DOTS
Internet Draft
Intended status: Standard Track
Expires: April 2016

T. Fu
Huawei
D. Zhang
Alibaba
L. Xia
M. Li
Huawei

October 19, 2015

IPFIX IE Extensions for DDoS Attack Detection
draft-fu-dots-ipfix-extension-00.txt

Status of this Memo

This Internet-Draft will expire on April 19, 200916.

Copyright Notice

Abstract

Although most of the existing IP Flow Information Export (IPFIX) Information Elements (IEs) are useful for network security inspection, there are still some gaps existing to identify a number of categories of the attacks. To fill in the gaps, this document defines some new IPFIX IEs and describes their formats for inspecting network security.

Table of Contents

1. Introduction 3
2. Conventions used in this document 3
2.1. Terminology 3
3. Connection Sampling and new IEs 4
3.1. Packet Sampling vs Connection Sampling 4
3.2. Use Cases for New IEs 5
3.2.1. Upstream/Downstream Counters 5
3.2.2. Fragment statistic 5
3.2.3. Extended Value of FlowEndReason 6
3.3. Definition of New IEs 6
4. Application of the New IEs for Attack Detection 12
4.1. Use of Upstream/Downstream Counters to Detect ICMP Attack12
4.2. Fragment Attack 14
5. Security Considerations 15
6. IANA Considerations 15
7. References 20
7.1. Normative References 20
7.2. Informative References 20
8. Acknowledgments 20

1. Introduction

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

The IPFIX Protocol [RFC7011] defines a generic exchange mechanism for flow information and events. It supports source-triggered exporting of information via the push model approach. The IPFIX Information Model [IPFIX-IANA] defines a list of standard Information Elements (IEs) which can be carried by the IPFIX protocol. The IPFIX requirement [RFC3917] points out that one of the target applications of IPFIX is attack and intrusion detection. Although the existing standard IEs provide a rich source of data for security inspection by checking the status/events of the traffic, there are still some gaps existing to identify a number of categories of the attacks. The scanty information will result in an inaccurate analysis and slowing down the effective response towards network attacks. More detailed gap analysis is given in the following section.

This document presents the IPFIX IEs which are available for the network attacks detection, some of them are the new defined IPFIX IEs and their formats are specified. The wise utilization of these IEs will improve the network security and will support the offline analysis of data from different operators in the future with minimal resource consumption.

This document is structured as following: Section 3 discusses the connection sampling mechanism and introduces the new IPFIX IEs derived from relevant use cases. Section 4 describes how to use these IEs to detect specific DDoS attacks.

2. Conventions used in this document

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 RFC-2119 [RFC2119].

2.1. 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.

3. Connection Sampling and new IEs

3.1. Packet Sampling vs Connection Sampling

Packet sampling is a widely used method to select packets from network traffic for reporting. Its selection operations include random selection, deterministic selection, hash-based selection and so on. Although it is easy and efficient, it still has a number of limitations:

- Several research projects [N. DUFFIELD, 2003], [D. BRAUCKHOFF 2006] show that packet sampling impacts larger on small flows (with only few packets) due to the smaller sampling probability compared to larger flows, unfortunately attacks such as SYN-Flood, ACK-Flood all have small flow characteristics, which means that packet sampling may impair the detection performance for small flow based DDoS attacks;

- Although the communication is 2-way between source and destination, current packet sampling is applied independently in each direction, which leads to difficulties correlating the statistic of both sides, despite that those metrics are essential indicators in detecting attacks such as SNMP/DNS Reflected Amplification (i.e. where there are not much or even no traffic in the opposite direction of the attacking flow);

- Today's packet sampling cannot provide detailed information of traffic between communication peers, which makes it impossible to distinguish some of the attacks, such as IP fragment attack and Slowloris HTTP attack, from ordinary traffic.

As a consequence from the above analysis, a layer 4 connection oriented sampling method is more suitable for the security application: Rather than sampling a small part of packets in the traffic between the communication peers, the connection sampling records all TCP/UDP connection packets (including packets during connection setup and close phase if there is) between them once that connection is selected to be sampled. Furthermore, several new IPFIX IEs are proposed in this document to represent the telemetry information that can obtain via this method.

3.2. Use Cases for New IEs

In this section, several use cases are discussed to identify the requirements where new IEs are desirable for the network attacks detection.

3.2.1. Upstream/Downstream Counters

Take ICMP attack as an example, 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). Usually, the normal ratio between ICMP echo to ICMP echo reply packets is around 1:1. When a DDOS attack happens, a sudden burst of messages from different sources to a destination endpoint can be detected. In turn, the ratio between echo and echo reply packets will be significantly biased from the normal ratio, i.e., exceed 20:1. So, the proper way to distinguish an attack from the normal communication is to check this ratio.

However, the current IPFIX IEs for ICMP contain the ICMP type and code for both IPv4 and IPv6 only 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 [RFC5477]. Similar problems occur in TCP, UDP, SNMP and DNS attacks. It would be useful to calculate the number of the upstream and downstream packets for one connection separately over time in order to detect the anomalies of the network. For ICMP attack, a more generic approach is to define two basic metrics icmpEchoCount and icmpEchoReplyCount as new IPFIX IEs to represent the cumulated upstream and downstream packets counter within a ICMP connection. Similar new IE definitions of pktUpstreamCount, pktDownstreamCount, octetUpstreamCount, and octetDownstreamCount are applied to the TCP, UDP, SNMP and DNS attacks.

3.2.2. Fragment statistic

Fragment attack employs unexpected formats of fragmentation, e.g. without last fragment or incorrect fragment offset[RFC791], which result in errors such as fragmentation buffer overrun and fragment overlapped. Existing IPFIX fragmentation metrics includes fragmentOffset, fragmentIdentification, fragmentFlags, which only indicate the attributes of a single fragment, and are not suitable for attack detection. Instead, the network attack should be observed based upon a historic, integrated view of fragmented packets of a connection. For instances, if more than 500 out of 1000 fragmented

packets have fragment errors, it is likely that a fragment attack happens.

Therefore, a number of new IEs associated with fragment statistics are proposed as follows:

- fragmentIncompleteCount: The completeness of fragmented packets of the same connection should be checked, and this metric is proposed to count the incomplete events;

- fragmentFirstTooShortCount: Attacker might intent to exclude destination port from the first fragment so as to bypass detection from firewall. This metric is proposed to indicate the number of the invalid first fragments in the observed connection;

- fragmentOffestErrorCount: This metric is proposed to count the number of fragments with offset error, and the value can be used to indicate attack occurs;

- fragmentFlagErrorCount: This metric is proposed to detect early whether the fragment flags are incorrectly set on purpose.

3.2.3. Extended Value of FlowEndReason

Refer to [IPFIX-IANA], 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.

3.3. Definition of New IEs

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, as well as their descriptions. Some of the IEs are already defined in [IPFIX-IANA]. While a number of new IE's IDs are not assigned yet, their explanations are presented in the previous sections. The recommended registrations to IANA are described in 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 "Acknowled gment 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	dateTimeMiliseconds	152	The absolute timestamp of the first packet of this Flow
flowEndMilliseconds	dateTimeMiliseconds	153	The absolute timestamp of the last packet of this Flow
flowStartMicroseconds	dateTimeMicoseconds	154	The absolute timestamp of the first packet of this Flow
flowEndMicroseconds	dateTimeMicoseconds	155	The absolute timestamp of the last packet of this Flow
fragmentFlags	8	197	Fragmentation properties indicated by flags in the IPv4 packet header or the IPv6 Fragment header, respectively
fragmentPacketDeltaCount	32	TBD	Counter of session fragments
fragmentFirstTooShort	32	TBD	Number of

DeltaCount			packets with first fragment too short
fragmentFlagError DeltaCount	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
icmpEchoDeltaCount	32	TBD	The number fo ICMP echo.
icmpEchoReplyDeltaCount	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.
octetVariance	64	TBD	IP packet byte variance

			statistic
tcpControlStateBits	16	TBD	tcp states
flowSessionEndMilliseconds	64	TBD	the absolute timestamp of the first FIN or RST packet of this Flow
tcpPayloadOctetTotalCount	64	TBD	tcp payload statistics, it is equal to the ACK's window value subtract INIT's window value
tcpOutoforderTotalCount	64	TBD	out of order packets statistic
flowTimeIntervalVariance	64	TBD	the interval time variance between upstream and downstream traffic of a flow
flowTimeInterval	64	TBD	the interval time between upstream and downstream traffic of a flow
serverResponseTime	64	TBD	the response time of a server
clientResponseTime	64	TBD	the response time of a client
sessionResponseTime	64	TBD	the response time of a session

Table 1: Information Element Table

4. Application of the New IEs for Attack Detection

This section presents a number of examples to help for the easy understanding of the application of these new IEs for attack detection.

4.1. Use of Upstream/Downstream Counters to Detect ICMP Attack

According to previous analysis, the template for detecting ICMP attack should at least contain IEs shown in Table 2.

Set ID = 2	Length = 24 octets
Template ID TBD	Field Count = 10
|0| sourceIPv4Address	Field Length = 4
|0| destinationIPv4Address	Field Length = 4
|0| protocolIdentifier	Field Length = 1
|0| packetDeltaCount	Field Length = 8
|0| protocolIdentifier	Field Length = 1
|0| packetDeltaCount	Field Length = 8
|0| pktUpstreamCount	Field Length = 4
|0| pktDownstreamCount	Field Length = 4
|0| flowStartSeconds	Field Length = 4
|0| flowEndSeconds	Field Length = 4

Table 2: Template example for detecting ICMP attack

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

{sourceIPv4Address = 192.168.0.101, destinationIPv4Address = 192.168.0.201, protocolIdentifier = 1, packetDeltaCount = 3000, icmpEchoCount = 2880, icmpEchoRelayCount = 120, flowStartSeconds = 100, flowEndSeconds = 200}

protocolIdentifier = 1 represents the ICMP proptocol. There are 30 ICMP messages transmited per second. Tthe ICMP Echo to ICMP Echo Reply packet ratio is 24:1, which indicates a high possibility of ICMP attack.

4.2. Fragment Attack

The template for detecting fragment attack should at least contain IEs shown in Table 3. It requires the observation point to trace complete fragmented packet and accumulate the errors.

Set ID = 2	Length = 24 octets
Template ID TBD	Field Count = 10
|0| sourceIPv4Address	Field Length = 4
|0| destinationIPv4Address	Field Length = 4
|0| protocolIdentifier	Field Length = 1
|0| fragmentIncompleteCount	Field Length = 4
|0| fragmentFirstTooShortCount	Field Length = 4
|0| fragmentOffestErrorCount	Field Length = 4
|0| fragmentFlagErrorCount	Field Length = 4
|0| flowStartSeconds	Field Length = 4
|0| flowEndSeconds	Field Length = 4

Table 3: Template example for detecting fragment attack

An example of the actual fragment attack record is shown below in a readable form as below:

{sourceIPv4Address = 192.168.0.101, destinationIPv4Address = 192.168.0.201, protocolIdentifier = 1,fragmentIncompleteCount = 0, fragmentFirstTooShortCount = 0, fragmentOffestErrorCount = 3000, fragmentFlagErrorCount = 0, flowStartSeconds = 100, flowEndSeconds = 200}

In this case, fragment offset errors are used to exhaust resource at the receiver.

5. Security Considerations

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

6. 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: fragmentPacketDeltaCount

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

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentFirstTooShortDeltaCount

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

Abstract Data Type: unsigned32

Data Type Semantics: TBD

Name: fragmentFlagErrorDeltaCount

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

Name: octetVariance

Description: IP packet byte variance statistic.

Abstract Data Type: unsigned64

Data Type Semantics: quantity

Name: tcpControlStateBits

Description: tcp states.

Abstract Data Type: unsigned16

Data Type Semantics: flags

Name: icmpEchoDeltaCount

Description: icmp Echo packets.

Abstract Data Type: unsigned32

Data Type Semantics: deltaCounter

Name: icmpEchoReplyDeltaCount

Description: icmp Echo Reply packets.

Abstract Data Type: unsigned32

Data Type Semantics: deltaCounter

Name: flowSessionEndMilliseconds

Description: the absolute timestamp of the first FIN or RST packet of this flow.

Abstract Data Type: dateTimeMilliseconds

Data Type Semantics: default

Name: tcpPayloadOctetTotalCount

Description: tcp payload statistics, it is equal to the ACK's window value subtract INIT's window value.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

Name: tcpOutoforderTotalCount

Description: out of order packets statistic.

Abstract Data Type: unsigned64

Data Type Semantics: totalCounter

Name: flowTimeIntervalVariance

Description: the interval time variance between upstream and downstream.

Abstract Data Type: unsigned64

Data Type Semantics: quantity

Name: flowTimeInterval

Description: the interval time between upstream and downstream.

Abstract Data Type: unsigned32

Data Type Semantics: quantity

Name: flowTimeInterval

Description: the interval time between upstream and downstream.

Abstract Data Type: unsigned64

Data Type Semantics: quantity

Name: serverResponseTime

Description: the response time of a server.

Abstract Data Type: unsigned16

Data Type Semantics: quantity

Name: clientResponseTime

Description: the response time of a client.

Abstract Data Type: unsigned16

Data Type Semantics: quantity

Name: sessionResponseTime

Description: the response time of a session.

Abstract Data Type: unsigned16

Data Type Semantics: quantity

A new values is added to FlowEndReason:

0x06: protocol exception timeout

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

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.

[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.

[RFC3917] Quittek, J., Zseby, T., Claise, B., Zander, S., "Requirements for IP Flow Information Export (IPFIX)", RFC 3917, October 2004.

7.2. Informative References

[IPFIX-IANA]

IANA, "IPFIX Information Elements registry",

<http://www.iana.org/assignments/ipfix>.

[D. BRAUCKHOFF 2006]

Daniela Brauckhoff, Bernhard Tellenbach, Arno Wagner, Martin May, and Anukool Lakhina. 2006. Impact of packet sampling on anomaly detection metrics. In Proceedings of the 6th ACM SIGCOMM conference on Internet measurement (IMC '06). ACM, New York, NY, USA, 159-164.

[N. DUFFIELD, 2003]

DUFFIELD, N., LUND, C., AND THORUP, M., Estimating Flow Distributions from Sampled Flow Statistics. In ACM SIGCOMM (Karlsruhe, August 2003).

8. Acknowledgments

The authors would thank Danping He and Yibo Zhang for their great help during the initial period of this draft.

This document was prepared using 2-Word-v2.0.template.dot.

Authors' Addresses

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

Email: futianfu@huawei.com

DaCheng Zhang
Alibaba

Email: Dacheng.zdc@alibaba-inc.com

Liang Xia (Frank)
Huawei

101 Software Avenue, Yuhuatai District
Nanjing, Jiangsu 210012
China

Email: Frank.xialiang@huawei.com

Min Li
Huawei

Huawei Technologies Duesseldorf GmbH, European Research Center,
Riesstr. 25, 80992 Muchen, Germany
Email: l.min@huawei.com