunnel is a logical tunnel and no message exchange takes place at ES level to set up a tunnel.

An implementation is required to do the following:

- ESP MUST provide the host name to L2TP. L2TP MUST use it for the host name AVP and other implementation specific details related to host name. L2TP MUST also use it to find other provisioned tunnel parameters if they are provisioned at L2TP layer (please see

below).

An implementation is free to choose the following:

- Whether the tunnel is provisioned at the lower layer, or ES signals the tunnel parameters to the lower layer. These parameters include remote end IP address, host name and DS value to be used for tunnel and sessions within the tunnel. If the host name is provisioned at L2TP layer, it must conform to the ES needs of "access_link_type.service_attributes" format - e.g. "FR.customer_a_gold".
- Whether the tunnel is accepted at L2TP layer, or ES layer. If it is accepted at ES layer, L2TP MUST confirm with ES whether the incoming tunnel request can be accepted. If the tunnel acceptance processing is done at ESP layer, and if ESP decides to not accept the request, ESP MUST pass to L2TP an appropriate error/cause code which L2TP can use to send out a StopCCN.
- The source IP address in the received packet used MUST match the remote endpoint address as provisioned.