Flow-based Network Measurement—NetFlow & IPFIX

Hitoshi Irino†*, Masaru Katayama, and Shinichiro Chaki*

Abstract

This article describes our activities related to NetFlow, which is a de facto standard protocol for exporting flow information, and IPFIX, which is a protocol standardized in IETF (Internet Engineering Task Force). The method of exporting flow information by sending aggregated packets from routers, which has been deployed recently, is a promising alternative to the conventional method that obtains values of counters at interfaces in routers.

1. Introduction

To manage a network, it is necessary to monitor the amount of traffic and detect problems when they occur such as failures or congestion. The method that obtains the values of transmission and reception counters from interfaces in routers by SNMP (simple network management protocol) is widely used. Although SNMP is simple and lightweight to process, it does not let us easily analyze each connection. When detailed analysis is required, an alternative approach is to collect information about each packet. There are two methods. In one, an external collecting device collects copied packets by using port mirroring on switches. In the other, an external collecting device collects partial information about packets by using the sFlow protocol, which was invented by InMon Corporation.

Recently, a different method has been used. In this method, an external collecting device collects flow information, namely information about aggregated packets having the same attributes, which is classified in network equipment. This method enables the external collecting device to collect more detailed information than one using SNMP. On the other hand, less flow information is obtained than when raw packets are collected. Therefore, the method using flow information is suitable when we want to know the rough tendency of network usage. Cisco's NetFlow is a de facto standard protocol for this method (**Table 1**). NetFlow technology can be classified into several versions: NetFlow version 9 (v9) has been published as RFC3954 (informational document).

IETF has also standardized the IPFIX (IP flow information export) protocol [1], [2], which is a standard protocol for IP (Internet protocol) networks based on NetFlow v9. IPFIX is a more reliable protocol than NetFlow v9, and it defines more collectable information than NetFlow v9.

2. Protocols for exporting flow information

In NetFlow and IPFIX, network equipment (e.g., a router) called an Exporter periodically sends flow information to a collecting device called a Collector (**Fig. 1**). Exporters export two kinds of information: Data and a Template. Data represents flow information. Its structure can be defined by the Templates in NetFlow v9 and in IPFIX because required traffic information depends on the purpose of the measurement and the structure of the network. The relationship between Template and Data is shown in **Fig. 2**.

The Template shown on the left side of the upper block defines the fields of the Data shown on the right side of the upper block by defining the ID and length of Information Elements (IEs). Any flow Data Record, which is a unit of flow information, can be defined as a combination of IEs. For example, an

† NTT Network Service System Laboratories
Musashino-shi, 180-8585 Japan
Email: irino.hitoshi@lab.ntt.co.jp

Table 1. Summary of protocols for measuring network traffic.

Protocol	SNMP	sFlow	NetFlow (v9)	IPFIX
Type of information	MIB counter	Partial packets chosen by sampling	Flow	
Amount of data	Small	Large (depending on sampling rate)	Between SNMP and sFlow (depending on sampling rate and flow creation conditions)	
Collectable information	Amount of data of interface	Data from data-link layer (containing packet header and data of partial packet payload)	Data from data-link layer to transport layer	
				Data other than the above is collected by vendor extensions.
Status of standardization	RFC3411, RFC3418, etc. (standard)	RFC3176 (informational by InMon)	RFC3954 (informational by Cisco)	Stage immediately before publication as an RFC (standard)

MIB: management information base
RFC: request for comments

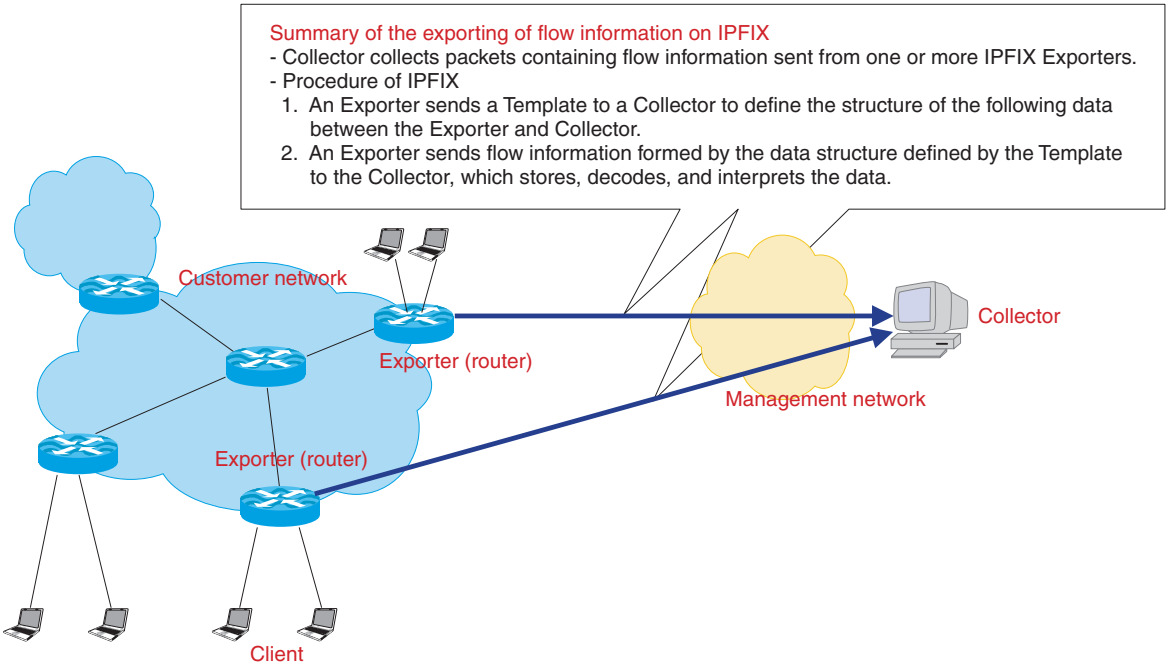

Fig. 1. Example of exporting flow information by IPFIX and NetFlow v9.

IPv6 (Internet protocol version 6) flow can be represented by using "sourceIPv6Address" instead of "sourceIPv4Address" and "destinationIPv6Address" instead of "destinationIPv4Address" in the Template shown in Fig. 2. Exporters send created Templates to Collectors by the following method.

The header of a Set (Set Header) is used to distinguish between Data and Templates. The Set ID contained in Set Header is 2 if the Set (which contains multiple Records) is a Template Set; it is 3 if the Set is an Option Template Set (described below), and it is a number between 256 and 65,535 if the Set is a Data Set.

A Template ID is used to relate a Template Record to a Data Record. The Template ID is contained in the Template Record Header if the Record is a Template Record. The Set ID is the same number as the Template ID (256 in Fig. 2) if the Record is a Data Record.

Option Templates and Data related to Option Templates provide optional information. An Option Template record is added to the Scope of a Template record to indicate the applicable scope of optional information. In the example shown in Fig. 2, the Scope is defined as a template ID in an Option Template Record, and the Option Data Record applies to

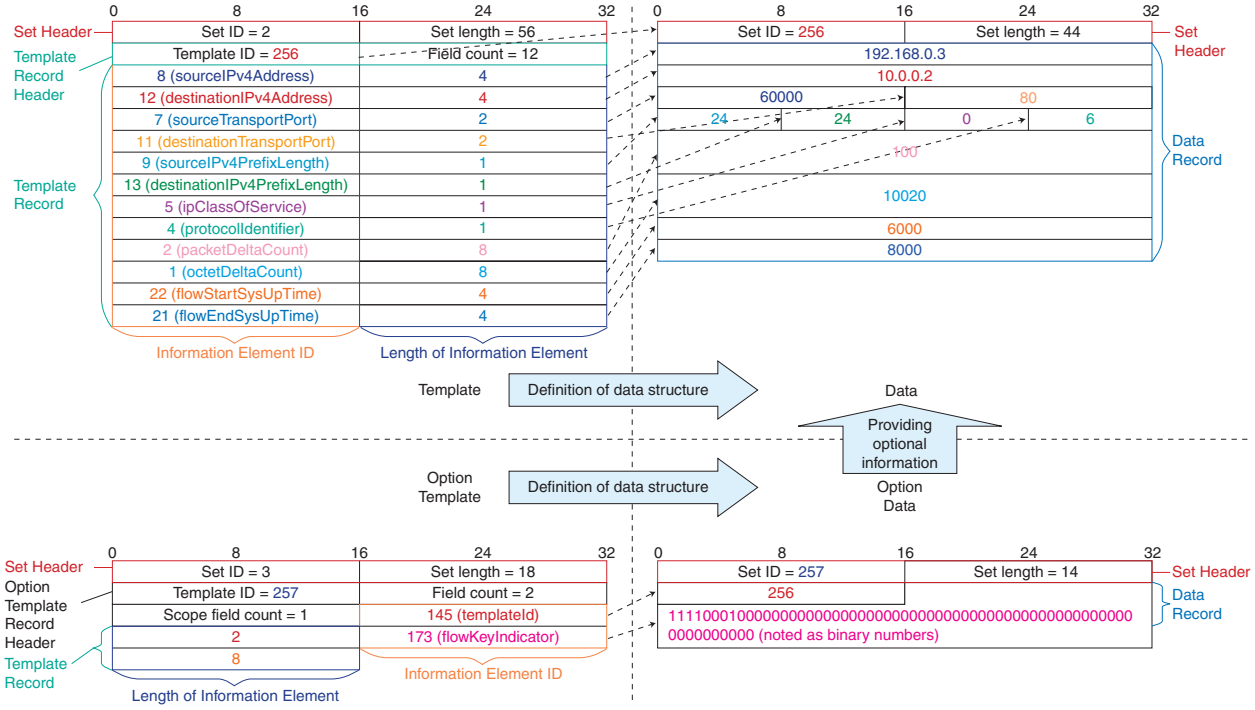

Fig. 2. Relation between Template and Data of IPFIX and NetFlow v9.

Photo 1. Permanent display of Moving Firewall version 4 in Musashino R&D Center.

a Data Record whose template ID is 256. An IE flowKeyIndicator shows the conditions for creating a flow.

The IEs, which will be standardized in IETF, contain the information contained in the IP header, transport header, and the header of the sub-IP layer protocol (e.g., MPLS (multiprotocol label switching) and routing information (e.g., AS (autonomous system) number). The number of standard IEs is 169. Moreover, a method for defining enterprise-specific IEs is defined. For example, information about the session layer (e.g., RTP (real-time transport protocol) can be represented by using vendor-specific IEs. IPFIX can manage any traffic information that can be represented by flows using a combination of the IEs described above.

3. Activities of NTT R&D

NTT Network Service System Laboratories (NS Labs.) developed Moving Firewall Version 4 (MFWv4) in conjunction with NTT Information Sharing Platform Laboratories (PF Labs.). MFWv4 detects anomalous traffic (e.g., a distributed denial of service (DDoS) attack) using NetFlow v5, which is an ancestor of IPFIX. A photograph of MFWv4 being displayed in Musashino R&D Center is shown in **Photo 1**.

To receive a large amount of flow information exported by many routers and process it quickly, MFWv4 has a hardware component: the Gbit-RNP with proprietary firmware for NetFlow v5 (**Photo 2**). When this technology is applied to networks in future, including the next-generation network (NGN), it will have to be extended to handle IPFIX because NetFlow v5 can handle only IPv4 flow information. There are many differences between IPFIX and NetFlow v5. It is more difficult for a Collector to process flow information using IPFIX than NetFlow v5 because IPFIX is a variable format based on the Template unlike NetFlow v5, which uses a fixed format. Therefore, it is difficult to receive a large amount of

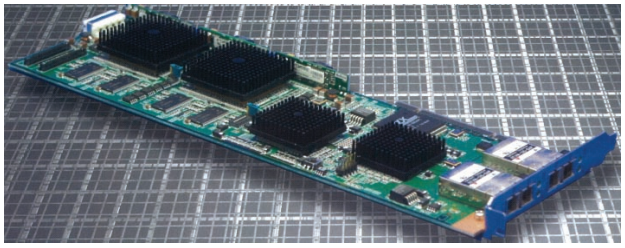

Photo 2. Gbit-RNP.

flow information and process it quickly. This difficulty is one of the main obstacles to the introduction of flow-based traffic measurement in advanced large networks like the NGN.

3.1 Proposals to IETF

The drafts "Reference Model for IPFIX Mediators" [3] and "Order of Information Elements" [4] have been proposed to the IPFIX working group (WG) of IETF by NTT NS Labs. and PF Labs., respectively, for the purpose of collecting flow information in a large-scale network.

3.1.1 IPFIX Mediators

One application of IPFIX Mediators is an aggregator. This aggregates flows exported from the Exporter and exports aggregated data to Collectors in a cascade connection of IPFIX devices. Operators can obtain not only the rough trend of network traffic in a large-scale network, but also detailed flow data in a portion of the network because the IPFIX Mediators store the original exported flow data before aggregation.

3.1.2 Order of IEs

This was proposed to achieve Collector implementation in a hardware-based fast collecting process with analyzing functions. IPFIX can configure exporting flow information by using the Template mechanism. Moreover, the Template mechanism allows IEs to be positioned regardless of data boundaries. Different orders of IEs among multiple Templates create different Templates with different formats even if the Templates contain the same set of IEs. Collectors must manage these templates individually even though their information is essentially the same. This redundancy is an inefficient implementation of the Collector in hardware, which has resource constraints. The proposal reduces the occurrences of inefficient situations. Even if the order is unified, the features of IPFIX will not be affected. In the draft, the order is considered based on the sizes of IEs.

Hardware designed based on the dataflow architecture is suitable for processing information that is ordered. Although the processing of this architecture depends on the order of incoming data, the architecture can process data in parallel using many small and simple processing units. An overview of the dataflow architecture is given in **Fig. 3**. This architecture can achieve a higher degree of parallel processing than a general CPU (central processing unit) architecture for an general-purpose personal computer, so the dataflow architecture can achieve higher processing performance than a general CPU running at the same clock frequency. MFWv4 with Gbit-RNP, which can process flow information at a wire speed of 1 Gbit/s, uses this architecture. We expect to achieve higher performance by introducing a unified order and using the dataflow architecture.

The unified order can yield higher performance with not only the hardware Collector using the dataflow architecture but also a software-based Collector running on an ordinary CPU. We implemented a primitive software-based Collector that copies data of predetermined fields in incoming data records into a file using the Collector's internal data structure. This collector supports the copying of multiple items of IE data at once if these multiple IEs are positioned sequentially in a Template exported from the Exporter and in a Collector's internal data structure. Our evaluation found that the speed of a primitive Collector's processing, which stores data of predetermined fields in incoming data records using the same proposed order for IEs as in the Template exported from the Exporter, was up to 80% faster [5] than when the order between a Template exported from the Exporter and the internal data structure stored in the Collector was different. The reason for the improvement is that the probability of using a multiple copy function was higher when the same order was used between an Exporter and a Collector.

4. Conclusion

We presented flow-based traffic measurement methods, especially IPFIX and NetFlow v9, and our activities concerning these protocols. We will work to propose our ideas to IETF, and we will also improve the feasibility of the hardware-based Collector to make a high-performance Collector that can measure network traffic in a large-scale network.

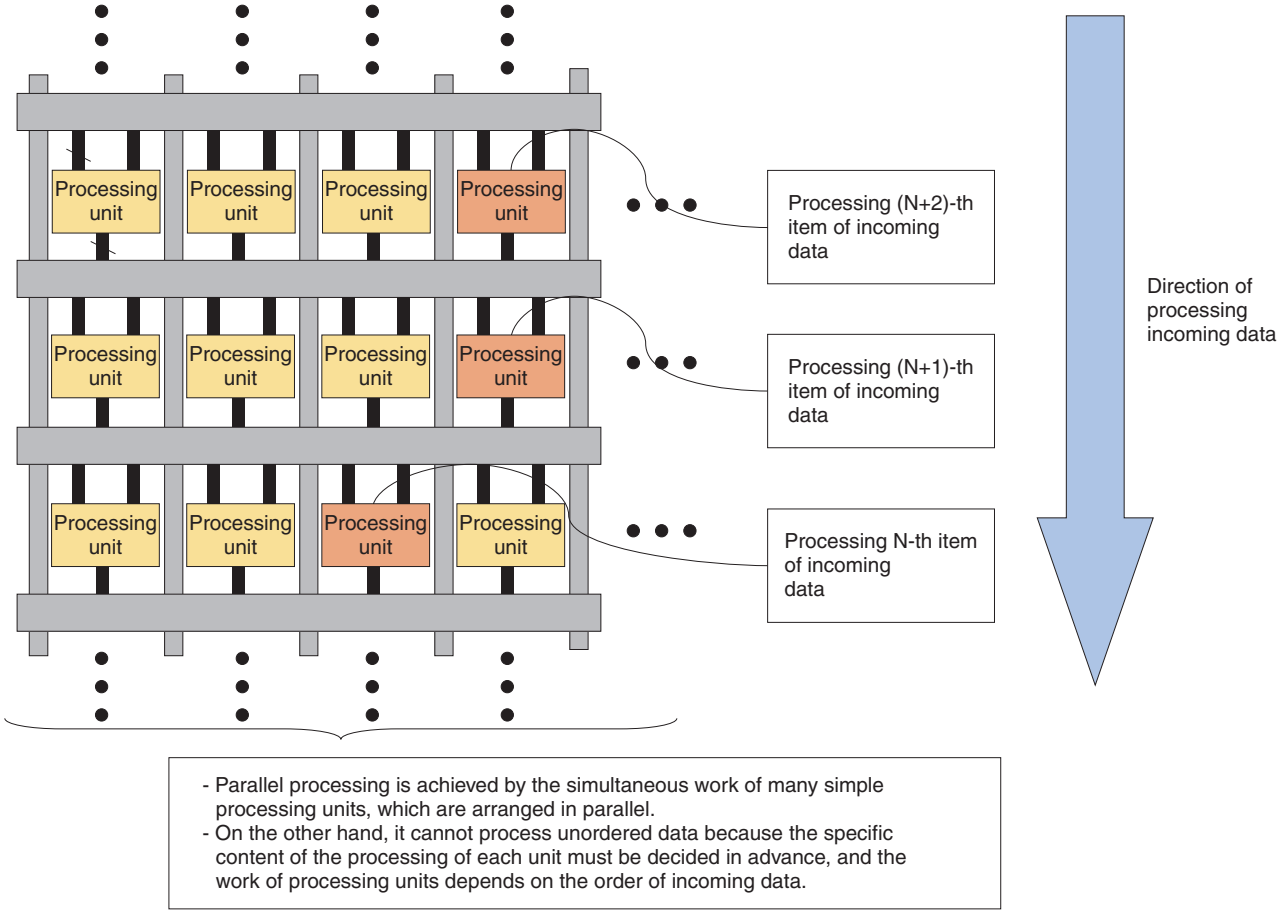

Fig. 3. Overview of the data flow architecture.

References

[1] http://www.ietf.org/html.charters/ipfix-charter.html
[2] B. Claise, "Specification of the IPFIX Protocol for the Exchange of IP Traffic Flow Information," Internet Draft, draft-ietf-ipfix-protocol-26.txt, Sep. 2007 (work in progress).
[3] A. Kobayashi, K. Ishibashi, T. Kondoh, and D. Matsubara, "Reference Model for IPFIX Mediators," Internet Draft, draft-kobayashi-ipfix-mediator-model-01.txt, Nov. 2007 (work in progress).
[4] H. Irino, "Guidelines for the Order of Information Elements," draft-irino-ipfix-ie-order-03, Internet Draft, Nov. 2007 (work in progress).
[5] http://www3.ietf.org/proceedings/07jul/slides/ipfix-10.pdf

Hitoshi Irino
Broadband Network Systems Project, NTT Network Service Systems Laboratories.

He received the B.A. degree in environmental information and the M.A. degree in media and governance from Keio University, Kanagawa, in 2003 and 2005, respectively. He joined NTT Network Service Systems Laboratories in 2005. His current research interest is in high-performance traffic analysis systems. He is a member of IEEE.

Masaru Katayama
Senior Research Engineer, Broadband Network Systems Project, NTT Network Service Systems Laboratories.

He received the B.E. and M.E. degrees from Hokkaido University, Hokkaido, in 1990 and 1992, respectively. He joined NTT in 1992 and has been engaged in research on system LSI design and its design methodologies using rapid prototyping systems. He has accumulated considerable experience in developing a telecommunication-oriented FPGA, called "PROTEUS-Lite," and its development software systems. His current research interests include a high-performance IP packet processing system and its control system with field programmable hardware systems (such as FPGAs and reconfigurable processors). He is a member of IEEE and the Institute of Electronics, Information and Communications Engineers (IEICE) of Japan.

Shinichiro Chaki
Group Leader, Broadband Network Systems Project, NTT Network Service Systems Laboratories.

He received the B.E. degree from Sophia University, Tokyo, and the M.E. degree from Waseda University, Tokyo, in 1986 and 1988, respectively. He joined NTT Switching Systems Laboratories in 1988 and worked on the development of an ATM traffic control scheme and the first commercial ATM switching system for frame relay service. He was transferred to NTT Access Network Service Systems Laboratories in 1998, where he worked on standardization of the VB5 interface in ITU-T and the development of B-PON systems for the ATM Mega-data link service. He joined NTT Service Integration Laboratories in 2002 and engaged in basic network design for FLET'S Hikari Premium service based on IPv6 technology. He has been engaged in R&D of value-added functions for the transport network since 2004. He is a member of IEEE and IEICE.