Internet Engineering Task Force (IETF)
Internet-Draft
Intended status: Informational
Expires: January 4, 2018

T. Fu
C. Zhou
H. Zheng
Huawei

July 03, 2017

IP Flow Information Export (IPFIX) Information Elements Extension for TCP Connection Tracking
draft-fu-ipfix-tcp-tracking-00

Abstract

This document proposes several new TCP connection related Information Elements (IEs) for the IP Flow Information Export (IPFIX) protocol. The new Information Elements can be used to export certain characteristics regarding a TCP connection. Through massive gathering of such characteristics, it can help build an image of the TCP traffic passing through a network. The image will facilitate the detection of anomaly TCP traffic, especially attacks targeting at TCP.

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 January 4, 2018.

Copyright Notice

Copyright (c) 2017 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

Due to its stateful operations, TCP [RFC0793] is vulnerable to attacks. The SYN Flood attack is an example. It is sourced from a massive number of malicious clients starting TCP connections with a server, but never completing the three-way handshake process, leaving the server-side of the connections in waiting states, eventually exhausting the server resources and no new connection can be created.

Attack aiming at TCP can also be low and slow in traffic pattern. These attacks may not take down the server, but just impair the provided service. Even though a victim server is still operating, its performance can be significantly degraded. Without the insight of what is going on with the TCP traffics, this kind of situation can be very hard to detect and analyze.

For a network device, such as a router, to detect anomaly TCP traffics, it has to understand the semantics of TCP operations, more specifically, it has to be able to track TCP connection states. If a

router has implemented such an ability, it can export characteristics information regarding the TCP connections. Offline analysis can be performed over the gathered information, which will facilitate the detection of anomaly TCP traffics and identify attacks.

The IP Flow Information Export (IPFIX) protocol [RFC7011] already defines a generic mechanism for flow information export. This document introduces several new Information Elements of IPFIX, that can be used to export TCP connection characteristics.

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 (e.g. Information Element, Template, Template Record, Options Template Record, Template Set, Collector, Exporter, Data Record) 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.

This document also makes use of the same terminology and definitions as Section 2 of [RFC5470].

o Victim: The target that suffers from DDoS attack.

3. New IEs and Connection Sampling

3.1. Proposed New Information Elements

The proposed new Information Elements are listed in Figure 1 below.

Field Name	IANA IPFIX ID
tcpHandshakeSyn2SynAckTime	TBD
tcpHandshakeSynAck2AckTime	TBD
tcpHandshakeSyn2AckRttTime	TBD
tcpConnectionTrackingBits	TBD
tcpPacketIntervalAverage	TBD
tcpPacketIntervalVariance	TBD
tcpOutOfOrderDeltaCount	TBD

Figure 1: New Information Elements

The Information Elements defined in Figure 1 are proposed to be incorporated into the IANA IPFIX Information Elements registry [IPFIX-IANA] Their definitions can be found at Section 7.

3.2. Use Cases for New IEs

Below are several use cases to identify the requirements where new IEs are desirable for the network attacks detection.

3.2.1. Response Time Calculation of TCP Handshake

For the DDoS attacks such as http slowloris, there will be many TCP inactive, low traffic connections that are kept in the victim (server), which leads to excessive resource consumption. As a result, the response time between valid clients and the server for even the TCP handshake will increase greatly. The Challenge Collapasar(CC) attack can also exhaust the resources of the server and generate the similar results. In summary, too much resource consumption in the victim will increase the response time of TCP handshake, which is a general network anomaly condition. The following IEs are proposed to report symptoms of these kinds of attacks:

tcpHandshakeSyn2SynAckTime: Denotes the time difference between the time point that the Metering Process detects the SYN packet from client to server and the time point that the Metering Process then detects the SYN-ACK packet from server to client.

tcpHandshakeSynAck2AckTime: Denotes the time difference between the time point that the Metering Process detects the SYN-ACK packet from server to client and the time point that the Metering Process then detects the ACK packet from client to server.

tcpHandshakeSyn2AckRttTime: Denotes the sum of tcpHandshakeSyn2SynAckTime and tcpHandshakeSynAck2AckTime. It is the Round Trip Time (RTT) of a TCP handshake between client and server.

3.2.2. Symptoms of Exceptions

Slow packet attacks at the application layer, such as http slowloris attack, slow http post attack, or slow read attack, the malicious clients may send packets to the victim periodically at a very low rate which causes the performance lost on the server. The characteristic of this kind of attack is that there are too many connections on the victim, while the traffic volume for these connections is small. In order to detect this attack, two new IEs, tcpPacketIntervalAverage and tcpPacketIntervalVariance are helpful. The IE tcpPacketIntervalAverage denotes the average time difference between two successive packets and the IE tcpPacketIntervalVariance denotes the variance of multiple time difference. Large tcpPacketIntervalAverage and small tcpPacketIntervalVariance can be a symptom of slow packet attack, since the attacker sends packets in large intervals just as to keep the connection open, and the intervals tend to differ very little in time.

The malicious clients may send too many out-of-order packets, which will consume too much memory on the server, thus degrading performance. Although out-of-order packets are permit in the TCP protocol, it is possible to be leveraged to cause these attacks. The IE tcpOutOfOrderDeltaCount is helpful to detect this kind of exception. The Metering Process maintains one counter for each TCP connection. The initial sequence number of the client is saved in the counter. The counter increases by the sequence number of the packets it sees from client to server. If the Metering Process sees a packet with a lower sequence number than the current counter value, then the packet will be considered as an out-of-order packet.

In IPFIX, the IE tcpControlBits is used to record the corresponding status bits in TCP header of the packets. In order to detect the application layer attacks which can cause the protocol exception such as the wrong use of the TCP status bits during the TCP connection establishment, another IE called tcpConnectionTrackingBits is needed. For example, when the Metering Process sees the SYN packet from client to server, it sets 15th bit of tcpConnectionTrackingBits to 1; when it sees the SYN-ACK packet from server to client, it sets 14th bit to 1, and so on. If one endpoint sends the packet with wrong bits during the establishment of the connection, then the Metering Process will identify the exception by the value of tcpConnectionTrackingBits.

4. Application of the New IEs for Attack Detection

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

4.1. Detect Slowloris Attack

The template for detecting resource exhausting application layer attack such as http slowloris attack should contain a subset of IEs shown in Figure 2.

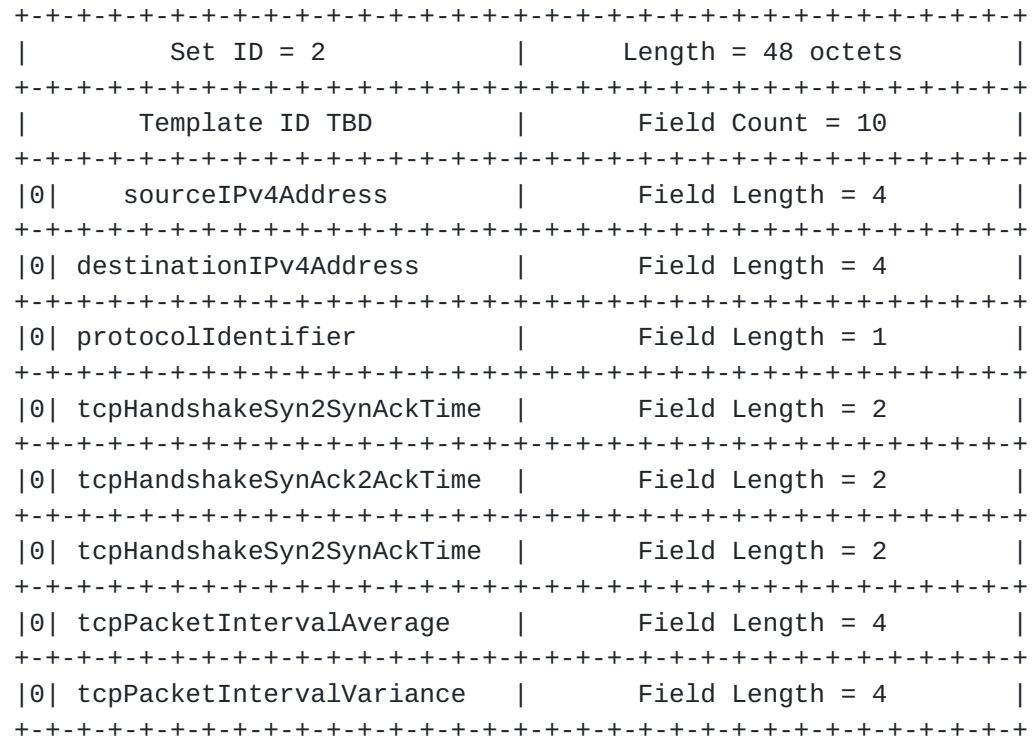

Figure 2: Template Example for Detecting Slowloris Attack

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

{sourceIPv4Address = 192.168.0.101, destinationIPv4Address = 192.168.0.201, protocolIdentifier = 6, tcpHandshakeSyn2SynAckTime = 1, tcpHandshakeSynAck2AckTime = 3, tcpHandshakeSyn2AckRttTime = 4, tcpPacketIntervalAverage = 5635, tcpPacketIntervalVariance = 38216}

4.2. Detect Out-of-order Packets Attack

The template for detecting out-of-order packets attack should contain IEs shown in Figure 3.

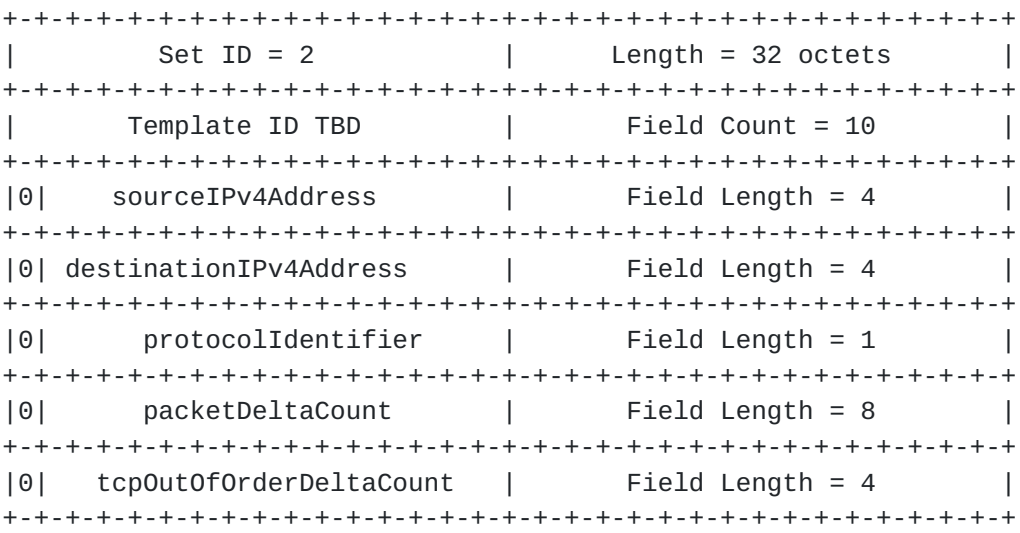

Figure 3: Template Example for Detecting Out-of-order Attack

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

{sourceIPv4Address = 192.168.0.101, destinationIPv4Address = 192.168.0.201, protocolIdentifier = 6, packetDeltaCount =3000, tcpOutOfOrderDeltaCount = 2000}

4.3. TCP Connection Tracking Status Report

Figure 4: Template Example for TCP Connection Tracking

The following text lists several examples. For a TCP connection that ends normally, the bit pattern is:

```
   Bit 15 (SYN) = 1
   Bit 14 (S/A) = 1
   Bit 13 (ACK) = 1
   Bit 12 (FIN) = 1
   Bit 11 (ACK) = 1
   Bit 10 (F/A) = 1
   Bit 09 (ACK) = 1
   Bit 08 (RST) = 0
   Bit 07 (TMR) = 0
   Bit 06 (END) = 1
   Bit 05,04 (END REASON) = 00
   Bit 03 (ROP) = 0
   Bit 02 (ROD) = 0
   Bit 01 (ERR) = 0
   Bit 00 (VLD) = 1

   tcpConnectionTrackingBits = 0b1111111001000001 = 65089
```

the actual record is shown in a readable form as below:

```
   {sourceIPv4Address = 192.168.0.101, destinationIPv4Address =
   192.168.0.201, protocolIdentifier = 6,
   tcpConnectionTrackingBits = 65089 }
```

Another example is an abnormal case, that RST is received after SYN, the bit pattern is:

```
   Bit 15 (SYN) = 1
   Bit 14 (S/A) = 0
   Bit 13 (ACK) = 0
   Bit 12 (FIN) = 0
   Bit 11 (ACK) = 0
   Bit 10 (F/A) = 0
   Bit 09 (ACK) = 0
   Bit 08 (RST) = 1
   Bit 07 (TMR) = 0
   Bit 06 (END) = 0
   Bit 05,04 (END REASON) = 01
   Bit 03 (ROP) = 0
   Bit 02 (ROD) = 0
   Bit 01 (ERR) = 0
   Bit 00 (VLD) = 0

   tcpConnectionTrackingBits = 0b1000000100010000 = 33040
```

the actual record is shown in a readable form as below:

{sourceIPv4Address = 192.168.0.101, destinationIPv4Address = 192.168.0.201, protocolIdentifier = 6, tcpConnectionTrackingBits = 33040 }

5. Summary

This document proposes several new TCP connection related IEs of the IPFIX protocol, which can be used to export certain characteristics regarding a TCP connection. Through gathering of such characteristics, an image can be built (normal baseline or anomaly) of the TCP traffics passing through a network. The image will facilitate the detection of the attacks targeting at TCP connections.

6. Security Considerations

This document proposes several new TCP connection related IPFIX IEs and their use in the detection of some kinds of TCP connection related attack. Comparing to IPFIX basic protocol [RFC7011] there is no new security threats brought by the new proposed IEs, as long as all the security considerations and mechanisms presented in [RFC7011] are followed.

The new proposed IEs and solutions do not cover all the existing TCP connection related attacks, let along those new attacks that will appear in future. DDoS attack and their detection is an 'arms race', useful telemetry information is always needed to protect the network resources better.

7. IANA Considerations

The following information elements are requested from IANA IPFIX registry. Upon acceptation, the 'TBD' values of the ElementIds should be replaced by IANA for assigned numbers.

Name: tcpHandshakeSyn2SynAckTime
Description:
The time difference between a SYN and its corresponding SYN-ACK when the Metering Process detects a new TCP connection is going to be set up.
Abstract Data Type: dateTimeMicroseconds
ElementId: TBD
Status: current
Units: microseconds

Name: tcpHandshakeSynAck2AckTime
Description:
The time difference between a SYN-ACK and its corresponding ACK

when the Metering Process observes a new TCP connection is going to be set up.

Abstract Data Type: dateTimeMicroseconds
ElementId: TBD
Status: current
Units: microseconds

Name: tcpHandshakeSyn2AckRttTime
Description:
The time difference between a SYN and its corresponding ACK sent from the same endpoint when the Metering Process observes a new TCP connection is going to be set up.

Conceptually tcpHandshakeSyn2AckRttTime can be thought as the sum of tcpHandshakeSyn2SynAckTime and tcpHandshakeSynAck2AckTime, but practically the values may differ.

Abstract Data Type: dateTimeMicroseconds
ElementId: TBD
Status: current
Units: microseconds

Name: tcpConnectionTrackingBits
Description:
These bits are used by the Metering Process to track a TCP connection. A bit is set to 1 if the corresponding condition is met. A value of 0 for a bit indicates the corresponding condition is not met.

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|1|1|1|1|1|1|0|0|0|0|0|0|0|0|0|0|
|5|4|3|2|1|0|9|8|7|6|5|4|3|2|1|0|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|S|S|A|F|A|F|A|R|T|E|END|R|R|E|V|
|Y|/|C|I|C|/|C|S|M|N|REA|O|O|R|L|
|N|A|K|N|K|A|K|T|R|D|SON|P|D|R|D|
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Bit 15 (SYN):
Set when there is no TCP connection between the endpoints and the Metering Process detects a SYN as it is used to setup a new TCP connection. The Metering Process starts to track the TCP connection.

Bit 14 (S/A):
Set when bit 15 has been set and the Metering Process detects a SYN-ACK in the flow, which effectively acknowledges the SYN (Ack = Seq + 1) causing bit 15 to be

set.

Bit 13 (ACK):
Set when bit 15 and bit 14 have been set and the Metering Process detects an ACK which effectively acknowledges the SYN-ACK (Ack = Seq + 1) causing bit 14 to be set. Upon setting this bit, it means handshake of the TCP connection setup has completed.

Bit 12 (FIN):
Set when the Metering Process detects the first FIN for the established and tracked TCP connection. It means the TCP connection is going to be closed.

Bit 11 (ACK):
Set when bit 12 has been set and the Metering Process detects an ACK which effectively acknowledges the FIN causing bit 12 to be set.

Bit 10 (F/A):
Set when bit 12 has been set and the Metering Process detects a FIN that is from the opposite of the endpoint which sent the FIN causing bit 12 to be set.

Bit 09 (ACK):
Set when bit 10 has been set and the Metering Process detects an ACK that is from the same endpoint which sent the FIN causing bit 10 to be set.

Bit 08 (RST):
Set when the Metering Process detects any RST from either party of the tracked TCP connection. Setting this bit indicates that TCP connection is abnormal and aborted.

Bit 07 (TMR):
Set when a flow record report is triggered by a periodic reporting timer. It means the TCP connection is still under tracking.

Bit 06 (END):
Set when the Metering Process has stopped tracking the TCP connection, as the connection has been closed or aborted.

Bit 05 & Bit 04 (END REASON):
00: as default value when the TCP connection is not closed, or the tracked TCP connection is closed normally.
01: the tracked TCP connection is aborted.
10: the tracked TCP connection is inactive after a period of time.
11: reserved.

Bit 03 (ROP):
Set when the Metering Process detects any SYN or SYNACK, after the both endpoints have sent FIN or an RST has been detected.

Bit 02 (ROD):
Set when the Metering Process detects at least 50 TCP segments being exchanged, after both endpoints have sent FIN

or an RST has been detected.
Bit 01 (ERR):
Set when the Metering Process detects any of the following abnormal signaling sequences for the TCP connection: SYN/FIN, SYN/FIN/PSH, SYN/FIN/RST, SYN/FIN/RST/PSH.
Bit 00 (VLD):
When the END bit is set, setting this bit indicates the tracked TCP connection is closed normally. Otherwise, indicates the tracked TCP connection is aborted.

Abstract Data Type: unsigned16
Data Type Semantics: flags
ElementId: TBD
Status: current

Name: tcpPacketIntervalAverage
Description:
The average time interval calculated by the Metering Process between two successive packets in the data flow of a TCP connection.
Abstract Data Type: dateTimeMilliseconds
ElementId: TBD
Status: current

Name: tcpPacketIntervalVariance
Description:
The variance of the time intervals calculated by the Metering Process between two successive packets in the data flow of a TCP connection.
Abstract Data Type: unsigned64
ElementId: TBD
Status: current

Name: tcpOutOfOrderDeltaCount
Description:
The number of out of order packets in the data flow of a TCP connection detected at the Observation Point since the previous report.
Abstract Data Type: unsigned64
Data Type Semantics: deltaCounter
ElementId: TBD
Status: current

8. Acknowledgments

The authors would like to acknowledge the following people, for their contributions on this text: DaCheng Zhang, Bo Zhang (Alex), Min Li, Robert Moskowitz.

9. References

9.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC5470] Sadasivan, G., Brownlee, N., Claise, B., and J. Quittek, "Architecture for IP Flow Information Export", RFC 5470, DOI 10.17487/RFC5470, March 2009, <http://www.rfc-editor.org/info/rfc5470>.

[RFC7011] Claise, B., Ed., Trammell, B., Ed., and P. Aitken, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information", STD 77, RFC 7011, DOI 10.17487/RFC7011, September 2013, <http://www.rfc-editor.org/info/rfc7011>.

9.2. Informative References

[IPFIX-IANA] IANA, "IPFIX Information Elements Registry", July 2017, <http://www.iana.org/assignments/ipfix>.

[RFC0793] Postel, J., "Transmission Control Protocol", STD 7, RFC 793, DOI 10.17487/RFC0793, September 1981, <http://www.rfc-editor.org/info/rfc793>.

Authors' Addresses

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

Email: futianfu@huawei.com

Chong Zhou
Huawei
156 Beiqing Road
M06 Shichuang Technology Demonstration Park
Haidian District
Beijing 100094
China

Email: mr.zhouchong@huawei.com

Hui Zheng (Marvin)
Huawei
101 Ruanjian Avenue, Yuhuatai District
Nanjing 210012
China

Email: marvin.zhenghui@huawei.com