IPFIX Working Group
Internet-Draft
Intended status: Standards Track
Expires: October 30, 2015

T. Fu
Huawei
D. Zhang

D. He
L. Xia
Huawei

April 28, 2015

IPFIX Information Elements for inspecting network security issues
draft-fu-ipfix-network-security-01

Abstract

IPFIX protocol has been used to carry Information Elements, which are defined to measure the traffic information and information related to the traffic observation point, traffic metering process and the exporting process. Network or device status are checked through analysing neccessary observed information. Although most of the existing Information Elements are useful for network security inspection, they are still not sufficient to determine the reasons behind observed events such as for DDOS attack, ICMP attack, and fragment attack. To allow administrators making effective and quick response to the attacks, this document extends the standard Information Elements and describes the formats for inspecting network security.

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 October 30, 2015.

Table of Contents

1. Terminology 2
2. Introduction 3
3. Information Elements and use cases 4
3.1. Information Elements 4
3.2. Packet upstream/downstream counters 9
3.3. ICMP echo/echo reply counters 9
3.4. Fragment statistic 9
3.5. Application error code 10
3.6. Extended value of FlowEndReason 10
4. Encoding 10
4.1. IPFIX 10
5. IANA Considerations 11
6. Security Considerations 13
7. References 13
7.1. Normative References 13
7.2. Informative References 13
Authors' Addresses 13

1. Terminology

IPFIX-specific terminology (Information Element, Template, Template Record, Options Template Record, Template Set, Collector, Exporter, Data Record, etc.) used in this document is defined in Section 2 of [RFC7011]. As in [RFC7011], these IPFIX-specific terms have the first letter of a word capitalized.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Introduction

As network security issues arising dramatically nowadays, network administrators are eager to detect and identify attacks as early as possible, generate countermeasurements with high agility. Due to the enormous amount of network attack types, metrics useful for attack detection are as diverse as attack patterns themselves. Moreover, attacking methods are evolved rapidly, which brings challenges to designing detect mechanism.

The IPFIX requirement [RFC3917] points out that one of the target applications of IPFIX is atack and intrusion detection. The IPFIX Protocol [RFC7011] defines a generic exchange mechanism for flow information and events. It supports source-triggered exporting of information due to the push model approach other than exporting upon flow-end or fixed time intervals.The IPFIX Information Model [RFC5102] defines a list of standard Information Elements (IEs) which can be carried by the IPFIX protocol. Eventhough the existing standard IEs are useful to check the status/events of the traffic, they are not sufficient to help network administrators identify categories of the attacks. The scanty information will result in an inaccurate analysis and slowing down the effective response towards network attacks.

For instance, CC (Challenge Collapsar) attack is a typical application layer DDoS attack, which mainly attacks the dynamic pages of web server. It makes the web server’s resources exhausted and paralyzed, so the server will be denial of service. Because CC attacker imitates normal users’ behavior pretty well by using different real IP addresses with relatively completive access process (even with low speed), it makes the attack concealed well compared with traditional network layer DDoS (e.g. SYN-Flood, etc). In addition, the attacker often manipulates the attack behind the scenes by non-direct communicate with target server, so the attack is not easy to be tracked and discovered. It would be useful to collect application status information for application layer attacks. In this case, CC attack is likely to happen if a large number of non 2XX HTTP status code replied from the server are observed.

Fragment attack employs unexpected formats of fragmentation, which will result in errors such as fragmentation buffer full, fragment overlapped, fragment incomplete. Existing IPFIX fragmentation metrics includes fragmentOffset, fragmentIdentification, fragmentFlags, which only indicate the attributes of a single fragment, and are not suitable for attack detection. Integrated measurements are needed to provide an holistic review of the session. Furthermore ICMP flow model has features such as the ICMP Echo/Echo Reply dominate the whole traffic flow, ICMP packet interval is

usually not too short (normally 1 pkt/s). The current ICMP information elements of IPFIX contains the ICMP type and code for both IPv4 and IPv6, however they are for a single ICMP packet rather than statistical property of the ICMP session. Further metrics like the cumulated sum of various counters should be calculated based on sampling method defined by the Packet SAMPling (PSAMP) protocol [RFC 5477]. Similar problems occur in TCP, UDP, SNMP and DNS attack, it would be useful to derive the number of the upstream and downstream packets separately and over time in order to detect the anomalies of the network.

Upon the above discussions and per IPFIX applicability [RFC 5472], derived metrics are useful to provide sufficient evidence about security incident. A wisely chosen sets of derived metrics will allow direct exporting with minimal resource consumption. This document extends the IPFIX Information model and defines Information Elements (IEs) that SHOULD be used to identify different attack categories, the standardization of those IEs will improve the network security and will support the offline analysis of data from different operators in the future.

3. Information Elements and use cases

This section presents the information elements that are useful for attack detection, the IPFIX templates could contain a subset of the Information Elements(IEs) shown in Table 1 depending upon the attack under concern of the network administrator. For example a session creation template contains

{sourceIPv4Address, destinationIpv4Address, sourceTransportPort, destinationTransportPort, protocolIdentifier, pktUpstreamCount, pktDownstreamCount, selectorAlgorithm, samplingPacketInterval, samplingPacketSpace}

An example of the actual event data record is shown below in a readable form

{192.168.0.201, 192.168.0.1, 51132, 80, 7, 67, 87, 3, 100,1000}

3.1. Information Elements

The following is the table of all the IEs that a device would need to export for attack statistic analysis. The formats of the IEs and the IPFIX IDs are listed below. Most of the IEs are defined in [IPFIX-IANA], while some of the IPFIX IE’s ID are not assigned yet, and hence the detailed explanation of these fields are presented in the following sections. The recommended registrations to IANA is described the IANA considerations section.

Field Name	Size (bits)	IANA IPFIX ID	Description
sourceIPv4Address	32	8	Source IPv4 Address
destinationIPv4Address	32	12	Destination IPv4 Address
sourceTransportPort	16	7	Source Port
destinationTransportPort	16	11	Destination port
protocolIdentifier	8	4	Transport protocol
packetDeltaCount	64	2	The number of incoming packets since the previous report (if any) for this Flow at the Observation Point
pktUpstreamCount	64	TBD	Upstream packet counter
pktDownstreamCount	64	TBD	Downstream packet counter
octetUpstreamCount	64	TBD	Upstream octet counter
octetDownstreamCount	64	TBD	Downstream octet counter
tcpSynTotalCount	64	218	The total number of packets of this Flow with TCP "Synchronize sequence numbers" (SYN) flag set
tcpFinTotalCount	64	219	The total number of packets of this Flow with TCP "No more data from sender" (FIN) flag set

tcpRstTotalCount	64	220	The total number of packets of this Flow with TCP "Reset the connection" (RST) flag set.
tcpPshTotalCount	64	221	The total number of packets of this Flow with TCP "Push Function" (PSH) flag set.
tcpAckTotalCount	64	222	The total number of packets of this Flow with TCP "Acknowledgment field significant" (ACK) flag set.
tcpUrgTotalCount	64	223	The total number of packets of this Flow with TCP "Urgent Pointer field significant" (URG) flag set.
tcpControlBits	8	6	TCP control bits observed for packets of this Flow
flowEndReason	8	136	The reason for Flow termination
minimumIpTotalLength	64	25	Length of the smallest packet observed for this Flow
maximumIpTotalLength	64	26	Length of the largest packet

			observed for this Flow
flowStartSeconds	dateTimeSeconds	150	The absolute timestamp of the first packet of this Flow
flowEndSeconds	dateTimeSeconds	151	The absolute timestamp of the last packet of this Flow
flowStartMilliseconds	dateTimeMilliseconds	152	The absolute timestamp of the first packet of this Flow
flowEndMilliseconds	dateTimeMilliseconds	153	The absolute timestamp of the last packet of this Flow
flowStartMicroseconds	dateTimeMicroseconds	154	The absolute timestamp of the first packet of this Flow
flowEndMicroseconds	dateTimeMicroseconds	155	The absolute timestamp of the last packet of this Flow
applicationErrorCodeCount	32	TBD	Number of packets with application error code detected
fragmentFlags	8	197	Fragmentation properties indicated by flags in the IPv4 packet header or the IPv6 Fragment header, respectively
fragmentIncompleteCount	32	TBD	Counter of incomplete

			fragments
fragmentFirstTooShortCount	32	TBD	Number of packets with first fragment too short
fragmentOffsettErrorCount	32	TBD	Number of fragments with offset error
fragmentFlagErrorCount	32	TBD	Number of fragments with flag error
icmpTypeIPv4	8	176	Type of the IPv4 ICMP message
icmpCodeIPv4	8	177	Code of the IPv4 ICMP message
icmpTypeIPv6	8	178	Type of the IPv6 ICMP message
icmpCodeIPv6	8	179	Code of the IPv6 ICMP message
icmpEchoCount	32	TBD	The number fo ICMP echo.
icmpEchoReplyCount	32	TBD	The number of ICMP echo reply.
selectorAlgorithm	16	304	This Information Element identifies the packet selection methods (e.g., Filtering, Sampling) that are applied by the Selection Process.
samplingPacketInterval	32	305	The number of packets that are consecutively sampled
samplingPacketSpace	32	306	The number of packets between two "s

			amplingPacketI nterval"s.

Table 1: Information Element table

3.2. Packet upstream/downstream counters

A sudden increase of Flow from different sources to one destination may be caused by an attack on a specific host or network node using spoofed addresses. However it may be cased by legitimate users who seek access to a recently published web content. Only reporting the total packet number is not sufficient to indicate whether attacks occur, as it lacks details to separate good packets from abnormoal packets. as a result, upstream and downstream packets should be monitored seperately so that upstream to downstream packet number ratio can be use to detect successful connections. pktUpstreamCount and pktDownstreamCount are added to IPFIX to represent the cumulated upstream and downstream packet number respectively.

3.3. ICMP echo/echo reply counters

An unusual ratio of ICMP echo to ICMP echo reply packets can refer to ICMP attack. However the existing set of IPFIX IEs provides the type and code of ICMP packet, countinuously export the information will result in serious resource consumption at the exporter, the collector and the bandwith. The number of echo and echo reply packets in a Flow can be derived for the Observation Domain in a specific time interval or once the ratio exceeds threshold. The basic metrics icmpEchoCount and icmpEchoReplyCount are defined as new IPFIX Information Elements.

3.4. Fragment statistic

Typical fragment attack includes fragmentation buffer full, fragment overlapped, fragment incomplete. Existing IPFIX fragmentation metrics includes fragmentIdentification,fragmentOffset, fragmentFlags, which are not sufficient to identify errors, and are not suitable for early attack detection. Integrated measurements are needed to provide an holistic review of the flow. fragmentIncompleteCount checks the number of incomplete fragmentation ,fragmentFirstTooShortCount verifies the number of fragments with first fragment too short, fragmentOffsetErrorCount checks the number of fragments with offset error, and fragmentFlagErrorCount detect early whether the fragmentation is caused by a malicious attack.

3.5. Application error code

The application layer attack requires IPFIX protocol capture packet payload. An initial consideration of the application error code comes from the HTTP status code except 2XX successful code. Other application layer protocol error code are also supported. The error code list can be expanded in the future as necessary. The data record will have the corresponding error code value to identify the error that is being logged.

3.6. Extended value of FlowEndReason

There are 5 defined reasons for Flow termination, with values ranging from 0x01 to 0x05:

0x01: idle timeout

0x02: active timeout

0x03: end of Flow detected

0x04: forced end

0x05: lack of resources

There is an additional reason caused by state machine anomaly. When FIN/SYN is sent, but no ACK is replied after a waiting timeout, the existing five reasons do not match this case.Therefore, a new value is proposed to extend the FlowEndReason, which is 0x06: protocol exception timeout.

4. Encoding

4.1. IPFIX

This document uses IPFIX as the encoding mechanism to monitor security events. However, the information that is logged SHOULD be the same irrespective of what kind of encoding scheme is used. IPFIX is chosen, because it is an IETF standard that meets all the needs for a reliable logging mechanism and one of its targets are for security applications. IPFIX provides the flexibility to the logging device to define the data sets that it is logging. The IEs specified for logging MUST be the same irrespective of the encoding mechanism used.

5. IANA Considerations

The following information elements are requested from IANA IPFIX registry.

Name : pktUpstreamCount

Description: The number of the upstream packets for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: TBD

Name: pktDownstreamCount

Description: The number of the downstream packets for this Flow at the Observation Point since the Metering Process (re-)initialization for this Observation Point.

Abstract Data Type: unsigned64

Data Type Semantics: TBD

Name: octetUpstreamCount

Description: The total number of octets in upstream packets for this Flow at the Observation Point 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: TBD

Name : octetDownstreamCount

Description: The total number of octets in downstream packets for this Flow at the Observation Point 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: TBD

Name: applicationErrorCodeCount

Description: This Information Element identifies the number of packets with application layer error code detected.

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentIncompleteCount

Description: This Information Element is the counter of incomplete fragments.

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentFirstTooShortCount

Description: This Information Element indicates the number of packets with first fragment too shortt.

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentOffsetErrorCount

Description: This Information Element specifies number of fragments with offset error.

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentFlagErrorCount

Description: This Information Element specifies number of fragments with offset error.When the DF bit and MF bit of the fragment flag are set in the same fragment, there is an error at the fragment flag.

Abstract Data Type: unsigned32

Data Type Semantics: TBD

A new values is added to FlowEndReason:

0x06: protocol exception timeout

The flow was terminated due to protocol state machine anomaly and unexpected timeout.

6. Security Considerations

No additional security considerations are introduced in this document. The same security considerations as for the IPFIX protocol [RFC7011] apply.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3971] Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure Neighbor Discovery (SEND)", RFC 3971, March 2005.

[RFC5102] Quittek, J., Bryant, S., Claise, B., Aitken, P., and J. Meyer, "Information Model for IP Flow Information Export", RFC 5102, January 2008.

[RFC5472] Zseby, T., Boschi, E., Brownlee, N., and B. Claise, "IP Flow Information Export (IPFIX) Applicability", RFC 5472, March 2009.

[RFC5477] Dietz, T., Claise, B., Aitken, P., Dressler, F., and G. Carle, "Information Model for Packet Sampling Exports", RFC 5477, March 2009.

[RFC7011] Claise, B., Trammell, B., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, September 2013.

7.2. Informative References

[IPFIX-IANA] IANA, "IPFIX Information Elements registry", <http://www.iana.org/assignments/ipfix>.

Authors' Addresses

Tianfu Fu
Huawei
Q11, Huanbao Yuan, 156 Beiqing Road, Haidian District
Beijing 100095
China

Email: futianfu@huawei.com

Dacheng Zhang

Email: dacheng.zhang@gmail.com

Danping He
Huawei
Q14, Huanbao Yuan, 156 Beiqing Road, Haidian District
Beijing 100095
China

Email: ana.hedanping@huawei.com

Liang Xia (Frank)
Huawei
No.101, Software Avenue, Yuhuatai District
Nanjing, Jiangsu 210012
China

Email: frank.xialiang@huawei.com