SIP Message Information Export using IPFIX

draft-trammell-ipfix-sip-msg-02

Abstract

This draft defines a set of Information Elements and example Templates for IP Flow Information Export (IPFIX) based on the SIP Common Log Format data model, as well as additional useful SIP Information Elements, to allow IPFIX export of application-layer information about SIP messages.

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 on April 29, 2012.

Copyright Notice

Copyright (c) 2011 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents.
carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction ........................................ 4
2. Base Information Elements for SIP Message Information Export ............. 5
   2.1. sipObservationType .................................. 5
   2.2. sipMethod ........................................ 5
   2.3. sipSequenceNumber .................................. 5
   2.4. sipRequestURI ...................................... 5
   2.5. sipFromURI ......................................... 6
   2.6. sipFromTag ......................................... 6
   2.7. sipToURI ........................................... 6
   2.8. sipToTag ........................................... 6
   2.9. sipCallId .......................................... 6
   2.10. sipResponseStatus .................................. 6
   2.11. sipServerTransaction ................................ 7
   2.12. sipClientTransaction ................................ 7
   2.13. sipMethod subregistry ............................... 7
3. Additional Information Elements for SIP Message Information Export .......... 8
   3.1. sipContactURI ...................................... 8
   3.2. sipRouteURI ......................................... 8
   3.3. sipPaiURI .......................................... 9
   3.4. sipPpiURI ........................................... 9
   3.5. sipPAssocURI ....................................... 9
   3.6. sipPCalledPartyURI .................................. 9
   3.7. sipVia ............................................. 10
   3.8. sipAuthUsername ..................................... 10
   3.9. sipSubscriptionEvent ................................ 10
   3.10. sipSubscriptionState ................................ 10
   3.11. sipExpires ......................................... 10
   3.12. sipPVisitedNetworkID ............................... 11
   3.13. sipPAccessNetworkInfo .............................. 11
   3.14. sipPChargingFunctionAddr ........................... 11
   3.15. sipPChargingVector ................................. 11
4. Recommended Templates for SIP Message Information Export .................. 12
5. Examples .............................................. 13
   5.1. Base Template Export ................................ 13
   5.2. UAC registration .................................... 14
   5.3. Direct Call ......................................... 16
   5.4. Single Downstream Branch Call ........................ 19
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.5</td>
<td>Forked Call</td>
<td>25</td>
</tr>
<tr>
<td>6</td>
<td>Security Considerations</td>
<td>35</td>
</tr>
<tr>
<td>7</td>
<td>IANA Considerations</td>
<td>35</td>
</tr>
<tr>
<td>8</td>
<td>Acknowledgments</td>
<td>35</td>
</tr>
<tr>
<td>9</td>
<td>References</td>
<td>35</td>
</tr>
<tr>
<td>9.1</td>
<td>Normative References</td>
<td>35</td>
</tr>
<tr>
<td>9.2</td>
<td>Informative References</td>
<td>36</td>
</tr>
<tr>
<td>Appendix A</td>
<td>Definition of Base SIP Message Information</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>Elements in IANA XML Registry format</td>
<td></td>
</tr>
<tr>
<td>Appendix B</td>
<td>Definition of sipMethod registry in IANA XML</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>Registry format</td>
<td></td>
</tr>
<tr>
<td>Appendix C</td>
<td>Definition of Additional SIP Message Information</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>Elements in IANA XML Registry format</td>
<td></td>
</tr>
<tr>
<td>Appendix D</td>
<td>Example messages in base64</td>
<td>47</td>
</tr>
<tr>
<td>Authors’ Addresses</td>
<td></td>
<td>50</td>
</tr>
</tbody>
</table>
1. Introduction

IPFIX [RFC5101] provides a standardized means of exporting flow information from IPFIX exporters to IPFIX collectors. This allows collectors to analyze flows from one or more sources for numerous uses, such as traffic patterns/trends, anomalies, failures, attacks, and much more. IPFIX supports exporting data in near real-time, in a secure manner, over multiple transports; as well as in local storage with a defined file format. The core IPFIX information model is maintained by IANA as a registry of Information Elements at http://www.iana.org/assignments/ipfix/. In addition to these, which cover many network measurement and management applications, enterprise-specific Information Elements may be defined, scoped to an SMI private enterprise number, for vendor-proprietary Information Elements.

Session Initiation Protocol (SIP), defined by [RFC3261] and its extensions, is used by many devices to perform a rendezvous service, initiate and manage real-time communication sessions, install and monitor state information, and more. In many deployments, SIP messages cross multiple systems managed by the same administrative entity, and thus providing a means of exporting and collecting SIP message information from such systems using a standard protocol is highly desirable.

This document defines a set of IPFIX Information Elements to enable SIP devices, such as user agents and proxies, to export SIP message information to IPFIX collectors using the IPFIX protocol. The purpose of doing so is to enable collectors to analyze the SIP "traffic", for similar purposes as those for any other IPFIX flows. Defining IANA-registered (i.e., well-known) IPFIX IE fields enables IPFIX records of SIP message information to be generated and consumed by different vendors. Within the context of this document’s IPFIX IE fields, a single SIP message is a complete IPFIX Flow as defined in [RFC5101]

The SIPCLF Working Group has defined a data model [I-D.ietf-sipclf-problem-statement] for logging information about SIP messages to ASCII-based SIPCLF files. While useful for on-box storage and analysis with ASCII-based tools, SIPCLF does not provide a means of exporting such information, nor is that its goal. This document borrows the data model from SIPCLF and represents these in IPFIX Information Elements. It additionally provides examples for IPFIX representation of the example SIP Messages provided in the SIPCLF problem statement.
2. Base Information Elements for SIP Message Information Export

The following Information Elements represent SIP-specific mandatory fields defined in [I-D.ietf-sipclf-problem-statement], many themselves taken from [RFC3261]. Together with Information Elements already available in the IPFIX IANA Information Elements registry, these can be used to export information about SIP Messages.

2.1. sipObservationType

Description: Denotes whether the entry was corresponds to a SIP message received, sent, or merely seen by a passive observer, as follows:
0: unknown: The Metering Process does not specify the observation type.
1: receiver: The Metering Process is, or is co-located with, the receiver of the SIP message.
2: sender: The Metering Process is, or is co-located with, the sender of the SIP message.
3: passive: The Metering Process passively observed the SIP message.

Data Type: unsigned8
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 419

2.2. sipMethod

Description: The SIP method from the CSeq header, encoded as per the IPFIX sipMethod subregistry.

Data Type: unsigned8
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 402

2.3. sipSequenceNumber

Description: The sequence number from the CSeq header.

Data Type: unsigned32
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 409

2.4. sipRequestURI
2.5. sipFromURI

Description: The URI from the SIP From: header
Data Type: string
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 403

2.6. sipFromTag

Description: The Tag parameter value from the SIP From: header
Data Type: string
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 404

2.7. sipToURI

Description: The URI from the SIP To: header
Data Type: string
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 405

2.8. sipToTag

Description: The Tag parameter value from the SIP To: header
Data Type: string
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 406

2.9. sipCallId

Description: The value of the SIP Call-ID: header
Data Type: string
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 408

2.10. sipResponseStatus
Description: The SIP Response code. The presence of this Information Element in a SIP Message record marks it as describing a SIP response; if absent, the record describes a SIP request.
Data Type: unsigned16
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 412

2.11. sipServerTransaction

Description: The transaction identifier associated with the server transaction.
Data Type: string
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 413

2.12. sipClientTransaction

Description: The transaction identifier associated with the client transaction.
Data Type: string
Data Type Semantics: identifier
PEN (provisional): 35566 (trammell.ch)
ElementId (provisional): 414

2.13. sipMethod subregistry

The sipMethod subregistry assigns a number to encode each of the SIP methods encoded in the Methods and Response Codes registry at http://www.iana.org/assignments/sip-parameters in a 16-bit integer Information Element. These numbers are assigned from 1 in alphabetical order for the Methods defined as of the publication time of this document; subsequent Methods added to the Methods and Response Codes registry will be added to the IPFIX sipMethod subregistry at such time they are added to the Methods and Response Codes registry, using the lowest available unassigned number at the time of addition.
### 3. Additional Information Elements for SIP Message Information Export

[TODO frontmatter]

#### 3.1. sipContactURI

Description: The addr-spec URI, including any URI parameters, of the first/top-most SIP Contact header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string  
Data Type Semantics:  
PEN (provisional): 35566  
ElementId (provisional): 415

#### 3.2. sipRouteURI

Description: The addr-spec URI, including any URI parameters, of the first/top-most SIP Route header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string  
Data Type Semantics:  
PEN (provisional): 35566
ElementId (provisional): 416

3.3. **sipPaiURI**

Description: The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Asserted-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 417

3.4. **sipPpiURI**

Description: The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Preferred-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 418

3.5. **sipPAssocURI**

Description: The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Associated-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 430

3.6. **sipPCalledPartyURI**

Description: The addr-spec URI, including any URI parameters, of the SIP P-Called-Party-ID header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.

Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 420
3.7. sipVia

Description: The value of the first/top-most Via header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 421

3.8. sipAuthUsername

Description: The value of the username field of the first/top-most Authorization header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 422

3.9. sipSubscriptionEvent

Description: The value of the Event header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 423

3.10. sipSubscriptionState

Description: The value of the Subscription-State header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 424

3.11. sipExpires

Description: The numeric value of the expires parameter of the first/top-most Contact header of a REGISTER request or response, or Subscription-State header of a SUBSCRIBE or NOTIFY request or response, or the Expires header if the expires parameter does not exist, as received by the metering process.
Data Type: unsigned32
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 425

3.12. sipPVisitedNetworkID

Description: The value of the first/top-most P-Visited-Network-ID header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 426

3.13. sipPAccessNetworkInfo

Description: The value of the P-Access-Network-Info header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 427

3.14. sipPChargingFunctionAddr

Description: The value of the first/top-most P-Charging-Function-Addresses header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 428

3.15. sipPChargingVector

Description: The value of the P-Charging-Vector header as a UTF-8 string, escaped according to SIP rules as received by the metering process.
Data Type: string
Data Type Semantics:
PEN (provisional): 35566
ElementId (provisional): 429
4. Recommended Templates for SIP Message Information Export

The SIPCLF data model represents SIP requests and SIP responses with separate records. The following Templates are defined as recommended base Templates for records describing requests and responses. Optional Information Elements MAY be added to them, and the IPv4 addresses within these Templates MUST be replaced with IPv6 addresses for logging IPv6 transport of SIP messages. A sipServerTransaction Information Element SHOULD be added for all messages logged by a User Agent Server, and a sipClientTransaction Information Element SHOULD be added for all messages logged by a User Agent Client. These templates follow the recommended fields for request and response logging in [I-D.ietf-sipclf-problem-statement], and are defined using the representation in section 9 of [I-D.trammell-ipfix-ie-doctors].

observationTimeMilliseconds(323)[8]
sipSequenceNumber(35566/409)[4]
sourceIPv4Address(8)[4]
destinationIPv4Address(12)[4]
sourceTransportPort(7)[2]
destinationTransportPort(11)[2]
protocolIdentifier(4)[1]
sipMethod(35566/402)[1]
sipObservationType(35566/419)[1]
sipRequestURI(35566/403)[v]
sipToURI(35566/406)[v]
sipToTag(35566/407)[v]
sipFromURI(35566/404)[v]
sipFromTag(35566/405)[v]
sipCallId(35566/408)[v]

Figure 1: Base Request Template (IPv4)
observationTimeMilliseconds(323)[8]
sipSequenceNumber(35566/409)[4]
sourceIPv4Address(8)[4]
destinationIPv4Address(12)[4]
sourceTransportPort(7)[2]
destinationTransportPort(11)[2]
protocolIdentifier(4)[1]
sipMethod(35566/402)[1]
sipObservationType(35566/419)[1]
sipResponseStatus(35566/412)[2]
sipToURI(35566/406)[v]
sipToTag(35566/407)[v]
sipFromURI(35566/404)[v]
sipFromTag(35566/405)[v]
sipCallId(35566/408)[v]

Figure 2: Base Response Template (IPv4)

Note that the Information Elements in these templates are ordered to place the fixed-length elements before the variable-length ones, which speeds random access to fixed-length elements. However, since element order within a record is unimportant in IPFIX, any ordering of the mandatory Information Elements within a record MUST be accepted by a Collecting Process as a valid SIP request or response record for that record type.

The record type is determined by the presence of the sipResponseStatus field. If present in the Template, the Template describes a response record. If absent, it describes a request record.

5. Examples

This section presents several views of an example SIP messages exported using the IPFIX templates described in this document. We present both binary and textual forms. The tools to generate this section are based upon the open-source ripfix [ripfix] implementation of IPFIX, maintained by one of the authors of this draft.

Here we show the IPFIX Messages generated by the situations in sections 9.1 through 9.4 of [I-D.ietf-sipclf-problem-statement].

5.1. Base Template Export

Before exporting any Request or Response records, the Templates describing them must be exported. In this example, the templates These Templates are derived from the base Templates as shown in
Figure 1 and Figure 2, with the sipClientTransaction and sipServerTransaction Information Elements appended. We use two templates here, one each for request and response for IPv4.

Exporting these Templates results in the following IPFIX message, illustrated as an annotated hexdump in Figure 3.

```
0000: 00 0a 00 fc 4c c0 2a a2 00 00 00 00 00 30 39 ....L.*.......09
     [ IPFIX message header, length 252 ]
0010: 00 02 00 ec                                      ....
     [ Template set (ID 2) header, length 236 ]
0014:             01 01 00 11 01 43 00 08 81 99 00 04      .....C......
0020: 81 a3 00 01 00 00 8a ee 81 93 ff ff 00 00 8a ee  ...........
0030: 81 96 ff ff 00 00 8a ee 81 97 ff ff 00 00 8a ee  ...........
0040: 81 94 ff ff 00 00 8a ee 81 95 ff ff 00 00 8a ee  ...........
0050: 81 98 ff ff 00 00 8a ee 81 9e ff ff 00 00 8a ee  ...........
0060: 81 9d ff ff 00 00 8a ee              ............
     [ Template 257, 17 elements (v4 request) ]
0088:                         01 02 00 11 01 43 00 08          .....C.. 0090: 81 99 00 04 00 00 8a ee 00 08 00 04 00 0c 00 04 00 02 00 04 00 01 81 92 00 01 81 93 ff ff 00 00 8a ee 81 94 ff ff 00 00 8a ee 81 95 ff ff 00 00 8a ee 81 96 ff ff 00 00 8a ee 81 97 ff ff 00 00 8a ee 81 98 ff ff 00 00 8a ee 81 9e ff ff 00 00 8a ee 81 9d ff ff 00 00 8a ee              ............
     [ Template 258, 17 elements (v4 response) ]
```

Figure 3: Base template message export

5.2. UAC registration

Having exported templates, now we create a simple IPFIX Message representing a UAC registration as seen from the UAC, corresponding to example 9.1 in [I-D.ietf-sipclf-problem-statement]. This message contains two records, including the UAS registration request, and the response received. This is shown in the annotated hexdump in Figure 4.
While this demonstrates the binary nature of the SIPCLF-IPFIX format, and shows the content framing for this message, it is not readable for illustration purposes. In Figure 5, we run the message through the ripcollect tool provided with ripfix to provide a more human-readable view. Note that the sipMethod and sipObservationType are encoded according to the registries in Section 2.
5.3. Direct Call

This example demonstrates the export of a direct call from Alice to Bob, as seen by Bob’s agent, corresponding to example 9.2 in [I-D.ietf-sipclf-problem-statement]. Here we have four records: an INVITE received from Alice, a 180 Ringing sent back followed by a 200 OK, and an ACK received from Alice. This is shown in the ripfix dump in Figure 6 and the hexdump in Figure 7. In the hexdump, message
headers, set headers, and data records are separated by ‘|’ characters for compactness. Note here that each record has its own data set to support high-speed seeking to a specific record, even when two messages using the same template are adjacent in the message.

===== message 12345/0 @2010-10-21 13:11:43 UTC (#2) =====
--- record 12345/257 (#1)---
observationTimeMilliseconds => 2010-06-07 17:12:23 UTC
sipSequenceNumber => 32
sourceIPv4Address => 198.51.100.1
destinationIPv4Address => 203.0.113.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipRequestURI => sip:bob@bob1.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => 76yhh
sipCallId => f82-d4-f7@example.com
sipClientTransaction => c-1-xt6
sipServerTransaction =>
--- record 12345/258 (#2)---
observationTimeMilliseconds => 2010-06-07 17:12:25 UTC
sipSequenceNumber => 32
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 180
sipToURI => sip:bob@example.net
sipToTag => b-in6-iu
sipFromURI => sip:alice@example.com
sipFromTag => 76yhh
sipCallId => f82-d4-f7@example.com
sipClientTransaction => c-1-xt6
sipServerTransaction =>
--- record 12345/258 (#3)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 32
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b-in6-iu
sipFromURI => sip:alice@example.com
sipFromTag => 76yhh
sipCallId => f82-d4-f7@example.com
sipClientTransaction => c-1-xt6
sipServerTransaction =>

--- record 12345/257 (#4)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 32
sourceIPv4Address => 198.51.100.1
destinationIPv4Address => 203.0.113.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 1
sipObservationType => 2
sipRequestURI => sip:bob@bob1.example.net
sipToURI => sip:bob@example.net
sipToTag => b-in6-iu
sipFromURI => sip:alice@example.com
sipFromTag => 76yhh
sipCallId => f82-d4-f7@example.com
sipClientTransaction => c-1-xt6
sipServerTransaction =>

Figure 6: Message containing four records for a simple call (ripfix dump)
Figure 7: Message containing four records for a simple call (hexdump)

5.4. Single Downstream Branch Call

The example in Figure 8 and Figure 9 demonstrates the export of a call with a downstream branch to Bob, as seen by the proxy which the call traverses, corresponding to example 9.3 in [I-D.ietf-sipclf-problem-statement]. See this example in the problem statement for more details.

Figure 7: Message containing four records for a simple call (hexdump)
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.1
destinationIPv4Address => 198.51.100.10
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObserveObservationType => 1
sipRequestURI => sip:bob@example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction =>
sipServerTransaction => s-x-tr
--- record 12345/258 (#2)---
observationTimeMilliseconds => 2010-06-07 17:12:24 UTC
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.10
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObserveObservationType => 2
sipResponseStatus => 100
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction =>
sipServerTransaction => s-x-tr
--- record 12345/257 (#3)---
observationTimeMilliseconds => 2010-06-07 17:12:24 UTC
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.10
destinationIPv4Address => 203.0.113.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObserveObservationType => 2
sipRequestURI => sip:bob@bob1.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com

sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/258 (#4)---
observationTimeMilliseconds => 2010-06-07 17:12:25 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 198.51.100.10
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 100
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/258 (#5)---
observationTimeMilliseconds => 2010-06-07 17:12:25 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 198.51.100.10
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 180
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/258 (#6)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.10
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 180
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/258 (#7)---
observationTimeMilliseconds => 2010-06-07 17:12:27 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 198.51.100.10
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/258 (#8)---
observationTimeMilliseconds => 2010-06-07 17:12:27 UTC
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.10
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => al-1
sipCallId => tr-87h@example.com
sipClientTransaction => c-x-tr
sipServerTransaction => s-x-tr
--- record 12345/257 (#9)---
observationTimeMilliseconds => 2010-06-07 17:12:29 UTC
Figure 8: Message containing ten records for a downstream branch call

(ripfix dump)
Figure 9: Message containing ten log entries for a downstream branch call (hexdump)

5.5. Forked Call

The example in Figure 11 and Figure 12 demonstrates the export of forked call to Bob, as seen by one of Bob’s instances which forks the call traverses, corresponding to example 9.4 in [I-D.ietf-sipclf-problem-statement]. See this example for more details. Note that, since Bob’s first instance is multihomed IPv4-IPv6, this example requires additional templates: request and response templates for IPv4 to IPv6 and back, these are shown in Figure 10.
Figure 10: Message containing templates for IPv4 to IPv6 requests and responses, and vice versa

===== message 12345/0  @2010-10-21 13:13:01 UTC (#3) =====
--- record 12345/257 (#1)---
observationTimeMilliseconds => 2010-06-07 17:12:23 UTC
sipSequenceNumber => 43
sourceIPv4Address => 198.51.100.1
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipRequestURI => sip:bob@example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction =>
sipServerTransaction => s-1-tr
--- record 12345/258 (#2)---
observationTimeMilliseconds => 2010-06-07 17:12:24 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 100
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction =>
sipServerTransaction => s-1-tr
--- record 12345/257 (#3)---
observationTimeMilliseconds => 2010-06-07 17:12:24 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 203.0.113.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipRequestURI => sip:bob@bob1.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction =>
sipServerTransaction => s-1-tr
--- record 12345/261 (#4)---
observationTimeMilliseconds => 2010-06-07 17:12:25 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv6Address => 2001:db8::9
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipRequestURI => sip:bob@bob2.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr
--- record 12345/258 (#5)---
observationTimeMilliseconds => 2010-06-07 17:12:25 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 100
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-1-tr
sipServerTransaction => s-1-tr
--- record 12345/264 (#6)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 43
sourceIPv6Address => 2001:db8::9
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 100
sipToURI => sip:bob@example.net
sipToTag => b2-2
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr
--- record 12345/264 (#7)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 43
sourceIPv6Address => 2001:db8::9
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 180
sipToURI => sip:bob@example.net
sipToTag => b2-2
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr
--- record 12345/258 (#8)---
observationTimeMilliseconds => 2010-06-07 17:12:26 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 180
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr
--- record 12345/258 (#9)---
observationTimeMilliseconds => 2010-06-07 17:12:27 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 180

sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-1-tr
sipServerTransaction => s-1-tr
--- record 12345/258 (#10)---
observationTimeMilliseconds => 2010-06-07 17:12:27 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.1
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-1-tr
sipServerTransaction => s-1-tr
--- record 12345/258 (#11)---
observationTimeMilliseconds => 2010-06-07 17:12:28 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 198.51.100.1
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-1-tr
sipServerTransaction => s-1-tr
--- record 12345/261 (#12)---
observationTimeMilliseconds => 2010-06-07 17:12:28 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv4Address => 2001:db8::9
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 2
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag => b1-1
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-1-tr
sipServerTransaction => s-1-tr
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 3
sipObservationType => 2
sipRequestURI => sip:bob@bob2.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr

--- record 12345/264 (#13)---
observationTimeMilliseconds => 2010-06-07 17:12:28 UTC
sipSequenceNumber => 43
sourceIPv6Address => 2001:db8::9
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 5
sipObservationType => 1
sipResponseStatus => 487
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr

--- record 12345/261 (#14)---
observationTimeMilliseconds => 2010-06-07 17:12:29 UTC
sipSequenceNumber => 43
sourceIPv4Address => 203.0.113.200
destinationIPv6Address => 2001:db8::9
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 1
sipObservationType => 2
sipRequestURI => sip:bob@bob2.example.net
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
--- record 12345/264 (#15)---
observationTimeMilliseconds => 2010-06-07 17:12:30 UTC
sipSequenceNumber => 43
sourceIPv6Address => 2001:db8::9
destinationIPv4Address => 203.0.113.200
sourceTransportPort => 5060
destinationTransportPort => 5060
protocolIdentifier => 17
sipMethod => 3
sipObservationType => 1
sipResponseStatus => 200
sipToURI => sip:bob@example.net
sipToTag =>
sipFromURI => sip:alice@example.com
sipFromTag => a1-1
sipCallId => tr-88h@example.com
sipClientTransaction => c-2-tr
sipServerTransaction => s-1-tr

Figure 11: Message containing fifteen records for a forked call

0000: 00 0a 07 8c 4c 02 2d 9b 00 00 00 00 00 00 30 39| ....L..........9
0010: 01 01 00 7e|00 00 01 29 13 66 13 93 00 00 0b |......).f.....+
0020: c6 33 64 0f cb 00 13 87 2a 00 00 00 2b cb 00  .3d...q........
0030: 71 c8 c6 33 64 01 13 c4 13 c4 11 05 01 13 q....3d......
0040: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
0050: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa
0060: 78 61 6d 70 6c 65 2e 63 6f 6d 04 61 31 2d 31 12  mple.net..sip:a1-
0070: 74 72 2d 38 38 68 40 65 78 61 6d 70 6c 65 2e 63  tr-88h@example.c
0080: 6f 6d 00 06 73 2a 73 2d 31 2d 74 72 |01 05 00 89|00 00  am..s-1-tr......
0090: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
00a0: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa
00b0: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
00c0: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa
00d0: 78 61 6d 70 6c 65 2e 63 6f 6d 04 61 31 2d 31 12  mple.net..sip:a1-
00e0: 74 72 2d 38 38 68 40 65 78 61 6d 70 6c 65 2e 63  tr-88h@example.c
00f0: 6f 6d 00 06 73 2a 73 2d 31 2d 74 72 |01 05 00 89|00 00  am..s-1-tr......
0100: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
0110: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa
0120: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
0130: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa
0140: 78 61 6d 70 6c 65 2e 63 6f 6d 04 61 31 2d 31 12  mple.net..sip:a1-
0150: 74 72 2d 38 38 68 40 65 78 61 6d 70 6c 65 2e 63  tr-88h@example.c
0160: 6f 6d 00 06 73 2a 73 2d 31 2d 74 72 |01 05 00 89|00 00  am..s-1-tr......
0170: 73 69 70 3a 62 6f 62 40 65 78 61 6d 70 6c 65 2e  sip:bob@example.
0180: 6e 65 74 00 15 73 69 70 3a 61 6c 69 63 65 40 65  et.sip:alice@exa

6. Security Considerations

[TODO]

7. IANA Considerations

This document defines the sipMethod subregistry for the IANA IPFIX Information Element registry at http://www.iana.org/assignments/ipfix for the values taken by the sipMethod Information Element. The initial content of this subregistry is specified in Section 2.13. Entries may be added to this subregistry subject to the same Standards Action [RFC5226] that adds new Methods to the Methods and Response Codes registry at http://www.iana.org/assignments/sip-parameters.

At such time as this document is prepared for publication as an RFC, the Information Elements defined herein will be defined for inclusion in the IANA IPFIX Information Element registry at http://www.iana.org/assignments/ipfix. Until such time, the Information Elements within this document are defined within Private Enterprise Number 35566, belonging to one of the authors.

8. Acknowledgments

Thanks to Cullen Jennings for his provided insightful discussions, specific comments and much needed corrections, and to Nico d’Heureuse for his help with the RFC 3665 examples.

9. References

9.1. Normative References

[I-D.ietf-sipclf-problem-statement]

9.2. Informative References

[I-D.kaplan-dispatch-session-id]
Kaplan, H., "A Session Identifier for the Session Initiation Protocol (SIP)",
draft-kaplan-dispatch-session-id-03 (work in progress),
March 2011.

[I-D.trammell-ipfix-ie-doctors]
Trammell, B. and B. Claise, "Guidelines for Authors and Reviewers of IPFIX Information Elements",
draft-trammell-ipfix-ie-doctors-02 (work in progress),
June 2011.

[RFC2976] Donovan, S., "The SIP INFO Method", RFC 2976,
October 2000.

Schooler, "SIP: Session Initiation Protocol", RFC 3261,
June 2002.

[RFC3262] Rosenberg, J. and H. Schulzrinne, "Reliability of Provisional Responses in Session Initiation Protocol (SIP)",
RFC 3262, June 2002.


Appendix A. Definition of Base SIP Message Information Elements in IANA XML Registry format

[EDITOR’S NOTE: frontmatter]

<registry xmlns="http://www.iana.org/assignments" id="ipfix">
  <registry id="ipfix-information-element-definitions">
    <record>
      <name>sipObservationType</name>
      <dataType>unsigned8</dataType>
      <dataTypeSemantics>identifier</dataTypeSemantics>
      <enterpriseId>35566</enterpriseId>
      <elementId>419</elementId>
      <status>current</status>
      <description>
        <paragraph>
          Denotes whether the entry was corresponds to a SIP message received, sent, or merely seen by a passive observer, as follows:
        </paragraph>
        <paragraph>0: unknown: The Metering Process does not specify the observation type.</paragraph>
        <paragraph>1: receiver: The Metering Process is, or is co-located with, the receiver of the SIP message.</paragraph>
        <paragraph>2: sender: The Metering Process is, or is co-located with, the sender of the SIP message.</paragraph>
    </record>
  </registry>
</registry>
3: passive: The Metering Process passively observed the SIP message.

The SIP method from the CSeq header, encoded as per the IPFIX sipMethod subregistry.

The sequence number from the CSeq header.

The SIP Request URI, including any parameters, as a UTF-8 string, escaped according to SIP rules as received by the metering process.
<record>
  <name>sipFromTag</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>405</elementId>
  <status>current</status>
  <description>
    <paragraph>The Tag parameter value from the SIP From: header</paragraph>
  </description>
</record>

<record>
  <name>sipToURI</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>406</elementId>
  <status>current</status>
  <description>
    <paragraph>The URI from the SIP To: header</paragraph>
  </description>
</record>

<record>
  <name>sipToTag</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>407</elementId>
  <status>current</status>
  <description>
    <paragraph>The Tag parameter value from the SIP To: header</paragraph>
  </description>
</record>

<record>
  <name>sipCallId</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>408</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the SIP Call-ID: header</paragraph>
  </description>
</record>

<record>
  <name>sipResponseStatus</name>
  <dataType>unsigned16</dataType>
  <description>
    <paragraph>The status code from the SIP response</paragraph>
  </description>
</record>
SIP Message Information Element definitions

Appendix B. Definition of sipMethod registry in IANA XML Registry format

[EDITOR’S NOTE: frontmatter]
<title>IPFIX sipMethod</title>
<registration_rule>Expert Review</registration_rule>
<record>
  <value>0</value>
  <description>Unknown</description>
  <comments>The Metering Process did not recognize the SIP method.</comments>
</record>
<record>
  <value>1</value>
  <description>ACK</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>2</value>
  <description>BYE</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>3</value>
  <description>CANCEL</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>4</value>
  <description>INFO</description>
  <comments/>
  <xref type="rfc" data="rfc6086"/>
</record>
<record>
  <value>5</value>
  <description>INVITE</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>6</value>
  <description>MESSAGE</description>
  <comments/>
  <xref type="rfc" data="rfc3428"/>
</record>
<record>
  <value>7</value>
  <description>NOTIFY</description>
  <comments/>
</record>
<record>
  <value>8</value>
  <description>OPTIONS</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>9</value>
  <description>PRACK</description>
  <comments/>
  <xref type="rfc" data="rfc3262"/>
</record>
<record>
  <value>10</value>
  <description>PUBLISH</description>
  <comments/>
  <xref type="rfc" data="rfc3903"/>
</record>
<record>
  <value>11</value>
  <description>REFER</description>
  <comments/>
  <xref type="rfc" data="rfc3515"/>
</record>
<record>
  <value>12</value>
  <description>REGISTER</description>
  <comments/>
  <xref type="rfc" data="rfc3261"/>
</record>
<record>
  <value>13</value>
  <description>SUBSCRIBE</description>
  <comments/>
  <xref type="rfc" data="rfc3265"/>
</record>
<record>
  <value>14</value>
  <description>UPDATE</description>
  <comments/>
  <xref type="rfc" data="rfc3311"/>
</record>
<record>
  <value>15-65535</value>
  <description>Unassigned</description>
  <comments/>
</record>
Appendix C. Definition of Additional SIP Message Information Elements in IANA XML Registry format

[EDITOR’S NOTE: frontmatter]

<?xml version="1.0" encoding="UTF-8"?>

<registry xmlns="http://www.iana.org/assignments" id="ipfix">
  <registry id="ipfix-information-element-definitions">
    <record>
      <name>sipContactURI</name>
      <dataType>string</dataType>
      <enterpriseId>35566</enterpriseId>
      <elementId>415</elementId>
      <status>current</status>
      <description>
        <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP Contact header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
      </description>
    </record>
    <record>
      <name>sipRouteURI</name>
      <dataType>string</dataType>
      <enterpriseId>35566</enterpriseId>
      <elementId>416</elementId>
      <status>current</status>
      <description>
        <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP Route header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
      </description>
    </record>
    <record>
      <name>sipPaiURI</name>
      <dataType>string</dataType>
      <enterpriseId>35566</enterpriseId>
      <elementId>417</elementId>
      <status>current</status>
      <description>
        <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP Route header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
      </description>
    </record>
  </registry>
</registry>
<status>current</status>
<description>
  <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Asserted-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
</description>
</record>
<record>
  <name>sipPpiURI</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>418</elementId>
  <status>current</status>
  <description>
    <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Preferred-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>
<record>
  <name>sipPAssocURI</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>430</elementId>
  <status>current</status>
  <description>
    <paragraph>The addr-spec URI, including any URI parameters, of the first/top-most SIP P-Associated-Identity header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>
<record>
  <name>sipPCalledPartyURI</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>420</elementId>
  <status>current</status>
  <description>
    <paragraph>The addr-spec URI, including any URI parameters, of the SIP P-Called-Party-ID header, as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>
<record>
  <name>sipVia</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>421</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the first/top-most Via header
    as a UTF-8 string, escaped according to SIP rules as
    received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipAuthUsername</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>422</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the username field
    of the first/top-most Authorization header
    as a UTF-8 string, escaped according to SIP rules as
    received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipSubscriptionEvent</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>423</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the Event header
    as a UTF-8 string, escaped according to SIP rules as
    received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipSubscriptionState</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>424</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the Subscription-State header
    as a UTF-8 string, escaped according to SIP rules as
    received by the metering process.</paragraph>
  </description>
</record>
<record>
  <name>sipExpires</name>
  <dataType>unsigned32</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>425</elementId>
  <status>current</status>
  <description>
    <paragraph>The numeric value of the expires parameter of the first/top-most Contact header of a REGISTER request or response, or Subscription-State header of a SUBSCRIBE or NOTIFY request or response, or the Expires header if the expires parameter does not exist, as received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipPVisitedNetworkID</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>426</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the first/top-most P-Visited-Network-ID header as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipPAccessNetworkInfo</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>427</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the P-Access-Network-Info header as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>

<record>
  <name>sipPChargingFunctionAddr</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>428</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the first/top-most P-Charging-Function-Addresses header</paragraph>
  </description>
</record>
as a UTF-8 string, escaped according to SIP rules as received by the metering process.

</description>
</record>
<record>
  <name>sipPChargingVector</name>
  <dataType>string</dataType>
  <enterpriseId>35566</enterpriseId>
  <elementId>429</elementId>
  <status>current</status>
  <description>
    <paragraph>The value of the P-Charging-Vector header as a UTF-8 string, escaped according to SIP rules as received by the metering process.</paragraph>
  </description>
</record>

Additional SIP Message Information Element definitions

Appendix D. Example messages in base64

This section contains the example messages from this revision of this draft in base64 encoding, for ease of processing by automated tools.

The base templates are in this message:

AAoA/EzALZsAAAAAAAAwQACA0wBAQARAUMACIGZAAQAAIrAAgABAAMAQA
BwACAAAsAAgEAEAGBkgABAACK7oGjAAEAIrZP/AAiu6B1v/AAACK7oGX
/8AIrZ2T/wAAiu6B1f/AAACK7oGY/8AAIrZ77/wAAiu6Bnf/AAACK7gECABEBQwAig2kABAAiu4ACAAMEAawABAAHAAICwACAAQAYGSAAEAAIr
gaMAQAIAiu6BnAACACK7oGW/8AAIrZ2f/wAAiu6BPl/AAACK7oGV/8A
AIrZj/wAAiu6Bnv/AAACK7oGd/8AAIr

The extended 4to6 and 6to4 templates are in this message:

AAoB5EzALZsAAAAAAAAwQACaGBQQAAUMACIgZAAQAAIrAAgABAAMAQA
BwACAAAsAAgEAEAGBkgABAACK7oGjAAEAIrZP/AAiu6B1v/AAACK7oGX
/8AIrZ2T/wAAiu6B1f/AAACK7oGY/8AAIrZ77/wAAiu6Bnf/AAACK7gEGABEBQwAig2kABAAiu4ACAAMEAABwAEAAHAAICwACAAQAYGSAAEAAIr
gMAQAIAiu6BnAACACK7oGW/8AAIrZ2f/wAAiu6BPl/AAACK7oGV/8A
AIrZj/wAAiu6Bnv/AAACK7oGd/8AAIrzuQcAEQFDAAiBmQAAEAACK7gAb
ABAADAAAEMcAgALAAIAABBgZlAQAAiu6BowABAACK7oGT/8AAIrZgb/
/AAiu6B1l/AAACK7oGU/8AAIrZ2X/wAAiu6Bon/AAACK7oGe/8AAIr uG3/AAiu4BAAA邬ACIGZAAQAAIrAAgABAAMAQA
kgABAACK7oGjAAEAAIrZwAAgAAiu6B1v/AAACK7oGX/8AAIrZ2T/wAA
iu6B1f/
The UAC registration in Section 5.2 is in this message:

```
AAoA2EzAo88AAAAAAAwOQEBAGsAAAEpE2YTkwAAAAHGM2QBxjNkChPee8QR
DAIPc21w0mV4YW1wbGUuY29tAAAACv21wOnFsawNLQGV4YW1wbGUuY29tBTc2
ewH0FWy4MS1kNC1mNkB1eGFtCx11mNvbQzjLXRyLTEAQAIAqXQAAAskTzUh
AAAACyZzArGM2QBEGQTxBEMACQDIAAAACv21wOnFsawNLQGV4YW1wbGUuY29t
BTc2ewH0FWy4MS1kNC1mNkB1eGFtCx11mNvbQzjLXRyLTEA
```

The direct call in Section 5.3 is in this message:

```
AAoCGkzA88AAAAAAAwOQEBAIgAAAEpE2YTkwAAAAHCDGM2QBwxBARPEE8QR
BQ9Yc121w0mJyYkBi2lXmV4YW1wbGUubmV0E3NpcDpi2J2XhhbXBsZS5u
ZXQFAPnxpdphbgIjJ2UBeGFtCx11mNvbQU3Nn1oaABVmODUtDQqtjdAZXhh
bXBsZS5jzb2OHy0xLOXh0NGABg5AAABKRNKgoKAAAAAgywBxAcYzZAEhBTpE
EQUBAQOC121wOmJyYkBFtCx11m5ldAh1LWU1NipRdVzAXA6WxpyY2V2A
ZXhhbXBsZS5jzb2OFNzZsaAaGzYjgjLWQLWY3QGV4YW1wbGUuY29tBTc2
dDYAAIAeOeAAASKTzhzAAAAAIMAcOSQHMQ2BE8QTxBEFAQDIE3NpcDpi2J2
ZXhhbXBsZS5uZXQYI1pjb1yTaUXV21wOnFsawNLQGV4YW1wbGUuY29tBTc2
eW0FWy4M1kNC1mN0BFtCx11mNvbQdjtLTEhHQAABEAJAAAAEpE2Yd
```

The downstream branch call in Section 5.4 is in this message:

```
AAoE4UzAPEoAAAAAAAwOQEBAl4AAAAApE2YTkwAAAACvE2YW1wbGUubmV0E3NpcDpi2J2Xhhb
ZX6yWxpyY2V2AXXhhbXBsZS5jbb20EYWwteMRJ0ci04N2hAZXhhbXBsZS5jbb20A
BnMteC10cgECAGwAAAAEpE2YW1wQUAACvGM2QKxjJKnAPEE8QRQLAZBNzAA
Ym9iQGV4YW1wbGUubmA0BvzaAXA6WxpyY2VAZXhhbXBsZS5jbb20EYWwteMRJ0
ci04N2hAZXhhbXBsZS5jzb20AbmMteC10cgEBAIAAAAERPAC2YpaaACAvGM2QK
ywBxAARPEE8QRQLAY21wOmJyYkBib2IxLmV4YW1wbGUubmV0E3NpcDpi2J2
ZXhhbXBsZS5uZXQAFXNPCdphbgIjJ2UBeGFtCx11mNvbQhrbC0xEnRyLTg3
aEB1eGFtCxGmLnvBqZjLXkgHgC14LXRyAQIAdAAASKTzh1lAAAAAK8a8a
chQHMQ2EK8QlXBEFAQBkE3NpcDpi2J2AXXhhbXBsZS5uZXQYjetMRVzaX6A
WxpyY2V2AXXhhbXBsZS5jbb20EYWwteMRJ0ci04N2hAZXhhbXBsZS5jbb20GYy14
LXRyBnMteC10cgECAYHAAAEpE2YbyAAAACvLAHEBxjJKnChPee8QRQBEAC8BNz
aX6AeM9iQGV4YW1wbGUubmV0BQIgLXEVc21wOnFsawNLQGV4YW1wbGUuY29t
BGFsLTEhHId0ddaQGV4YW1wbGUuY29tBmMteC10cgZzLXgtdHlBAgB2AA
AKRNMIJ0AAAAXxjNkCzyzaAETxPUEQUCALQc21wOnJyYkB1eGFtCxGl1m51
daRiM0fxNpCdphbgIjJ2UBeGFtCx11mNvbQhrbC0xEnRyLTg3aEB1eGFt
CxGl1mNvbQZjLXkgHgC14LXRyAQIAdAAASKTz1dAAAlA8sAcOSQHM2QK
E8QXTBEFAQBkE3NpcDpi2J2AXXhhbXBsZS5uZXQYjetMRVzaX6A6WxpyY2V2A
ZXhhbXBsZS5jbb20EYWwteMRJ0ci04N2hAZXhhbXBsZS5jbb20GYy14LXRyBnM
tC10cgECAYHAAAEpE2YbyAAAACvLAHEBxjJKnChPee8QRQLAZBNzAA
```

The downstream branch call in Section 5.4 is in this message:
The forked call in Section 5.5 is in this message:

...The forked call in Section 5.5 is in this message:...
Authors’ Addresses

Brian Trammell
Swiss Federal Institute of Technology Zurich
Gloriagstrasse 35
8092 Zurich
Switzerland

Email: trammell@tik.ee.ethz.ch

Saverio Niccolini
NEC Laboratories Europe, NEC Europe Ltd.
Kurfuersten-Anlage 36
Heidelberg 69115
Germany

Phone: +49 (0) 6221 4342 118
Email: niccolini@neclab.eu
URI: http://www.neclab.eu

Benoit Claise
Cisco Systems Inc.
De Kleetlaan 6a b1
Diegem, 1813
Belgium

Phone: +32 2 704 5622
Fax:
Email: bclaise@cisco.com
URI:
Hadriel Kaplan
Acme Packet
71 Third Ave.
Burlington, MA  01803
USA

Phone:  
Email: hkaplan@acmepacket.com