Internet Engineering Task Force
Internet Draft
Category: Informational
Expires: August 2017

M. Foschiano
K. Ghosh
M. Mehta
Cisco Systems

February 2017

Cisco Systems' Encapsulated Remote Switch Port Analyzer (ERSPAN)
draft-foschiano-erspan-03.txt

Abstract

This document describes an IP-based packet capture format that can be used to transport exact copies of packets to a network probe to analyze and characterize the operational load and protocol distribution of a network as well as to detect anomalies such as network-based worms or viruses. This replication and transport mechanism operates over one or multiple switch or router ports whose traffic can be mirrored and forwarded to a destination device in charge of traffic analysis and reporting.

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire in August 2017.

Table of Contents

1. Introduction

Today one of the most common network monitoring and troubleshooting tools is the so-called Switch Port Analyzer (SPAN) feature, also known as port mirroring. It allows a user to monitor network traffic non-intrusively and send a copy of the monitored traffic to a local or remote device, which can be a sniffer, an intrusion detection system (IDS), or other type of network analysis tool.

Some of the most popular use cases of SPAN are:

1. Debugging network problems by tracking control/data frames

2. Monitoring Voice-over-IP (VoIP) packets for delay and jitter analysis

3. Monitoring network transactions for latency analysis

4. Monitoring network traffic for anomaly detection

SPAN can operate locally and mirror traffic to other ports of the same source device, or it can operate remotely mirroring traffic to a different network device that is layer-2 adjacent to the source.

This document describes the frame formats used by the "encapsulated remote" extension of the SPAN feature, which supports remote monitoring of network traffic across a generic IP transport.

2. ERSPAN's Basic Principles of Operation

The ERSPAN feature enables a network device to deliver a copy of the monitored traffic to a destination system through an IP network.

To do so, a source network device filters the portion of the traffic the user is interested in, makes a copy of it and then encapsulates each replicated frame into the payload of a special "container super-frame". Such frame carries enough additional information for it to be properly routed all the way to the receiver device and for such device to be able to extract and fully restore the original monitored Ethernet frame (or a selected portion of it).

The receiver device can be another networking device that supports ERSPAN decapsulation or, when direct connectivity is available, even a (non passive) network probe.

The frame formats used to enable such capabilities are described in the following sections.

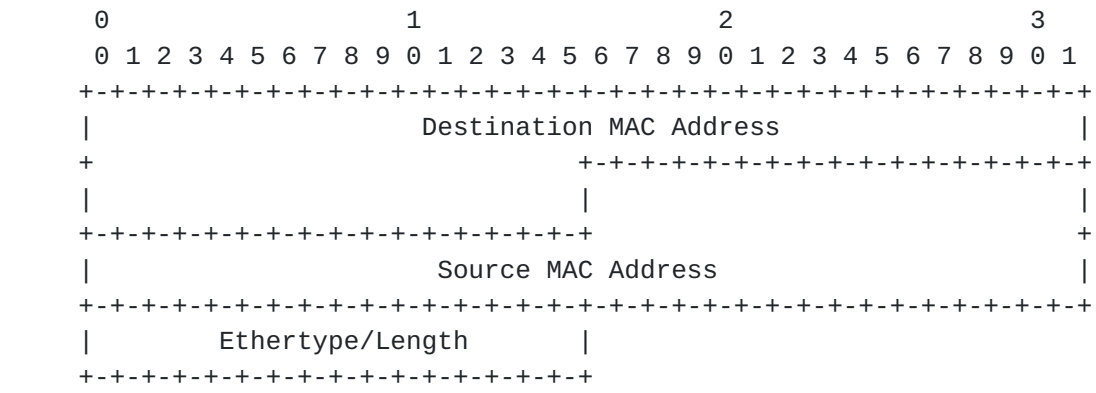
3. ERSPAN's Common Encapsulation Components

The ERSPAN packet format is GRE-based [RFC1701], and for it most legacy implementations assume an underlying IPv4 [RFC791] over Ethernet [802.3] transport. However, even though IPv4 is normally used, IPv6 support has become a requirement too.

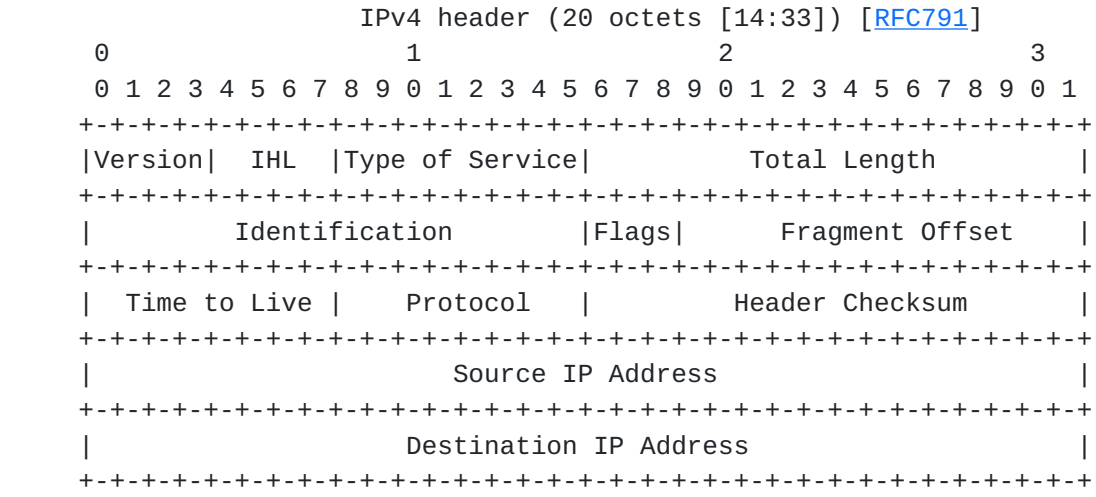
The use of the IP protocol as part of the transport is critical to be able to carry the mirrored traffic across any IP-based network. The GRE component instead is particularly important to be able to piggyback different data formats along with the copy of the original frame.

The ERSPAN frame format's common components are described below:

802.3 Ethernet MAC header (14 octets [0:13])

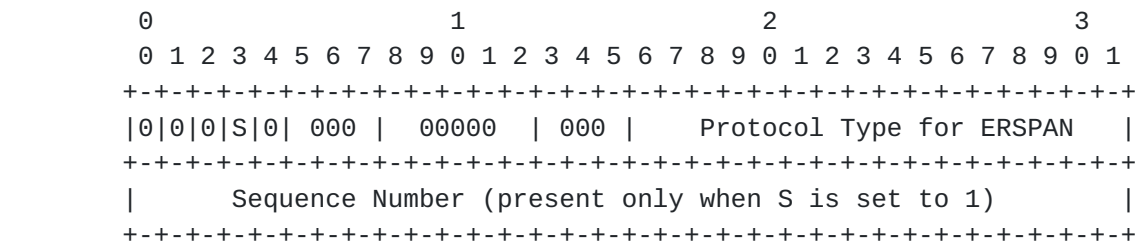


IPv4 header (20 octets [14:33]) [RFC791]



(Above, for simplicity's sake the described frame format is IPv4's, yet IPv6 can be supported too in certain implementations.)

GRE header for ERSPAN encapsulation (8 octets [34:41])



Note that in the above GRE header [RFC1701] out of the C, R, K, S, s, Recur, Flags, Version fields only S (bit 03) may be set to 1. The other fields are always set to zero.

4. ERSPAN Types and Specific Sub-Headers

Different frame variants known as "ERSPAN Types" can be distinguished based on the GRE "Protocol Type" field value: Type I and II's value is 0x88BE while Type III's is 0x22EB [ETYPES].

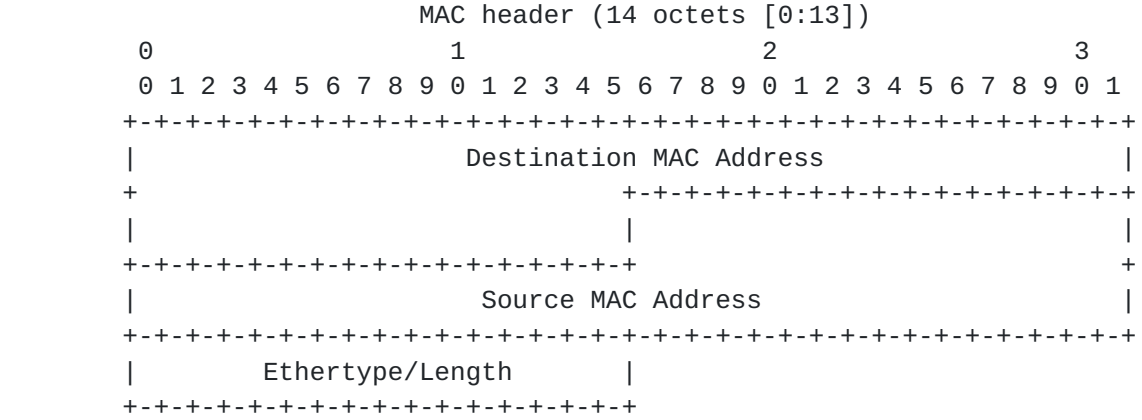
On top of the GRE header, some ERSPAN formats may include an additional "feature" header described in the sections below.

4.1 ERSPAN Type I

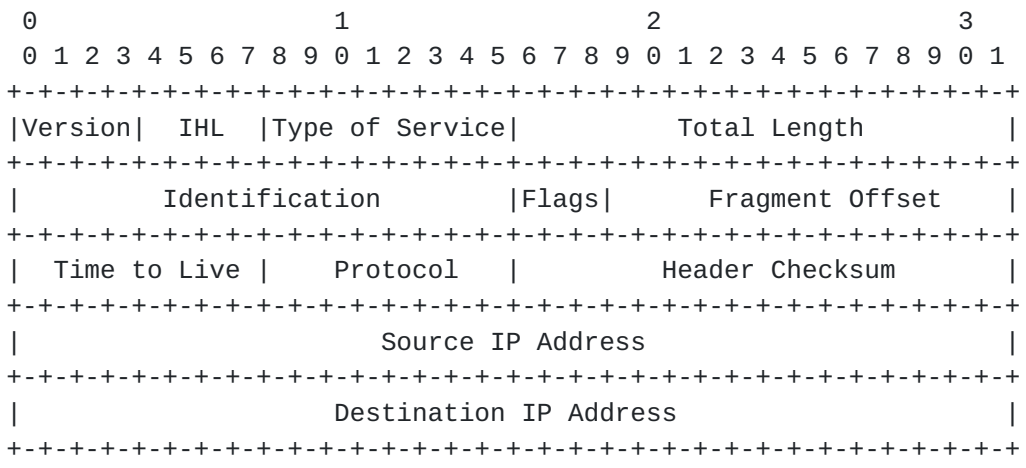
The Type I ERSPAN frame format is based on the barebones IP+GRE encapsulation (as described above) on top of the raw mirrored frame. The GRE protocol type for Type I must be programmable and is currently set to 0x88BE on supported devices. This format can be terminated in hardware so that the original raw frame is decapsulated and sent to the destination device as originally received on the source device.

The various sections of the frame format are described below:

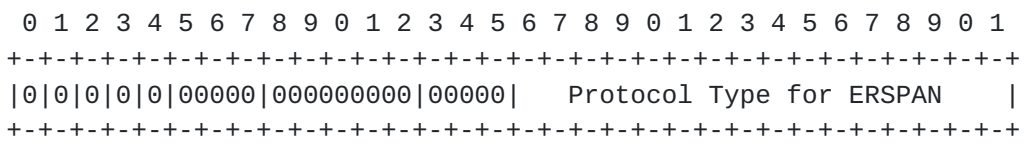
MAC header (14 octets [0:13])



IPv4 header (20 octets [14:33])



GRE header for ERSPAN type I encapsulation (4 octets [34:37])



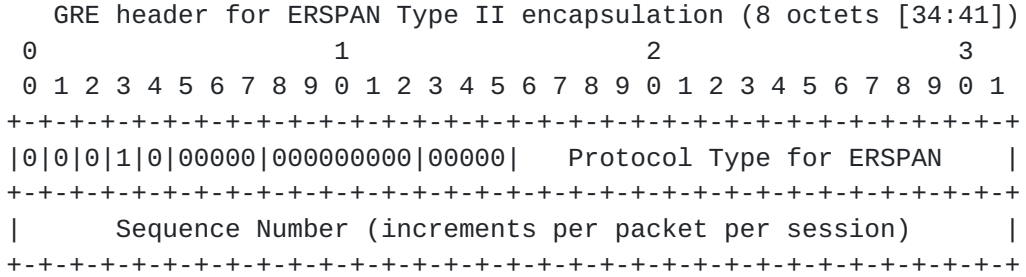
This encapsulation adds 38 octets to the original frame length: 14 (MAC) + 20 (IP) + 4(GRE). The advantage of this format is its size, which is more compact than Type II's and therefore can be less expensive to implement.

4.2 ERSPAN Type II

Historically, ERSPAN Type II's header was the first variant to be extensively used by Cisco Systems products.

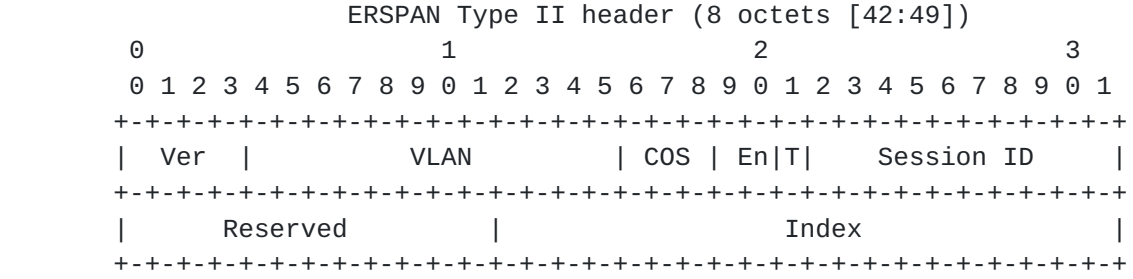
In this case, in the GRE header (see below) out of the C, R, K, S, s, Recur, Flags, Version fields the S bit is set to 1 while the others are set to zero, hence a Sequence Number field is present in Type II's GRE header.

GRE header for ERSPAN Type II encapsulation (8 octets [34:41])

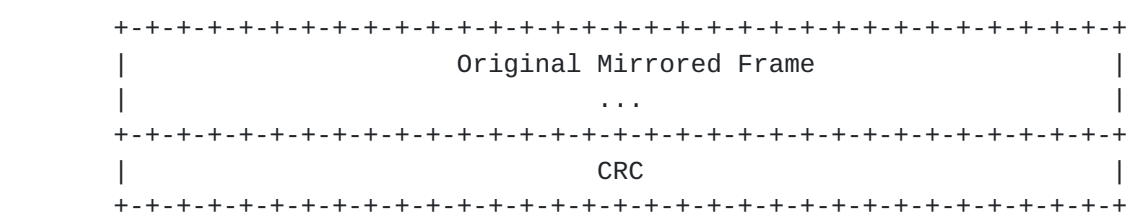


ERSPAN Type II's frame format also adds a special 8-octet ERSPAN "feature" header on top of the MAC/IPv4/GRE headers to enclose the raw mirrored frames.

The ERSPAN Type II feature header is described below:



The above 8-octet header is immediately followed by the original mirrored frame and then by the standard 4-octet Ethernet CRC:



Therefore, the ERSPAN Type II encapsulation adds to the original frame (sans its FCS) a composite header comprising: 14 (802.3) + 20 (IP) + 8 (GRE) + 8 (ERSPAN) octets, in addition to a new trailing 4-octet Ethernet CRC value that is calculated based on the entire ERSPAN frame.

Note that an 802.1Q encapsulation [802.1Q] would add 4 additional octets but not reduce the Ethernet MTU size of the container frame.

Also note that in this context (and in the context of Type I) the copy of the original mirrored frame does not include the original CRC octets, which are not preserved in the encapsulation process and need to be recomputed in case of decapsulation by a networking device. This means that on the receiving device it is not possible to verify the CRC correctness of the original frame (in these cases the assumption is simply that only uncorrupted frames are mirrored).

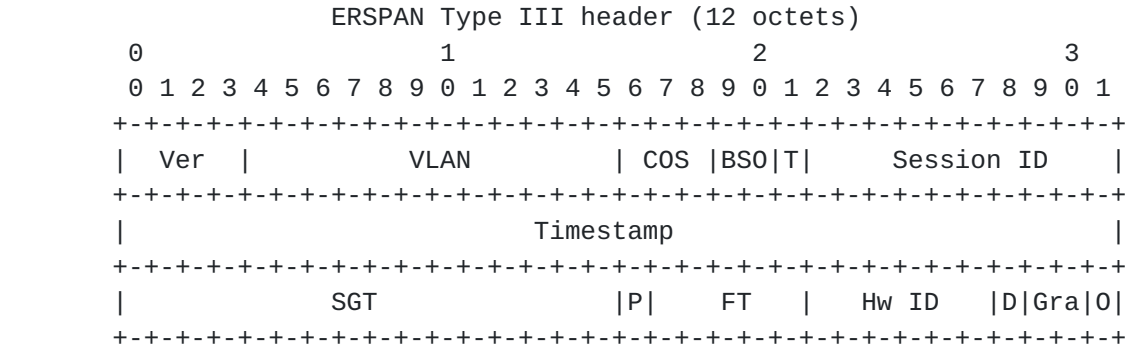
The various fields of the above header are described in this table:

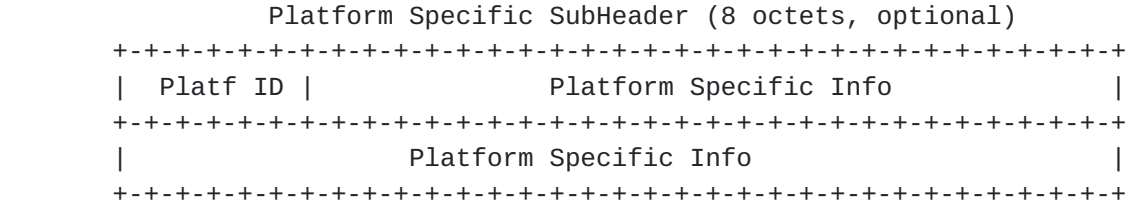
Field	Position [octet:bit]	Length (bits)	Definition
Ver	[42:0]	4	ERSPAN Encapsulation version. This indicates the version of the ERSPAN encapsulation specification. Set to 0x1 for Type II.
VLAN	[42:4]	12	Original VLAN of the frame, mirrored from the source. If the En field is set to 11, the value of VLAN is undefined.
COS	[44:0]	3	Original class of service of the frame, mirrored from the source.
En	[44:3]	2	The trunk encapsulation type associated with the ERSPAN source port for ingress ERSPAN traffic. The possible values are: 00-originally without VLAN tag 01-originally ISL encapsulated 10-originally 802.1Q encapsulated 11-VLAN tag preserved in frame.
T	[44:5]	1	This bit indicates that the frame copy encapsulated in the ERSPAN packet has been truncated. This occurs if the ERSPAN encapsulated frame exceeds the configured MTU.
Session ID (ERSPAN ID)	[44:6]	10	Identification associated with each ERSPAN session. Must be unique between the source and the receiver(s). (See section below.)
Reserved	[46:0]	12	All bits are set to zero
Index	[47:4]	20	A 20 bit index/port number associated with the ERSPAN traffic's port and direction (ingress/egress). N.B.: This field is platform dependent.

4.3 ERSPAN Type III

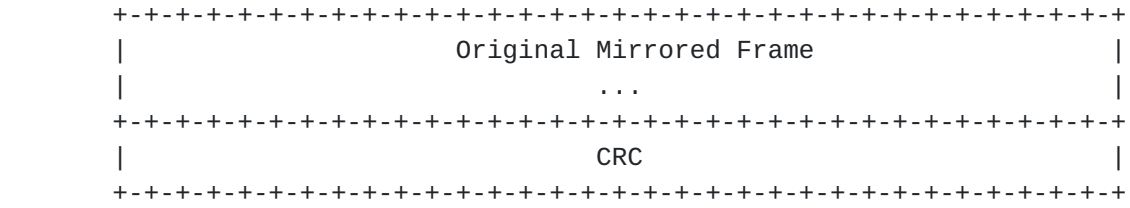
Type III introduces a larger and more flexible composite header to support additional fields useful for applications such as network management, intrusion detection, performance and latency analysis, etc. that require to know all the original parameters of the mirrored frame, including those not present in the original frame itself.

The ERSPAN Type III composite header includes a mandatory 12-octet portion followed by an optional 8-octet platform-specific sub-header as described below:





The above composite header is immediately followed by the original mirrored frame and then by the standard 4-octet Ethernet CRC.



See section 7 below for a discussion on how to encapsulate the original packet's Ethernet CRC.

The various fields of the above header are described in this table:

Header Fields in Common with ERSPAN Type II

Field	Length (bits)	Definition
Ver	4	ERSPAN Encapsulation version. For Type-III packets it is set to 0x2.
VLAN	12	VLAN of the frame monitored by an ERSPAN source session: for ingress monitor this will be the original source VLAN whereas for egress monitor this will be the destination VLAN.
COS	3	Class of Service of the monitored frame. Ingress or egress CoS value is to be used depending on the monitor type/direction.
T	1	This bit indicates that the frame copy encapsulated in the ERSPAN packet has been truncated. This occurs if the ERSPAN encapsulated frame exceeds the configured MTU and hence has to be truncated.
Session ID (ERSPAN ID)	10	Identification associated with each ERSPAN session. Must be unique between the source and the receiver(s). (See section below.)

ERSPAN Type-III Header Specific Fields

Field	Length (bits)	Definition
BSO (Bad/Short/ Oversized)	2	A 2-bit value indicating the integrity of the payload carried by ERSPAN: 00 --> Good frame with no error, or unknown integrity 11 --> Payload is a Bad Frame with CRC or Alignment Error 01 --> Payload is a Short Frame 10 --> Payload is an Oversized Frame
Timestamp	32	The timestamp value needs to be derived from a hardware clock which is synchronized to the system-clock. This 32-bit field should support at least a

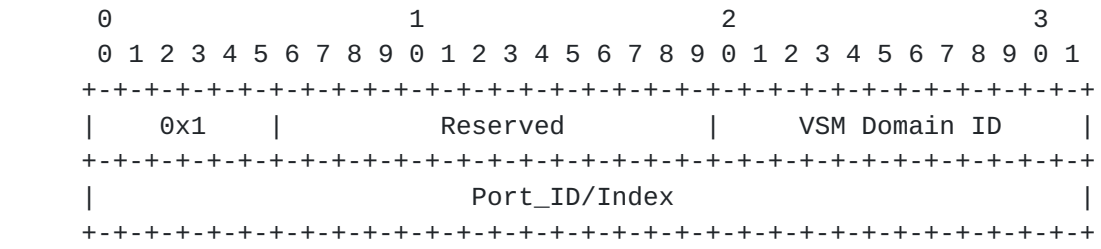
		timestamp granularity of 100 microseconds (see the Timestamp Granularity field).
SGT	16	Security Group Tag of the monitored frame.
P	1	This bit indicates that the ERSPAN payload is an Ethernet protocol frame (PDU frame).
FT (Frame Type)	5	This field can be used to reconstruct the original frame's encapsulation if it is supported by the receiver. This field may also be used by ERSPAN engines to indicate that the mirrored frame's L2 encapsulation header (or a portion of it) was skipped and not included in the ERSPAN packet. 00000 --> Ethernet frame (802.3 frame) 00010 --> IP Packet Other values --> Reserved for future use
Hw (Hardware) ID	6	Unique identifier of an ERSPAN engine within a system.
D (Direction)	1	Indicates whether the original frame was SPAN'ed in ingress or in egress. Ingress (0) or Egress (1).

Gra (Timestamp Granularity)	2	Time unit to be supported for time-stamping:
		00b --> granularity = 100 microseconds
		01b --> granularity = 100 nanoseconds
		10b --> granularity = IEEE 1588 TimeRepresentation format (see definition below; with nanoseconds portion stored in the Timestamp field and seconds portion stored in the ERSPAN platform-dependent sub-header)
		
		struct TimeRepresentation
		{
		UInteger32 seconds;
		UInteger32 nanoseconds;
		};
		11b --> user configurable time unit (platform dependent, for example specific to an isolated non-synchronized system with very high local accuracy)

O (Optional Sub-header)	1	The O flag indicates whether or not the optional platform-specific sub-header is present. If it's present, the next octet indicates the platform specific format used (Platf ID). The ERSPAN payload starts after the O flag when O == 0b or after 8 octets when O == 1b.
Platf ID	6	Platform identifier that needs to be recognized in order to parse the optional platform-specific sub-header that follows.
Platform Specific Info	58	Platform Specific Information field. It is a container for data that is used by a specific set of devices only.

Currently only the following Platform ID values are used and correspond to defined Platform Specific Info field formats:

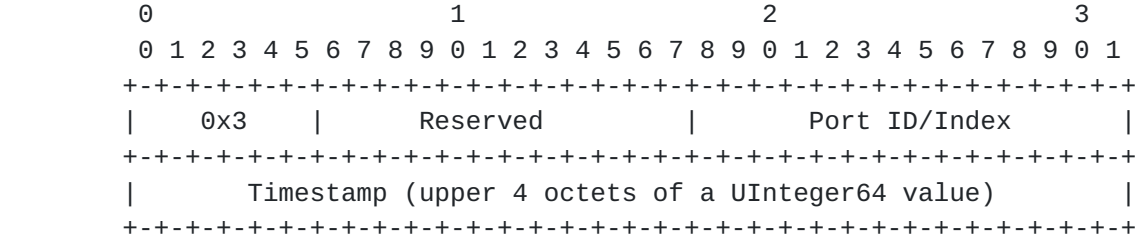
Platf ID Value	Description
0x0	Reserved. In some implementations it is used as an alias to 0x07.
0x1	Corresponds to the following format:



When the 0x1 value is used, the timestamp in the base header is in 100 microseconds and the Gra field is set to '00'.
The VSM Domain ID field is the identifier of a Cisco Nexus VSM domain.

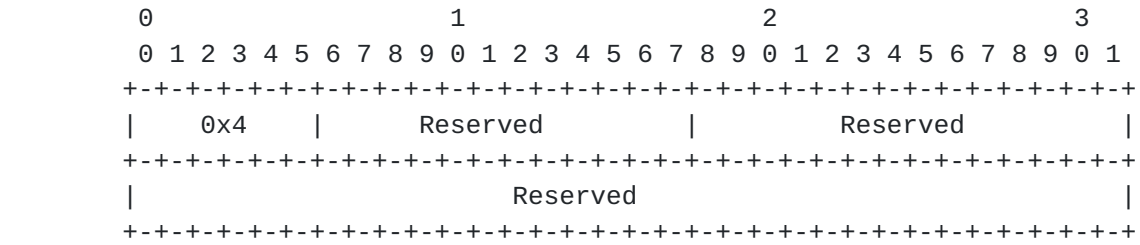
0x2	Reserved

0x3 Corresponds to the following format:



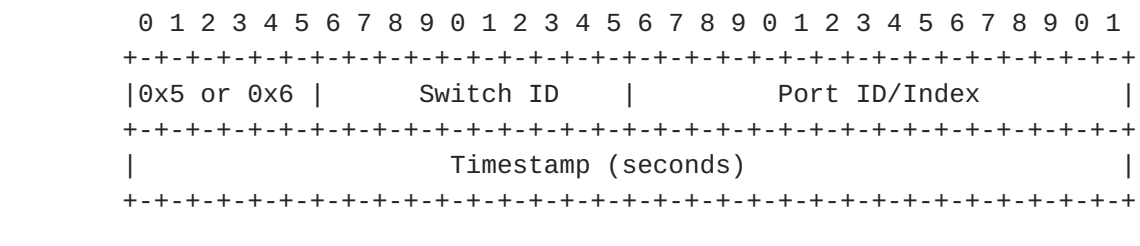
The granularities supported when the Platform ID is set to 0x3 are 00b (100 microseconds), 01b (100 nanoseconds) and 11b (nanoseconds).
An unsigned 64-bit timestamp value can be derived from combining the base ERSPAN header's 32-bit value (lower 4 octets) with the Platform Specific Info's 32-bit value (upper 4 octets) and can be interpreted based on the granularity value set in the Gra field.

0x4 Corresponds to the following format:



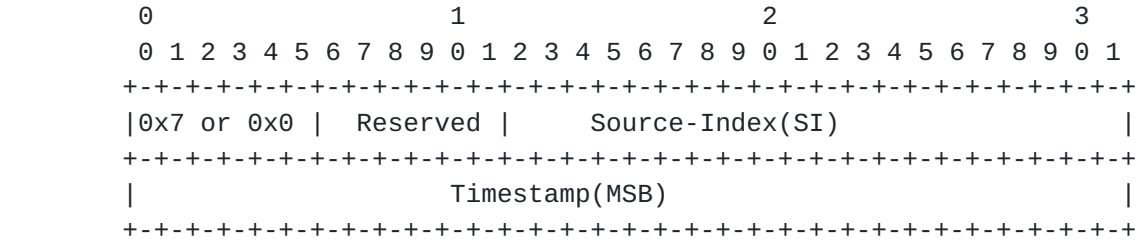
When the 0x4 value is used, the timestamp value in the base header represents a UInteger32 timestamp value expressed in 100 microsecond units (Gra field = '00').

0x5-0x6 Correspond to the following format:



When the 0x5 or the 0x6 value is used, the timestamp value in the base header represents the IEEE 1588 nanoseconds field while the timestamp value in the Platform Specific Info represents the IEEE 1588 seconds. The Gra field is set to '10'. Switch ID is a value configurable in the CLI to identify a source switch at the receiving end. Port ID identifies the source switch port for the SPAN'd traffic.

0x7 Corresponds to the following format:



When the 0x7 value is used, an 8-octet timestamp can be derived from the base ERSPAN header's Timestamp field (least significant four octets) combined with the most significant 4 octets present in corresponding field of the sub-header. The "Gra" field value is 0x3(nanoseconds). For ingress ERSPAN the lower 8 octets of the 20-octet SI field are populated with the port index while the upper 12 bits are populated with an index of the group the traffic's source port belongs to.

0x8-0x63 Reserved

It should be noted that the various supported header fields above can be used in regular ERSPAN applications to mirror even an errored frame or a bridge PDU (BPDU) frame and to preserve the original trunking encapsulation and VLAN number. In addition, crucial time-stamping information as well as other informational fields can be added to each ERSPAN frame on the source device during the frame mirroring/replication process.

The use of various feature header fields is discussed in the following sections.

5. ERSPAN Session Numbers

An ERSPAN session is a configuration parameter that the user can employ to differentiate between mirrored traffic. It is typically associated to a single or to a group of physical or logical entities, such as one or more ports or one or more VLANs, whose traffic requires mirroring. In general, a session number identifies a subset of the traffic of a source device that a user wishes to replicate and analyze. Session numbers therefore represent a context in which a particular stream of mirrored frames can be placed and by which it can be identified. Such context is usually meaningful when associated to a particular source-destination device pair.

As a matter of fact, within the ERSPAN IP header two key identification parameters are the source IP address and the destination IP address of the ERSPAN packets: the former uniquely identifies the source device of the mirrored frames while the latter uniquely identifies the destination device, which is to terminate the flow of ERSPAN packets.

Different source-destination device pairs can reuse session numbers as long as they represent fully disjoint ERSPAN contexts.

6. Ethernet and IP fields

Noteworthy parameters in the Ethernet and IP portion of an ERSPAN frame are its Quality of Service (QoS) fields, CoS and ToS, which users can program on a per session basis in such a way as to meet the priority and delay requirements of their traffic analysis applications.

For example, in certain conditions of network congestion it may be desirable to configure a higher QoS priority for ERSPAN frames to allow them to reach the analysis device despite congestion and so allow the network administrator to troubleshoot potential bandwidth utilization issues. In other cases, instead, dropping of ERSPAN traffic may not constitute a problem for the network administrator and therefore lowering of the ERSPAN QoS priority can be considered completely acceptable.

In addition, the IP Identification field of ERSPAN Type II packets is sometimes used to distinguish between different ERSPAN source engines by writing in the field's upper 6 bits a unique fixed ERSPAN Engine ID value while incrementing the remaining 10 bits for each mirrored frame. This parameter provides an additional level of granularity for traffic identification (and therefore feature flexibility) in addition to the source device's IP address and the ERSPAN session number, as described above.

On the other hand, for the purpose of identifying different ERSPAN source engines, ERSPAN Type III uses the dedicated Hardware ID field instead.

7. Use of Other Relevant ERSPAN Fields

The En(capsulation) field is used to distinguish the VLAN encapsulation format of the original mirrored frame: IEEE 802.3/untagged (no VLAN number in the frame), IEEE 802.1Q tagging format or Cisco Systems' ISL tagging format. Some implementations may strip the original VLAN encapsulation during encapsulation (for example due to hardware constraints) while others may preserve it in the frame. Hence this field can be used to differentiate the various cases.

The T(runcated) field is an indication of whether the original frame had to be truncated to fit into the MTU used for ERSPAN transmission. Note that ERSPAN may recalculate and overwrite the CRC of the original Ethernet frame when adding the ERSPAN L2/IP/GRE encapsulation, so even truncated frames can reach the analysis device with a good CRC. However, the T field can be used to indicate that the original (good or bad) CRC was preserved (i.e., not truncated from the original frame).

8. Security Considerations

Source and destination IP addresses used for ERSPAN must be fully routable addresses, so that for example in certain implementations users could even ping such addresses to ascertain the aliveness of the corresponding source or destination device.

Moreover, specifically in the case of a destination ERSPAN device, its IP address oftentimes represents a "front end" or proxy to an IP-less device such as a passive sniffer or other analysis tool. This proxying capability is extremely beneficial when the end device acts in stealth mode and cannot appear as an active and reachable node of the network.

Although ERSPAN does not offer specific capabilities for security such as authentication or encryption, the IP TTL of the ERSPAN packets is configurable and can be used to limit the reach of ERSPAN packets within an IP cloud. In addition, as any standard IP packet, ERSPAN packets could be transported in regular tunnels, such as IPSec or MPLS tunnels, however by design they have the IP Don't Fragment bit set to avoid the need for packet reassembly.

9. IANA Considerations

This document has no actions for IANA.

10. Changes from the Previous Version

Added Kalyan as co-author. Made minor edits.

11. Acknowledgements

Various people have contributed to the definition of the ERSPAN format and have provided input and reviewed this document. For both contributions the authors would like to thank, in alphabetical order, Ian Cox, Nicolas Delecroix, Sonia Gulrajani, Archana Kamath, Nageswara Ponugoti and Ayushma Sinha, as well as Liangda Ho for finding a typo. For fundamental contributions to the invention of the various ERSPAN types the authors would especially like to thank Tom Edsall and Suresh Gurajapu.

12. Normative References

[802.3] IEEE Std 802.3-2012, IEEE Standard for Ethernet.

[802.1Q] IEEE Std 802.1Q-2003, IEEE Standards for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks.

[RFC791] Postel, J. (ed.), "Internet Protocol - DARPA Internet Program Protocol Specification," RFC 791, USC/Information Sciences Institute, September 1981.

[RFC1701] Hanks, S., Li, T., Farinacci, D. and P. Traina, "Generic Routing Encapsulation", RFC 1701, October 1994.

[ETYPES] IEEE Ethertype List: http://standards.ieee.org/develop/regauth/ethertype/eth.txt.

Authors' Addresses

Marco Foschiano, Cisco Systems, Inc., Via Torri Bianche 7, Vimercate, MI, 20059, Italy, email address: mfoschiano@gmail.com

Kalyan Ghosh, Cisco Systems Inc., 3625 Cisco Way, SAN JOSE, CALIFORNIA 95134, USA, email address: kghosh@cisco.com

Munish Mehta, Cisco Systems Inc., 3625 Cisco Way, SAN JOSE, CALIFORNIA 95134, USA, email address: mmehta@cisco.com