Internet-Draft
Document: draft-boschi-ipfix-biflow-01.txt
Expires: April 2006

E. Boschi
Hitachi Europe
B. Trammell
CERT/NetSA

October 2005

Biflow implementation support in IPFIX
draft-boschi-ipfix-biflow-01.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

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."

The list of current Internet-Drafts can be accessed at http://www.ietf.org/ietf/1id-abstracts.txt.

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html.

This Internet-Draft will expire on April 27, 2006.

Copyright Notice

Copyright (C) The Internet Society (2005).

Biflow implementation support in IPFIX

Abstract

This document describes how bidirectional flows (biflows) can be implemented in the IP Flow Information Export (IPFIX) protocol. We propose a variety of methods for biflow export, including an extension to the IPFIX Information Model that allows the specification of biflow semantics and a set of counters needed for biflow processing.

Table of Contents

1. Introduction

Many flow analysis tasks benefit from easy association of the upstream and downstream flows of a bidirectional communication, e.g., separating answered and unanswered TCP requests, calculating round trip times, etc. Metering processes that are not part of an asymmetric routing infrastructure are well positioned to observe bidirectional flows, and IPFIX is very nearly complete as a solution for exporting biflow data.

We propose several methods to export biflow information using IPFIX. Two of these methods do not require any changes to the IPFIX information model, at the expense of export and collection efficiency. The final method requires an extension to the IPFIX

Information Model to include some biflow-related Information Elements (IEs).

2. Terminology

Uniflow (unidirectional flow)

A uniflow is a set of IP packets passing an Observation Point in the network during a certain time interval. All packets belonging to a particular Uniflow have a set of common properties. Each property is defined as the result of applying a function to the values of:

1. One or more packet header fields, transport header fields, or application header fields.

2. One or more characteristics of the packet itself.

3. One or more fields derived from packet treatment.

A packet is said to belong to a Uniflow if it completely satisfies all the defined properties of the Uniflow.

This definition is simply the IPFIX definition of an IP Flow [IPFIX-PROTO].

Biflow (bidirectional flow)

A biflow is the product of matching the two uniflow sides of a bidirectional communication session (e.g., TCP session, UDP DNS question and answer) into a single entity. The semantics of "source" and "destination" information elements within the context of a given biflow are variable; since "source" and "destination" are not derived directly from their respective packet header fields, the IPFIX definition of IP Flow alone is not sufficient to describe biflows.

3. Biflow Implementation Strategies

This section introduces the problems associated with representing biflows in IPFIX, and presents a variety of strategies for implementing biflow support.

3.1 Biflow Semantics

When handling uniflows, the semantics of "source" and "destination" information elements are clearly defined by the semantics of the underlying packet header data. When grouping biflows into single IPFIX data records, the definitions of "source" and "destination" become less clear.

The most basic method for classifying the two addresses in a biflow is to define the source address of the flow as the source address of the first packet seen in that flow by the metering process. Some metering technologies may attempt to improve upon this method using some knowledge of the transport or application protocols (e.g., TCP flags, DNS question/answer counts) in order to define the source address of the flow as the connection or transaction initiator. In either case, the design is the same: one of the underlying uniflows is assumed to be in the "forward" direction, and one in the "reverse" direction; the "forward" uniflow is selected based upon some characteristic of the connection itself.

Another way to classify biflow addresses is by perimeter; in this method, a metering process discriminates between "inside" and "outside" the network, and defines the source address as the address on one side of this perimeter (generally the "outside" address; defining source loosely as "attacker"). Perimeter biflows may require additional information elements to identify the flow initiator, if such functionality is supported by the metering process. This revision of this draft does not address perimeter biflows further.

These two are by no means an exhaustive list of biflow semantics. However, IPFIX should aim to provide better support for the simpler, more common semantics in preference to more exotic schemes.

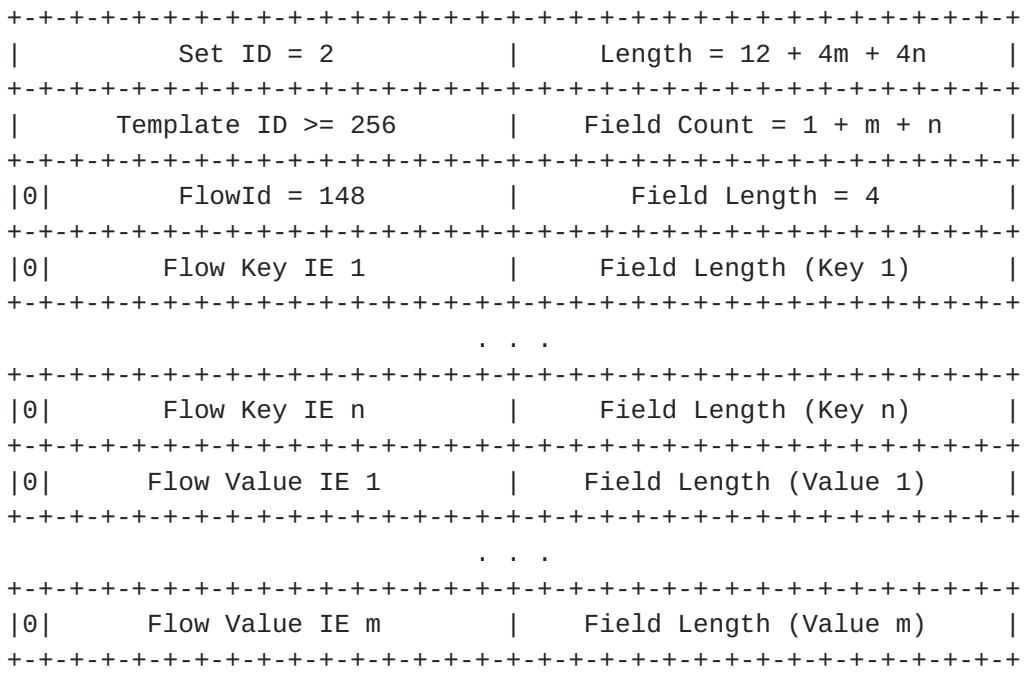
3.2 Two-Record Flow Association using flowId IE

The simplest way to implement biflow support using IPFIX is to sidestep the single-record unification entirely, and annotate two uniflow records as being part of the same biflow by using the existing flowId information element. If an exporting process supports biflow export via this method, it is recommended that it export the two flow data records comprising the biflow appear sequentially, within the same data set, in order of start time. This ensures that the collecting process does not need to cache more than a single flow to support biflow reassembly on its end.

This method has the advantage of simplicity and minimal impact on the protocol. However, as two uniflows in a single biflow share all flow key data, it is not the most efficient method for transporting biflow data.

3.2.1 Example

Both data records representing each biflow are described by the same template, which follows the pattern below:



The fields have been simply distinguished by FlowID (that identifies the two uniflows as part of the same biflow), flow key fields and flow value IEs to be exported.

3.3 Multiple Record Flow Aggregation using flowId IE

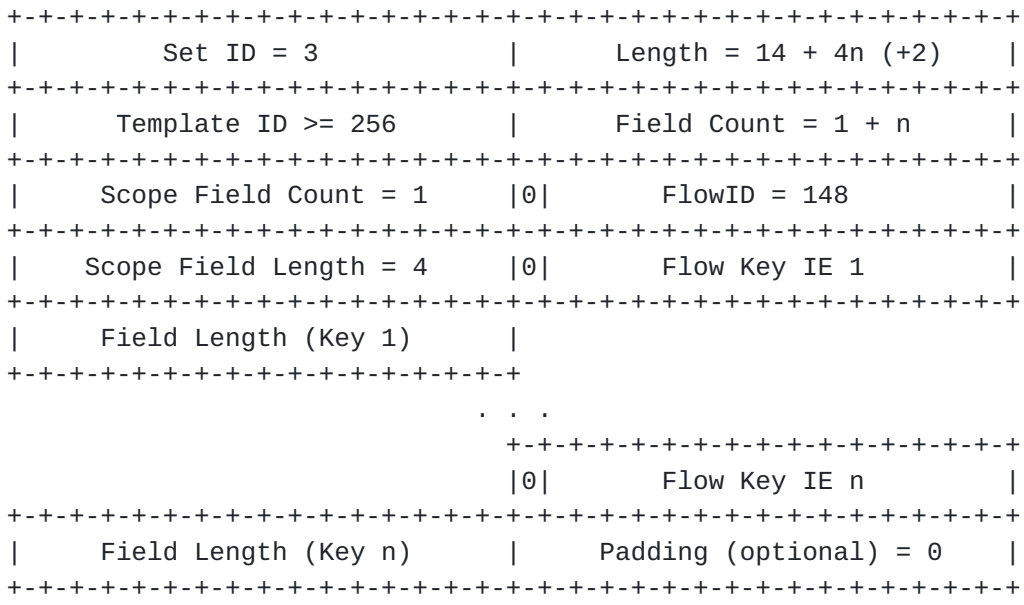
The main shortcoming of the previous method is the replication of information exported, since two uniflows in a biflow share all flow key data. This second method aggregates the flow fields common to both uniflows sending them in an Option record with FlowID as scope. The FlowID IE represents the identifier of the biflow. The two uniflow records part of the same biflow export the flow information (but not the flow key fields), the flowID and the source or the directionDomain IE (see section 3.4.1.15) to identify the two uniflows.

Using the source doesn't require any extension to the Information Model, but exporting three records per biflow and using an IP address (especially an IPv6 address) to identify

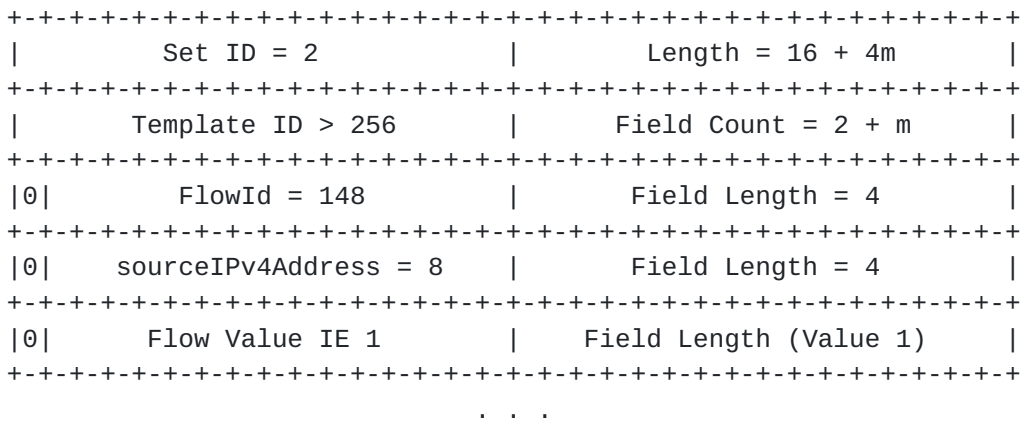
If an exporting process supports biflow export via this method, it MUST ensure that the option record is sent first. The exporter sends the flow key fields just once and subsequently refers to them using the flowID. This method makes sense when the records are exported at regular intervals but doesn't introduce an improvement if the data is exported just at the end of the flow.

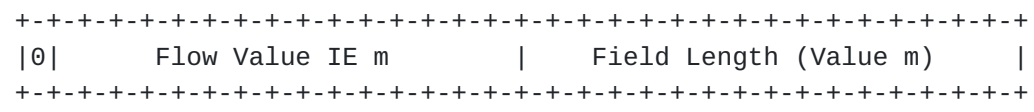
3.3.1 Example

The flow keys are exported in an options data record described by a template following the pattern below:



Each of the flow value data records is defined by a template following the pattern below:





3.4 Single Record biflows using directionDomain IE

Biflows can be represented in IPFIX using a single data record per flow by extending the IPFIX information model. First, a set of "reverse" counters are defined to count packets sent by the biflow destination, and the current "forward" counters are defined to count packets send by the biflow source. Second, an additional information element, directionDomain, is defined to specify the semantics of biflow source and destination.

The reverse counter information elements MUST be semantically defined by a directionDomain information element. The

directionDomain IE can appear in the biflow record itself, or be scoped to the sourceId or templateId associated with the biflow record.

Note that no reverse post-multicast counters are defined, because multicast flows are inherently unidirectional.

3.4.1 New Information Elements for Single Record biflows

The following information elements are required to support single record biflows:

3.4.1.1 reverseOctetDeltaCount

Description:
The number of octets since the previous report (if any) in packets sent by the biflow destination for this biflow. The number of octets includes IP header(s)and IP payload.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: 216
Status: proposed
Units: octets

3.4.1.2 reversePostOctetDeltaCount

Description:
The definition of this Information Element is identical to the definition of Information Element 'reverseOctetDeltaCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the observation point.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: 217
Status: proposed
Units: octets

3.4.1.3 reverseOctetDeltaSumOfSquares

Description:
The sum of the squared numbers of octets since the previous report (if any) in packets sent by the biflow destination for this biflow. The number of octets include IP header(s) and IP payload.
Abstract Data Type: unsigned64

ElementId: 218
Status: proposed
Units: octets

3.4.1.4 reverseOctetTotalCount

Description:
The total number of octets in packets sent by the biflow destination since the Metering Process (re-)initialization for this Observation Point. The number of octets includes IP header(s) and IP payload.
Abstract Data Type: unsigned64
Data Type Semantics: totalCounter
ElementId: 219
Status: proposed

Units: octets

3.4.1.5 reversePostOctetTotalCount

Description:
The definition of this Information Element is identical to the definition of Information Element 'reverseOctetTotalCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the observation point.
Abstract Data Type: unsigned64
Data Type Semantics: totalCounter
ElementId: 220
Status: proposed
Units: octets

3.4.1.6 reverseOctetTotalSumOfSquares

Description:
The total sum of the squared numbers of octets in packets sent by the biflow destination for this biflow since the Metering Process (re-)initialization for this Observation Point. The number of octets include IP header(s) and IP payload.
Abstract Data Type: unsigned64
ElementId: 221
Status: proposed
Units: octets

3.4.1.7 reversePacketDeltaCount

Description:
The number of packets since the previous report (if any) for this biflow sent by the biflow destination.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: 222
Status: proposed
Units: packets

3.4.1.8 reversePostPacketDeltaCount

Description:
The definition of this Information Element is identical to the definition of Information Element 'reversePacketDeltaCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the observation point.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter

ElementId: 223
Status: proposed
Units: packets

3.4.1.9 reversePacketTotalCount

Description:
The total number of packets sent by the biflow destination for this biflow at the since the Metering Process (re-) initialization for this Observation Point.
Abstract Data Type: unsigned64
Data Type Semantics: totalCounter
ElementId: 224
Status: proposed
Units: packets

3.4.1.10 reversePostPacketTotalCount

Description:
The definition of this Information Element is identical to the definition of Information Element 'reversePacketTotalCount', except that it reports a potentially modified value caused by a middlebox function after the packet passed the observation point.
Abstract Data Type: unsigned64
Data Type Semantics: totalCounter
ElementId: 225
Status: proposed
Units: packets

3.4.1.11 reverseDroppedOctetDeltaCount

Description:
The number of octets since the previous report (if any) in packets sent by the biflow destination for this biflow dropped by packet treatment. The number of octets include IP header(s) and IP payload.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: 226
Status: proposed
Units: octets

3.4.1.12 reverseDroppedPacketDeltaCount

Description:
The number of packets since the previous report (if any) sent by
the biflow destination for this biflow dropped by packet treatment.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: 227
Status: proposed
Units: packets

3.4.1.13 reverseDroppedOctetTotalCount

Description:
The total number of octets in packets sent by the biflow destination for this biflow dropped by packet treatment since the Metering Process (re-)initialization for this Observation Point. The number of octets include IP header(s) and IP payload.
Abstract Data Type: unsigned64

Data Type Semantics: totalCounter
ElementId: 228
Status: proposed
Units: octets

3.4.1.14 reverseDroppedPacketTotalCount

Description:
The total number of packets sent by the biflow destination for this biflow dropped by packet treatment since the Metering Process (re-)initialization for this Observation Point.
Abstract Data Type: unsigned64
Data Type Semantics: totalCounter
ElementId: 229
Status: proposed
Units: packets

3.4.1.15 directionDomain

Description:

Defines the semantics of source and destination information elements, and of reverse counters. This information element is intended to be bound to a sourceID or a templateID in an options data record. Defined values include:

0x00: Uniflow. Source and destination are defined as in the packet headers from which the flows were generated. Reverse counters have no defined meaning. This directionDomain is not appropriate for exporting biflows in single data records. This is the present default IPFIX behavior.

0x01: First Packet Biflow. The source of the biflow is defined as the source of the first packet seen within the flow by the Metering Process, and the destination of the biflow is defined as the destination of that first packet. Counters count packets sent by the first packet source, and reverse counters count packets sent by the first packet destination.

0x02: Initiator Biflow. The source of the biflow is defined as the source of the packet initiating the connection as determined by the Metering Process, and the destination of the biflow is defined as the destination of the packet initiating the connection. This is similar to First Packet Biflow, except it gives the Metering Process some latitude to attempt to determine the connection initiator when the initiator is not necessarily the sender of the first packet seen by the Metering Process.

0x03: Listener Biflow. The destination of the biflow is defined as the source of the packet initiating the connection as determined by the Metering Process or the Exporting Process, and the source of the biflow is defined as the destination of the packed initiating the connection. This is similar to Initiator Biflow, except the source and destination are reversed. This is primarily useful for the case when the directionDomain information element is present in each flow data record, and an intermediate process handling multiple Metering Processes determines an Initiator biflow should be reversed.

0x03 - 0x7F: Reserved for future assignment

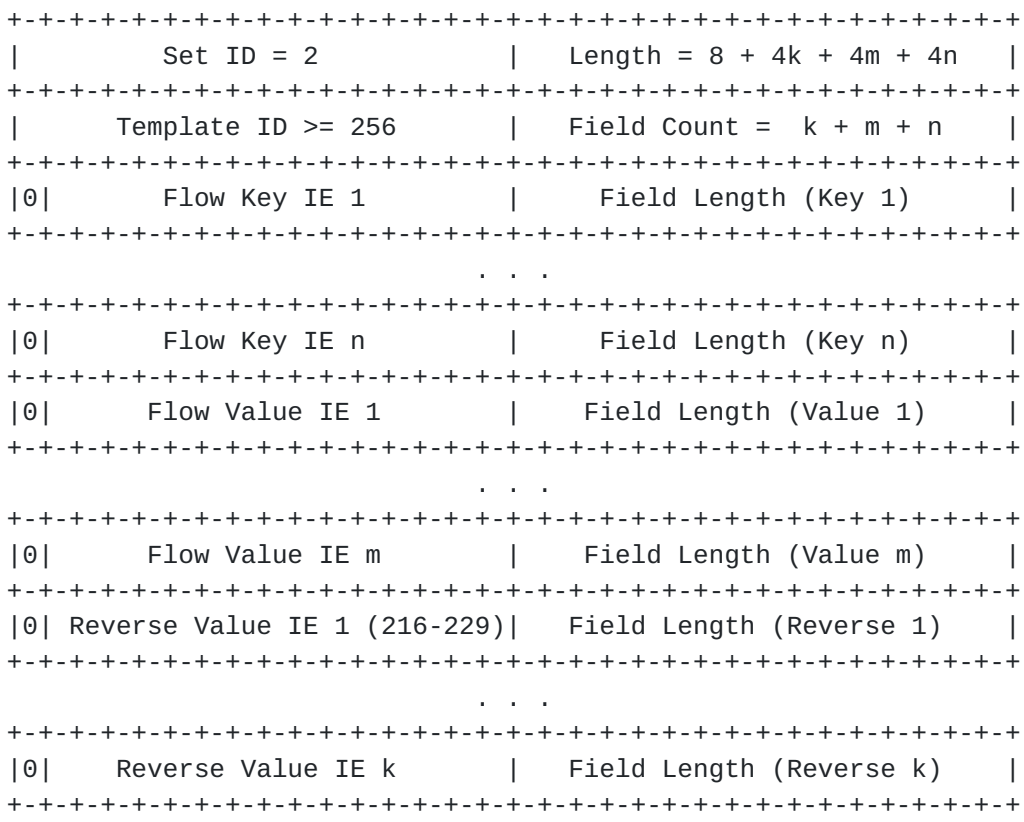
0x80 - 0xFF: Reserved for private or experimental use.

Abstract Data Type: octet

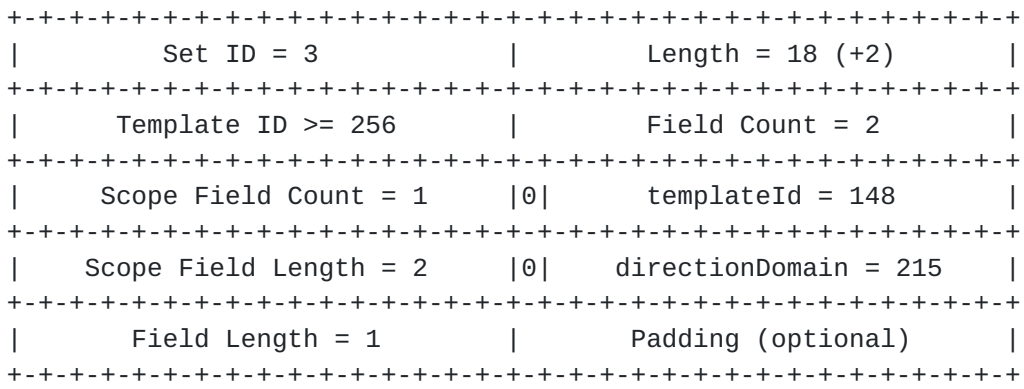
Data Type Semantics: identifier
ElementId: 215
Status: proposed

3.4.2 Example

Each single biflow data record can be described by a template following the pattern below.



The options template below describes an options record that MUST be present to semantically define the reverse counter information. The directionDomain could also be associated with a sourceID, or for the cost of one octet per record, appear in the biflow record itself.



4. IANA Considerations

This documents defines a set of new IPFIX Information Elements that extend those already defined in [IPFIX-INFO].

As specified in [IPFIX-INFO] IANA needs to create a new registry for IPFIX Information Element identifiers. New assignments for IPFIX Information Elements will be administered by IANA, on a First Come First Served basis [RFC2434], subject to Expert Review [RFC2434], i.e. review by one of a group of expert designated by an IETF Operations and Management Area Director. The group of experts must double check the Information Elements definitions with already defined Information Elements for completeness, accuracy, redundancy, and correct naming following the naming conventions in [IPFIX-INFO]. Those experts will initially be drawn from the Working Group Chairs and document editors of the IPFIX and PSAMP Working Groups [IPFIX-INFO].

5. Security Considerations

The same security considerations as for the IPFIX protocol [IPFIX-PROTO] apply.

6. References

6.1 Normative References

[IPFIX-PROTO] B. Claise et Al.: IPFIX Protocol Specification, Internet-draft work in progress <draft-ietf-ipfix-protocol-19.txt>, September 2005

[IPFIX-INFO] J. Quittek, S.Bryant, B.Claise, J. Meyer: Information Model for IP Flow Information Export Internet-draft work in progress <draft-ietf-ipfix-info-11.txt>, September 2005

6.2 Informative References

[RFC2434] Narten, T. and H. Alvestrand: Guidelines for Writing an IANA Considerations Section in RFCs, BCP 26, RFC 2434, October 1998.

7. Acknowledgements

We would like to thank Lutz Mark for his contribution and comments.

8. Author's Addresses

Elisa Boschi
Hitachi Europe SAS
Immeuble Le Theleme
1503 Route des Dolines
06560 Valbonne, France
Phone: +33 4 89874180
Email: elisa.boschi@hitachi-eu.com

Brian H. Trammell
CERT Network Situational Awareness
Software Engineering Institute
4500 Fifth Avenue

Pittsburgh, PA 15213
US
Phone: +1 412 268 9748
Email: bht@cert.org

9. Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org.

10. Copyright Statement

Copyright (C) The Internet Society (2005). This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.

11. Disclaimer

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.