



HF-PEP: STANAG 5066 TCP Performance Enhancing Proxy Protocol (S5066-APP9)

30th June 2020

Version: 1.0

Status: Experimental

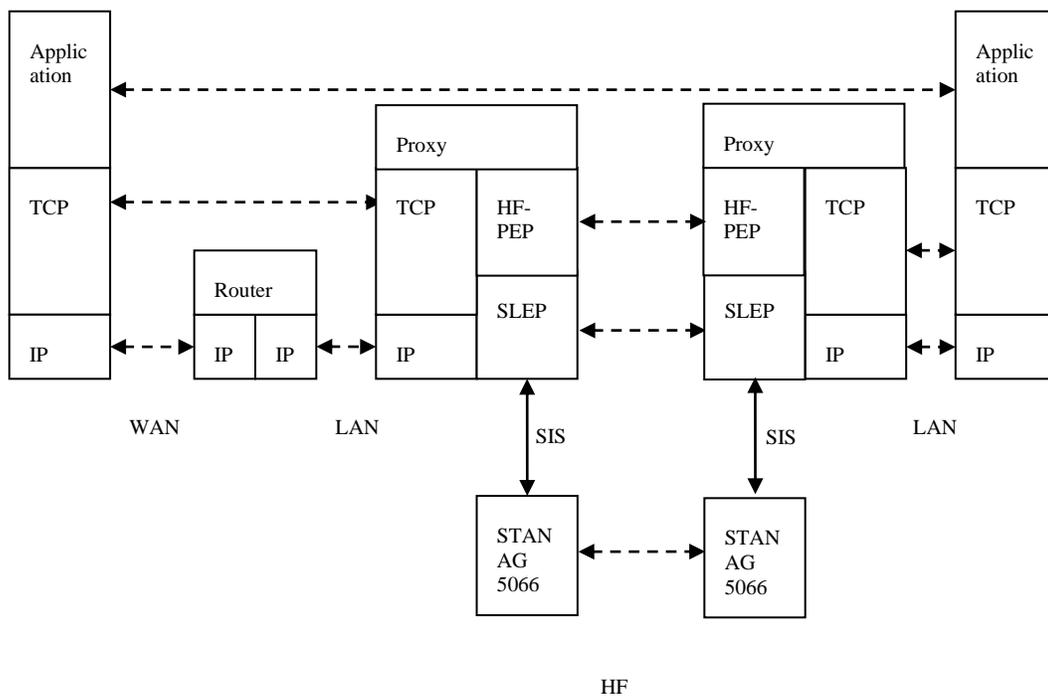
1 Purpose

This document specifies a protocol operating over STANAG 5066 to support TCP PEP (Performance Enhancing Proxy) based operation, to enable efficient operation of TCP based protocols over HF.

This document is published in the STANAG 5066 Application (S5066-APP) series.

2 Operational Model

End to end TCP using IP over an HF Subnet (following STANAG 5066 Annex F.12) often leads to poor or very poor performance. HF-PEP addresses this by using a PEP (Performance Enhancing Proxy) architecture outlined in RFC 3135 “Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations”. The overall architecture is shown in the following diagram.



In this TCP Proxy architecture, an application is communicating over TCP, running over IP in the normal manner. The TCP connection from each application is peered with a proxy, rather than the other application. The proxies communicate using the HF-PEP protocol specified here.

HF-PEP operates over SLEP (SIS Layer Extension Protocol), specified in S5066-APP3. SLEP provides the Stream Services used by HF-PEP. SLEP communicates over STANAG 5066, using the local STANAG 5066 SIS (Subnet Interface Service) to connect. STANAG 5066 peers communicate over an HF network, as shown.

The Proxy can multiplex TCP connections over a single HF link, so that a single STANAG 5066 SAP can be shared by multiple TCP connections and multiple applications running over TCP.

The proxy may control TCP applications based on port including:

- Blocking certain ports, perhaps based on HF conditions and other load being handled by the proxy. For example, to only allow one VOIP connection at any time.
- Giving priority at the proxy level to selected TCP applications, so that data from some streams is submitted ahead of other streams.
- Associating STANAG 5066 priority with each stream, so that the STANAG 5066 layer will give priority to specified streams.

A simple proxy may connect to just one peer and route all traffic to that peer. A proxy may connect to multiple peers and choose the peer proxy based on destination IP address of the destination specified in an inbound TCP connection.

3 HF-PEP Service Definition

This section defines the service provided by the HF-PEP layer, which is based on the SLEP Streaming Service.

3.1 Bind and Unbind Services

3.1.1 HFPEP_BIND_REQUEST

Proxy -> HF-PEP.

Arguments:

1. SAP. Value of SAP (0-16).
2. Extended Address. Optional. 0-256. 0 is equivalent to no extended address.
3. RANK. As for S5066 S_BIND_REQUEST

3.1.2 HFPEP_BIND_ACCEPTED

HF-PEP -> Proxy.

No arguments.

3.1.3 HFPEP_BIND_REJECTED

HF-PEP -> Proxy.

Arguments:

1. Reason. Values as for S5066 S_BIND_REJECTED, plus:
 - a. Extended Addressing Not Supported

3.1.4 HFPEP_UNBIND_REQUEST

Proxy -> HF-PEP

No arguments.

3.1.5 HFPEP_UNBIND_INDICATION

HF-PEP -> Proxy.

Arguments:

1. Reason. Values as for S5066 S_UNBIND_INDICATION

3.2 Stream Services

The HF-PEP service provides a stream service based on the service provided by SLEP. A proxy initiating a connection using HF-PEP needs to communicate information to the peer proxy needed to initiate the second TCP connection.

3.2.1 HFPEP_STREAM_INIT_REQUEST

Proxy -> HF-PEP. Starts a new stream.

Arguments:

1. ID. Application provided ID to identify stream.
2. Address. STANAG 5066 Unicast Address of peer.
3. Priority. STANAG 5066 Priority.
4. Immediate Connect. Controls if connect confirm is immediate (local). If this is set, stream may get rejected after initial confirm and data.
5. Compression. Optional. If set, stream is compressed.
6. Source TCP Port.
7. Destination IP Port.
8. Source IP Address.
9. Destination IP Address.

3.2.2 HFPEP_STREAM_INIT_CONFIRM

HF-PEP -> Proxy.

Arguments:

1. ID.
2. Maximum Data Block Size. Integer, up to 4096.

3.2.3 HFPEP_STREAM_INIT_REJECT

HF-PEP -> Proxy.

Arguments:

1. ID.
2. Reasons:
 - a. Invalid Parameter.
 - b. Too many streams.

3.2.4 HFPEP_STREAM_INIT_INDICATION

HF-PEP -> Proxy. Incoming stream being offered to responder.

Arguments:

1. ID. SLEP generated ID to identify stream.
2. Address. STANAG 5066 Address of Initiator.
3. Priority. STANAG 5066 Priority.
4. Maximum Data Block Size. Integer, up to 4096.
5. Source TCP Port.
6. Destination IP Port.
7. Source IP Address.
8. Destination IP Address.

3.2.5 HFPEP_STREAM_INIT_RESPONSE

Proxy -> HF-PEP. Accept or reject incoming stream.

Arguments:

1. ID.
2. Accept or Reject.

3.2.6 HFPEP_STREAM_DATA_REQUEST

Proxy -> HF-PEP. Initiator or responder sends a block of data.

Arguments:

1. ID.
2. Data.

3.2.7 HFPEP_STREAM_DATA_CONFIRM

HF-PEP -> Proxy

Argument:

1. ID

This confirms data is accepted and allows client to send another data block. This reflects the SLEP mechanism to flow control the application.

3.2.8 HFPEP_STREAM_DATA_INDICATION

HF-PEP -> Proxy.

Arguments:

1. ID.
2. Data.

Proxy must accept the data.

This may be a zero-length block of data, which indicates that the stream is active. This may be helpful for controlling application timers.

3.2.9 HFPEP_STREAM_CLOSE_REQUEST

Proxy -> HF-PEP.

Arguments:

1. ID.

Initiator or Responder requests SLEP to close stream. There is no confirmation.

3.2.10 HFPEP_STREAM_CLOSE_INDICATION

HF-PEP -> Proxy.

Arguments:

1. ID.
2. Reason:
 - a. Initiator Request
 - b. SLEP failure.

HF-PEP tells responder that stream is closed. Note that this close is handled at the SLEP layer, so there are no HF-PEP specific errors.

4 Proxy Operation

This section summarizes how a proxy will make use of the HF-PEP service and how it interacts with TCP. A proxy will provide both initiating and responding services.

4.1 Proxy Binding

An HF-PEP proxy (“proxy”) will bind to one or more STANAG 5066 servers using HFPEP_BIND. A proxy will typically bind on a single SAP to a given STANAG 5066 server, but **may** bind on multiple SAPs.

A proxy will handle all or selected TCP ports for selected IP addresses or ranges. These IP addresses will usually be the IP addresses of the target applications on the other side of the proxy pair. To achieve this, a proxy will generally need to operate at the IP router level.

4.2 Initiating Proxy Operation

An initiating proxy will work as follows:

1. Receive an inbound new TCP connection.
2. Determine, based on TCP parameters, which HF-PEP bind session to use.
3. Initiate an HF-PEP connection using HFPEP_STREAM_INIT_REQUEST with parameters as follows:
 - a. Address. The STANAG 5066 address of the chosen HF-PEP peer. The peer will typically be chosen based on the destination IP address of the inbound TCP connection.
 - b. Priority. This will generally be chosen based on destination port of the inbound TCP connection.
 - c. Compression. This will generally be chosen based on destination port of the inbound TCP connection.
 - d. Immediate Connect. This will usually be chosen to reduce latency. It may be appropriate to not use it for some TCP applications (ports) where connection reject after initial data is problematic.
 - e. Source Port, Destination Port, Source IP Address and Destination address will generally be taken directly from the inbound TCP connection. However, there may be more complex lookup.
4. If HFPEP_STREAM_INIT_REJECT is received, close the inbound TCP connection. If HFPEP_STREAM_INIT_CONFIRM is received, data is now handled in both direction, referenced by the Transfer ID.
5. Inbound TCP data is written using HFPEP_STREAM_DATA_REQUEST
6. Inbound HFPEP data received by HFPEP_STREAM_DATA_INDICATION is written to the TCP connection.
7. (Late reject procedure). If Immediate Connect is selected an HF_STREAM_INIT_REJECT may be received after data is written. If this happens, the inbound TCP connection is abandoned.

8. (Normal close procedure). TCP connection is closed by initiator, and stream is then closed using HFPEP_STREAM_CLOSE_REQUEST.
9. (Abnormal close). If HFPEP_STREAM_CLOSE_INDICATION is received, the inbound TCP connection is abandoned.

4.3 Responding Proxy Operation

An responding proxy will work as follows:

1. Receive an inbound new HF-PEP stream, with HFPEP_STREAM_INIT_REQUEST.
2. Open a TCP connection using Source Port, Destination Port, Source IP Address and Destination address taken directly or derived from HFPEP_STREAM_INIT_REQUEST parameters.
3. If TCP connection fails to open, reject the stream using HFPEP_STREAM_INIT_RESPONSE. Otherwise accept the stream using HFPEP_STREAM_INIT_RESPONSE.
4. Inbound TCP data is written using HFPEP_STREAM_DATA_REQUEST
5. Inbound HFPEP data received by HFPEP_STREAM_DATA_INDICATION is written to the TCP connection.
6. (Normal close procedure). HFPEP_STREAM_CLOSE_INDICATION is received and the TCP connection is closed.
7. (Abnormal close). The TCP connection is monitored, either by timeout or keepalive, appropriate to the destination port. If the proxy determines that the TCP connection is “dead”, then the TCP connection is closed and stream closed using HFPEP_STREAM_CLOSE_REQUEST.

4.4 When to Push Data

An HF-PEP proxy maps between a stream protocol and a PDU based service.

4.4.1 SLEP to TCP

When a HFPEP_STREAM_DATA_INDICATION provides a block of data it is sent to TCP and an TCP Push **should** be used at the end of writing this block of data to TCP.

4.4.2 TCP to SLEP

When data arrives from TCP, a series of data blocks will be written using HFPEP_STREAM_DATA_REQUEST. Where there is a large stream of data, blocks of maximum data block size can be written. When some data arrives over TCP a decision has to be made as to how long to wait for more data. There is a trade-off:

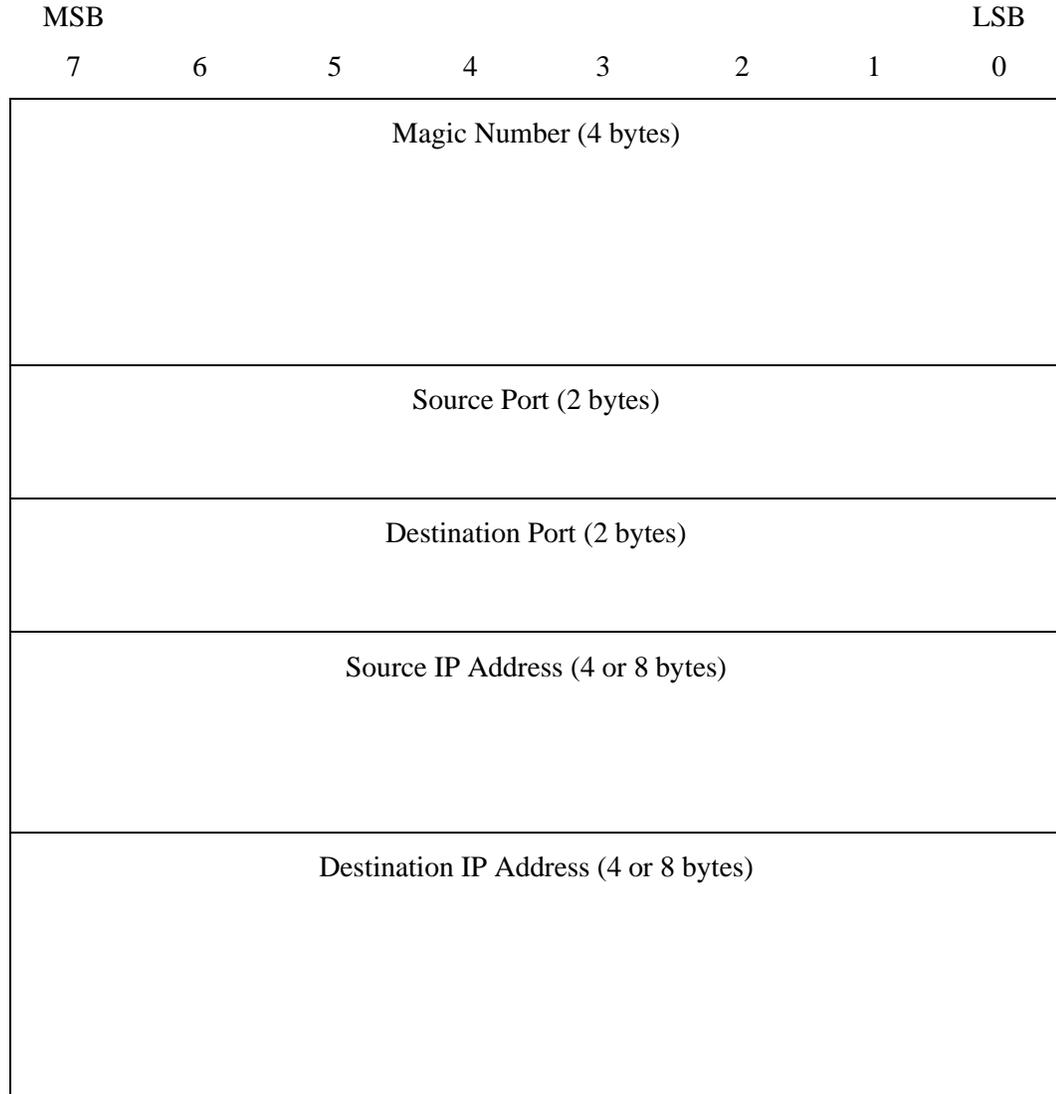
1. If the wait is too long, extra latency may be added.
2. If the wait is too short, too many data blocks with associated overhead are sent.

It is anticipated that the TCP side will generally be fast, so that a wait of a few hundred milliseconds will often be appropriate. This will be sufficient for many applications and will not

have significant HF impact. It may be appropriate to configure this for different applications and different TCP peers.

5 HF-PEP Protocol

5.1 HF-PEP Stream Header



The HF-PEP stream is encoded as follows:

1. The first four bytes are a “magic number”. This is used to detect misconfiguration of other SLEP applications and ensure correct protocol. The values are ASCII encoded with first character in first byte. Two values are defined:
 - a. “PEP4” (for IPv4)
 - b. “PEP6” (for IPv6).
2. Source Port and Destination Port are each encoded in two bytes in network byte order

3. Source and Destination IP address are encoded in four bytes for IPv4 and eight bytes for IPv6 in network byte order.

5.2 SLEP Reason Codes

SLEP applications can define reason codes to be carried in SLEP. HF-PEP defines a number of SLEP codes to communicate error information between a pair of HF-PEP proxies. In some cases, these errors can be used to communicate information back to the TCP peer.

Code	Description
1000	TCP general error. To cover generic failures not covered by specific subsequent codes
1001	Return IP is not routable. Used by a proxy when the source IP address is not locally routable, which would mean that no return IP traffic is possible.
1002	HF-PEP Magic Number not known or not supported.
1003	SLEP final-data failure. The final data of the stream failed to arrive after the close, so some data for the TCP stream may be missing.
1004	Return STANAG 5066 address not routable. The receiving proxy is not able to route IP traffic correctly back to the sender. A routing table error similar to 1001.
1005	HF-PEP Stream Header invalid.
1006	HF-PEP Capability not supported. Valid HF-PEP PDU requesting a capability not supported by receiver.
1007	Invalid destination IP address. IP address rejected (e.g., routing error)
1008	Connection refused. TCP rejection of requested connection.
1009	Connection time out.

6 HF-PEP Procedures

The HF-PEP service maps very closely onto the SLEP Stream Service. Each HF-PEP service maps onto the equivalent SLEP service.

The additional parameters in SLEP_STREAM_INIT_REQUEST are sent as the HF-PEP stream header which is sent on the stream prior to data. This is decoded by the responder and provided in SLEP_STREAM_INIT_INDICATION.

7 Default SAP

It is recommended that SAP 13 is used as the default SAP for HF-PEP.