IPFIX
Internet-Draft
Intended status: Informational
Expires: December 24, 2009

F. Huici
S. Niccolini
NEC Europe Ltd.
S. Anderson
Goettingen University

June 22, 2009

SIPFIX: Use Cases and Problem Statement for VoIP Monitoring and Exporting
draft-huici-ipfix-sipfix-00

Status of this Memo

Copyright Notice

Abstract

The deployment of Voice-over-IP (VoIP) telephony is increasing fast. VoIP's paradigm and the features it offers differ significantly from that of regular telephony, and, as a result, its monitoring requirements do so as well. This draft employs use cases to derive these requirements and introduces SIPFIX, an extension to IPFIX (IP Flow Information eXchange), that meets them.

Table of Contents

1. Introduction 3
2. Use Cases 4
2.1. Quality-of-Service Monitoring 4
2.2. Security and Troubleshooting 4
2.3. Billing 5
2.4. Law Enforcement 6
3. Problem Statement and Requirements 7
4. Solution: SIPFIX 8
4.1. Building Blocks 8
4.2. New Information Elements 8
4.2.1. SIP 8
4.2.2. Media 9
4.2.3. Performance Metrics 9
4.3. Flow Type Definitions 10
4.4. Recommended IPFIX Extensions 11
4.4.1. Bidirectional Flows 11
4.4.2. Common Properties 11
5. Security Considerations 13
6. IANA Considerations 14
7. Conclusions 15
8. References 16
Authors' Addresses 17

1. Introduction

The deployment of Voice-over-IP (VoIP) telephony is increasing fast. VoIP's paradigm and the features it offers differ significantly from that of regular telephony, and, as a result, its monitoring requirements do so as well. In addition to its wider set of features, the fact that VoIP runs over an unreliable network makes monitoring even more important if operators are to ensure good quality of service and secure VoIP calls against attack.

Fulfilling these requirements, however, presents a number of difficult challenges. The main goal of this draft is to introduce SIPFIX, a set of extensions to IPFIX [RFC5101] aimed at meeting these VoIP monitoring challenges, and in particular those of the two most popular protocols currently in use for VoIP telephony: SIP and RTP. In addition, the draft presents a number of use cases to illustrate VoIP monitoring requirements and to show the need for a standard solution such as SIPFIX.

2. Use Cases

This section introduces a number of VoIP monitoring use cases. The aim is to show that important VoIP applications need monitoring as well as a flexible mechanism such as SIPFIX to export monitored data. Further, such mechanism should be standardized in order to provide a better alternative to today's wide array of custom, non-interoperable solutions. Finally, this section provides the background and requirements needed to discuss the problem statement in the next section.

2.1. Quality-of-Service Monitoring

Quality of service monitoring is an important issue for VoIP providers and operators wishing to comply with service-level-agreements, to monitor the performance of the infrastructure for upgrade planning, or to generally provide a good call experience for their users. VoIP quality of service includes measuring signalling quality (e.g., session request delay, session completion ratio or hops for request), media QoS (e.g., jitter, delay or bit rate) and user experience (e.g., Mean Opinion Score). Calculating these metrics requires dealing not only with a potentially large amount of traffic in real-time, but also having to collect data from monitoring probes distributed throughout the network and aggregate them in a scalable way. In addition, the types of metric are quite varied, and so require a flexible, general data export mechanism.

2.2. Security and Troubleshooting

Because VoIP runs over a significantly different network than regular telephony, it is vulnerable to a host of different attacks. This section's focus is on SIP and RTP, since these are the most popular protocols currently in use for VoIP telephony. The following list, which is by no means exhaustive, gives a good overview of the types of attacks possible against VoIP traffic as well as the requirements for a monitoring system aimed at mitigating them.

- Spoofed media sender: Most SIP devices currently do not take the security of media streams into account. They expect packets to arrive at a certain IP address and port but normally do not inspect their origin, making it easy for an attacker to inject media packets in order to take over or disturb the media stream. A monitoring system could prevent these sort of attacks by detecting that two different media flows matched the same media flow descriptor of a SIP session.

- Stateful, cross-protocol IDS: Previous work on VoIP Intrusion Detection Systems proposed analyzing the traffic across different

protocol levels and tracking their states [scidive]. In order to do so, a monitoring system should be able to track the states of SIP sessions and correlate them with information from other protocols such as TCP.

- DoS attacks: Despite its popularity, many SIP implementations are still in their infancy and vulnerable to malicious messages. Cancel and Bye session attacks, as well as unregister attacks, flooding and SIP parser attacks are certainly feasible, and so a strong monitoring infrastructure is needed in order to detect and filter them. Such a monitoring system should also be able to detect DoS flooding attacks. Further, the system should be distributed and its results aggregated in order to catch attackers who try to avoid detection by sending small flooding rates to many destinations.

- Spam-over-IP telephony (SPIT): SPIT has recently become a problem and is likely to keep growing as VoIP adoption increases. A monitoring system could keep track of repeat offenders and block them from initiating further calls.

- Routing misconfigurations: Signalling packets may traverse a

number of routing hops when traveling from source to destination. Routing misconfigurations can potentially lead to degraded signalling quality or loss of signalling packets. To detect such misconfigurations, monitoring could be use to sample signalling packets in order to ensure that they are traversing the correct paths.

2.3. Billing

Billing is an essential element in telephony. Calculating accurate billing data for VoIP traffic presents new challenges when compared to regular telephony. A naive approach would be to monitor SIP INVITE and BYE messages, deriving a call's duration from these packets. Unfortunately, because VoIP has separate signalling and media streams, such a simplistic approach would be vulnerable to billing fraud: an attacker could reduce his or her bill by sending a BYE message while keeping the media session open.

To prevent this, the operator could distributedly monitor both streams, exporting the data from them to a common point for correlation. In this way, the operator would be able to detect billing frauds by noticing media streams that are alive despite their corresponding SIP session having finished. Because streams can follow different paths through a network (and even different return paths), such a system would have to ensure that no streams are missed. In addition, tracking media streams in real-time would

stress the monitoring system, so care should be taken to ensure that the infrastructure is scalable.

2.4. Law Enforcement

In order to comply with law enforcement agencies, operators are required to not only monitor VoIP traffic, but also to record it for after-the-fact auditing. Such a monitoring system should be flexible enough to export only the relevant data to reduce storage costs.

3. Problem Statement and Requirements

The VoIP applications described in the previous section impose a set of challenges and requirements on the monitoring infrastructure. Because VoIP traffic may traverse various points in the network, distributed monitoring is clearly a necessity. In addition, since the type of VoIP application varies quite significantly, the monitoring infrastructure needs application-specific, L7 monitoring

and exporting that is flexible enough to accommodate them.

Performing real-time distributed monitoring and exporting on a potentially large amount of traffic presents obvious scalability concerns. The naive approach of exporting data directly to a central collector does not scale, and so intermediate nodes called mediators are needed to pre-aggregate data before it reaches the collector. Such a mediator would have to be flexible and configurable in order to allow for the different types of VoIP applications that might use the monitoring infrastructure. For instance, mediation may consist of data reduction by aggregation or filtering, data correlation and combination from different devices, data modification (e.g., anonymization, encryption), or data storage. Another requirement for a monitoring infrastructure is a flexible export mechanism so that applications only export the data that they need, and no more.

While introducing intermediate mediators is a necessary step towards achieving scalability, these additional hops mean that data takes longer to arrive at the collector. Essentially this is a trade-off between aggregation and delay, and different applications will have different priorities. For instance, a billing application might care more about aggregation and not be so concerned if data are not reported every couple of seconds. As a result, in order to accommodate the largest possible number of applications, the monitoring infrastructure should allow per-application mediation and export settings.

A further difficulty arises from the fact that VoIP traffic splits signalling and media streams. As mentioned, several applications need to correlate these separate streams which may traverse disjoint paths through the network (indeed, due to path asymmetry, even a single stream may traverse different paths). The challenge for a monitoring infrastructure is then to ensure that both streams are captured and that, when correlation is needed, that exported data from them do not have to traverse a large number of hops before arriving at a common mediator that can correlate them.

4. Solution: SIPFIX

This section describes SIPFIX, a SIP/media-focused extension to IPFIX targeted at addressing some of the problems and requirements discussed in the previous section.

4.1. Building Blocks

SIPFIX is based on IPFIX (IP Flow Information Export), which consists of a protocol and an information model. IPFIX defines the transport and storage of general IP flow information, while allowing for a distributed, efficient and extensible monitoring architecture. In IPFIX, the traffic observation is handled by IPFIX Devices which obtain flow information from direct network observation and export these data to one or more receivers as Flow Records. The final receiver of the Flow Records is called a Collector, which is responsible for centrally processing and storing the flow information. IPFIX supports flexible flow definitions by using Template Records describing the order, type and size of each field for certain types of Flow Records. The type of a field is given by Information Elements (IE), and besides having a base set of IEs, IPFIX supports the definition of new ones.

In addition to IPFIX, SIPFIX also relies on so-called mediators. The original architecture of IPFIX comprises IPFIX Devices including Exporters that send out flow information, and Collectors that directly receive it. Since clearly this architecture does not scale, a draft [draft-mediators] has introduced the concept of a Mediator. A Mediator receives Flow Records from IPFIX devices or other Mediators and can process the data in a number of different ways, such as data reduction by aggregation or filtering, data correlation and combination from different devices, data modification (anonymization for instance), and data storage in distributed repositories.

4.2. New Information Elements

This section presents the SIPFIX Information Elements used to send SIP and media-related information to Mediators and Collectors. IPFIX supports new IEs either by defining enterprise-specific ones or by registering new IEs at the IANA registry. The new IPFIX IEs introduced here are either mandatory or optional, showcasing the possible functionality and feature extensions.

4.2.1. SIP

The following IEs contain information gathered from the header of SIP packets

Required fields

- sipFrom: the value of the From: header field.
- sipTo: the value of the To: header field.
- sipCallId: the value of the CallID: header field.

Optional fields

- sipRequestMethod: the method of a SIP request.
- sipRequestURI: the request-URI of a SIP request.
- sipResponseStatus: numerical status code of a SIP response.

4.2.2. Media

The IEs in this section describe specific properties of media streams related to a SIP session. This information is either gathered from media descriptors in the content of SIP packets (usually SDP), or from directly monitoring media packets.

- sipMediaId: a unique identifier for a media stream description of a SIP dialog.
- sipMediaProtocol: the transport protocol from a media stream description.
- sipMediaType: the media type from a media stream description.
- sipMediaEncoding: the encoding name from a media stream description.
- rtpPayloadType: the RTP payload type number.

4.2.3. Performance Metrics

Many metrics can be used in order to describe the characteristics of signalling and media streams ([voip-perf],[draft-pmol]); this section presents just a few possibilities.

- mediaPacketLoss: the ratio of lost packets to total packets.
- mediaDelay{To,From}Terminal: the one way delay from a media gateway to the terminal and vice versa.

- mediaDelayMGW: the one way delay from the ingress media gateway to the egress media gateway.

- rtpJitter: the inter-arrival jitter as defined by the RTP specification.

4.3. Flow Type Definitions

In order to transmit the information about SIP sessions and their related media streams, SIPFIX defines a set of special Flows. These Flows are constructed to make sure that the data can be correctly correlated by a Mediator or Collector.

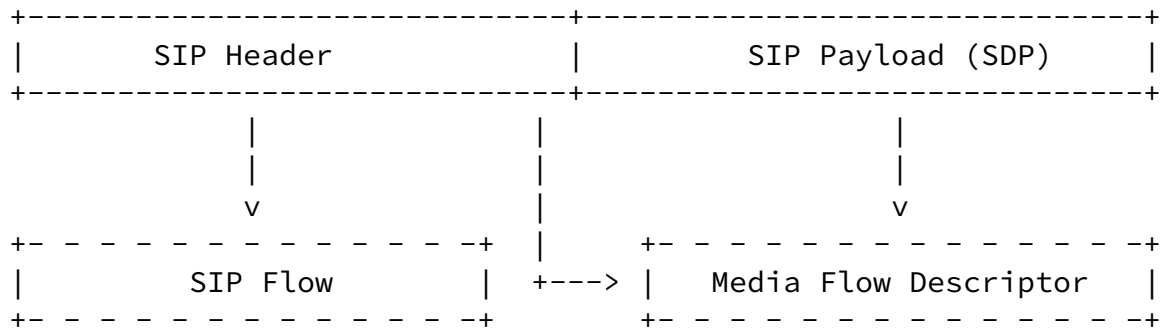

Figure 1: Dependencies of SIP and media flow descriptors

- SIP flow: A SIP Flow is a normal flow of SIP packets, but in addition to the normal fields it must include fields with the Information Elements <sipFrom, sipTo, sipCallId> which represent the sipDialogId. SIP Flow fields may include any number of SIP specific IEs such as those described in the previous section.

- Media Flow: A Media Flow is a normal flow of media packets. There are no mandatory fields, as these flows may also be exported by standard IPFIX devices unrelated to SIP monitoring. However, for

applications based on media-specific information like metrics for performance and QoS monitoring, the media probe can gather this information and export it in the Media Flow using media-specific IEs.

- Media Flow Descriptor: SIP media stream sessions are defined by media descriptions in the SIP packets' payloads. This data cannot be exported as normal SIP Flows fields since there can be an arbitrary number of media streams described in one single SIP

packet. To address this, SIPFIX defines a Media Flow Descriptor. This descriptor is not a real flow based on measured packet properties, but rather a pseudo flow that describes an expected Media Flow based on media descriptions contained in SIP packets, usually in the form of SDP information (see Figure 1). The Media Flow Descriptor must include the sipDialogId IEs and the sipMediaId in order to identify a SIP dialog and its corresponding media stream, respectively. As it is not a measured flow, it does not contain any kind of counter fields like number of packets or bytes.

4.4. Recommended IPFIX Extensions

This section proposes the use of two existing IPFIX extensions that optimize the export of the Flow Types described in the previous section. Although not strictly necessary, they are highly recommended as they improve the efficiency and functionality of SIPFIX.

4.4.1. Bidirectional Flows

RFC 5103 [RFC5103] defines a method to export associated bidirectional Flows (Biflows) in a single Flow Record. Two flows combine to a Biflow if all non-directional fields directly match and all source-related fields match the corresponding destination-related field of the other flow. The flows are merged by adding special IEs for counter fields of the "reverse" direction from the destination to the source.

This approach has several advantages. First, in most cases it is more efficient to assemble Biflows at the measuring device rather than at a Collector. Further, Biflows share information, so

exporting them individually generates a large amount of redundant data. Finally, the most important advantage for SIP monitoring is that Biflows keep directional information which might otherwise be lost: by using Biflows, the SIP flows of requests and responses can be merged so that the normal counter fields refer to the SIP requests, while reverse-counters refer to the SIP response packets.

4.4.2. Common Properties

Standard IPFIX may export several Flow Records with common properties or values, leading to a large amount of redundant data being transmitted. To improve this, RFC5473 [RFC5473] proposes a method to reduce the used bandwidth of IPFIX exports arising from redundant data.

SIPFIX can make extensive use of this method. For instance, the set

of IEs called sipDialogId described in Section 4.2.1 is often used as an identifier throughout SIFPIX and is even mandatory for SIP Flows and Media Flow Descriptors. As a result, it is advisable to define a template <commonPropertiesID | sipFrom, sipTo, sipCallId>.

5. Security Considerations

This document does not yet have any security considerations; these will be added in future revisions.

6. IANA Considerations

This document has no actions for IANA.

7. Conclusions

The deployment of Voice-over-IP (VoIP) telephony is increasing fast. VoIP's paradigm and the features it offers differ significantly from

that of regular telephony, and, as a result, its monitoring requirements do so as well. This draft introduced SIPFIX, a set of extensions to IPFIX aimed at meeting these VoIP monitoring requirements and providing a standard exporting mechanism in order to avoid the inter-operability problems generated by the wide array of solutions available today. Further, the draft presented a number of use cases to illustrate VoIP monitoring requirements and to show the need for a standard solution such as SIPFIX.

8. References

[RFC5101] Claise, B., "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information", RFC 5101, January 2008.

[RFC5103] Trammell, B. and E. Boschi, "Bidirectional Flow Export Using IP Flow Information Export (IPFIX)", RFC 5103, January 2008.

[RFC5473] Boschi, E., Mark, L., and B. Claise, "Reducing Redundancy in IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP) Reports", RFC 5473, March 2009.

[draft-mediators] Kobayashi, A. and B. Claise, "IPFIX Mediation: Problem Statement", draft-ietf-ipfix-mediators-problem-statement-03, April 2009.

[draft-pmol] Morton, A., "SIP End-to-End Performance Metrics", draft-ietf-pmol-sip-perf-metrics-03, March 2009.

[scidive] Wu, Y., Bagchi, S., Garg, S., Singh, N., and T. Tsai, "Scidive: A stateful and cross protocol intrusion detection architecture for voice-overip environments", In DSN 2004: Proceedings of the 2004 International Conference on Dependable Systems and Networks, 2004.

[voip-perf] ITU-T, "Recommendation Y.1530: Call processing performance for voice service in hybrid IP networks", 2004.

Authors' Addresses

Felipe Huici
NEC Laboratories Europe
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 241
Fax: +49 6221 4342 155
Email: felipe.huici@nw.neclab.eu
URI: http://www.nw.neclab.eu/

Saverio Niccolini
NEC Laboratories Europe
Kurfuerstenanlage 36
Heidelberg 69115
Germany

Phone: +49 6221 4342 118
Fax: +49 6221 4342 155
Email: saverio.niccolini@nw.neclab.eu
URI: http://www.nw.neclab.eu/

Sven Anderson
Goettingen University
Nikolausberger Weg 31
Goettingen 37073
Germany

Phone: +49 551 9969285
Fax: +49 551 37075678
Email: sven@anderson.de