Open access • Journal Article • DOI:10.1007/S10922-009-9138-0

Using NetFlow/IPFIX for Network Management — Source link

Aiko Pras, Ramin Sadre, Anna Sperotto, Tiago Fioreze ...+2 more authors

Institutions: University of Twente, University of Zurich, Jacobs University Bremen

Published on: 01 Dec 2009 - Journal of Network and Systems Management (Springer)

Topics: NetFlow and Network management

Using NetFlow/IPFIX for Network Management

Pras, A; Sadre, R; Sperotto, A; Fioreze, T; Hausheer, D; Schönwälder, J

Using NetFlow/IPFIX for network management

Report of the joint IRTF/NMRG & Emanics workshop

Aiko Pras* Ramin Sadre* Anna Sperotto*
Tiago Fioreze* David Hausheer** Jürgen Schönwälder***

Abstract

To exchange experiences with, and to discuss ideas on the usage of NetFlow/IPFIX in network management, the IRTF/NMRG, together with the European EMANICS Network of Excellence, organized a one-day workshop in October 2008. This paper presents a report of that meeting.

keywords: NetFlow, IPFIX, IRTF/NMRG, EMANICS

1 Introduction

NetFlow [1] is a technology developed by Cisco for the purpose of monitoring traffic flows within (high speed) networks. Version 9 of NetFlow is currently standardized by the IETF under the name *IP Flow Information Export* (IPFIX)[2]. According to the IPFIX standard, a flow is defined as

> "a set of IP packets passing an observation point in the network during a certain time interval. All packets belonging to a particular flow have a set of common properties."

These properties are generally expressed in terms of *flow keys*, which can for example be the source and destination addresses, the source and destination port numbers and the IP protocol. All packets that share the same values for these flow keys belong to the same flow.

To exchange experiences with, and to discuss ideas on the usage of NetFlow/IPFIX in network management, the Network Management Research Group (NMRG) [3] of the Internet Research Task Force (IRTF), together with the European EMANICS Network of Excellence [4], organized a one day workshop on October 30th, 2008, at the Leibniz Rechenzentrum (LRZ) in Munich. The workshop was attended by about 40 people from industry and academia.

*University of Twente, The Netherlands; [a.pras|r.sadre|a.sperotto|t.fioreze]@utwente.nl
**University of Zürich, Switzerland, hausheer@ifi.uzh.ch
***Jacobs University Bremen, Germany, j.schoenwaelder@jacobs-university.de

The workshop was opened by Benoit Claise, who gave an overview of NetFlow/IPFIX. Amongst others, he discussed the differences between NetFlow versions 5, 7, 8 and 9 (IPFIX). Of these four versions, version 5 and 9 are the most popular ones. Compared to version 9, version 5 supports a relatively small number of flow keys. In addition, version 5 exports flow information in a fixed format, whereas version 9 allows the network operator to tailor the export format, using so-called templates. Flexible NetFlow, which provides an even larger set of possible flow keys and supports the definition of new keys, was introduced shortly. Also the relation between NetFlow and Packet Sampling (PSAMP), which are complementary techniques, was discussed. His keynote concluded with future challenges, and the question "what, if we could start IPFIX from scratch"?

The remainder of the day was structured to answer the following questions:

- What technologies exist to capture flows at high data rates?
- What technologies exist to analyze flow data?
- How do sampling and aggregation affect the volume and accuracy of data collection and analysis?
- For what kind of applications can NetFlow/IPFIX be used?

This report provides an overview of the various workshop presentations, and summarizes the main conclusions. The report does not intend to give complete answers to all these questions; for more details the reader is referred to the slides and minutes, which can be downloaded from the NMRG website [5][6].

2 Technologies for flow capturing

The workshop started with two presentations on how to capture flows at 10 Gbps. Luca Deri (ntop.org) discussed the use of commodity hardware for this purpose. Although PC adapters for 10 Gigabit Ethernet are already available for prices below 1000 Euro, the problem with current CPUs is that the increase in processor performance is primarily realized by increasing the number of cores, and not by increasing the speed of an individual core. To cope with higher network speeds, a single thread (running on a single core) will no longer be sufficient to analyze all packets. Instead, it is important to spread the packets over multiple threads, running on multiple cores. The drawback of this approach, however, is that threads need to compete for packets, which requires expensive synchronization (mutex) operations. Such kernel operations have severe impact on performance. To overcome this problem, Luca Deri proposed *Threaded New API* (TNAPI), which is a kernel extension that exploits the capabilities of 10 Gbps hardware cards by creating multiple receive queues. These queues are polled by different TNAPI threads, and packets can be cached in different Packet Filtering constructs (PF-RINGs). In this way, high speed flow capturing becomes possible on commodity hardware.

In the second presentation, Jiri Novotny (Masaryk University) proposed the opposite approach. He discussed the problems of using standard PCs or routers for flow capturing, and argued that special monitoring hardware is needed. An example of such hardware is FlowMon, which was developed within the European Scampi project[7], and which uses co-called Combo-cards that support speeds up to 10 Gbps. In his presentation, Jiri Novotny presented the architecture and performance of this hardware.

3 Technologies for flow analysis

The next group of presentations discussed current and future technologies for flow analysis. All presentations shared the same, common problem, namely *how to collect, how to store and how to analyze large amount of flow data.*

In her talk "Using SQL databases for flow processing", Anna Sperotto (University of Twente) reported on the operational experiences using large databases for flow analysis. For the purpose of intrusion detection and, more recently, botnet detection, she collected flow traces over a period of two years from university, national and international research networks. The size of these traces is huge; one trace covering a period of three days from the university network, for example, contained fifty GB of data, and more than one billion flow records. A conservative approach, which relies on SQL databases, was taken to store and analyze the data. Although this choice made data querying easy, *a posteriori* considerations, such as database size and query time, suggest that this database approach has severe drawbacks. The generation of database indexes, for example, took several days and increased the database size from 50 to 87 GB. Another drawback is that similar queries, but defined in slightly different ways, could show considerable differences in query time. As illustration, one example was presented in which query time could be reduced from 7 days to 11 minutes.

The other presenters proposed novel approaches to overcome the limitation of the existing tools. Vladislav Marinov (Jacobs University) showed that existing query languages (such as SQL, BPF, ACL and SRL) can't describe complex traffic patterns, particularly in cases where flows have causal dependencies. He proposed therefore a new flow record query language, and used the example of the Blaster worm to illustrate the application of this language.

Cristian Morariu (University of Zürich) proposed to distribute flow data over multiple analyzers. He presented his Distributed IP (DIP) Storage architecture, which allows dynamic configuration, avoids a single point of failure, provides load balancing, and allows the use of commodity hardware. Since a prototype has been developed, performance figures were also presented.

4 The effect of sampling and aggregation

The primary goal of sampling and flow aggregation is to decrease the amount of data that needs to be collected and processed. With packet sampling, every n^{th}

packet will be inspected, instead of all packets. With flow aggregation, flows with similar characteristics will be merged into one bigger flow.

Tiago Fioreze (University of Twente) analyzed the effects of sampling for 3 flow metrics: octets, packets, and flow duration. For this purpose, he selected from a large collection of flows only those that traverse three concatenated networks. The first network did not perform any sampling, the second used a sampling ratio of 1:100, and the third a sampling ratio of 1:1000. To make results as representative as possible, he decided to capture real data, instead of simulated data or data from a controlled lab environment. His experiments showed that, as may have been expected, sampling did not create any artifacts in the number of reported octets and packets (of course, the numbers reported by NetFlow should be compensated by the sampling ratio). Flow duration, however, was considerably affected by sampling. The analysis showed that average flow duration was reduced by 15% in case of 1:100 sampling, and 31% in case of 1:1000 sampling. The cause of this deviation is that traffic tends to be bursty; the few packets that are exchanged in the relative silent periods, may be missed in the case sampling is applied. As a result, NetFlow assumes the flow has expired and reports, instead of one single long flow, multiple shorter flows. In case of sampling, NetFlow also reported a large number of flows with duration zero (due to the fact that only a single packets was captured).

In the same session, Christoph Sommer (University of Erlangen) discussed hierarchical flow aggregation, and identified problems and open questions. The key idea is to reduce the number of flows, either by merging similar flows into bigger, aggregated flows, or by filtering flows (which is comparable to sampling).

5 Management applications

The last session discussed possible flow-based management applications. Many of these applications focus on intrusion detection. In his presentation, Tobias Limmer (University of Erlangen) discussed the use of flow techniques to distinguish between successful and unsuccessful TCP connections; such unsuccessful connections can, for example, be created by scans. A test setup was described, in which two hours of TCP data were captured and analyzed. Flow data were merged into two-directional "biflows", and analyzed using a number of custom scripts. The outcome was compared to that of Bro, which is a well-known packet-based IDS system. Initial results are interesting, although this research is still in its initial stage.

In another presentation Sven Anderson (University of Göttingen / NEC Labs) discussed the application of NetFlow/IPFIX for VoIP monitoring. Special flow-based SIP probes, called SIPFIX, were developed, which inspect and export application layer information. These probes not only inspect the SIP header to determine source, destination and call-id, but also the SIP content, to describe characteristics of the media. A number of new IPFIX information elements were proposed, which can be exchanged using option templates. In the discussion that followed the presentation possible extensions to NetFlow/IPFIX were discussed.

In the final presentation Olivier Festor (INRIA Loria) discussed how to discover application level dependencies, using flows. This work, which is still at an initial stage, might be useful for fault and configuration management, intrusion detection and for improving the performance of networks and systems. Current work in this area focuses on (deep) packet inspection, and has several drawbacks that may be overcome by using flow data only. For example, flow analysis is cheaper, in terms of resource consumption, than packet analysis. In addition, legal issues make analysis of packet payload problematic, if not impossible. It was interesting to observe that there is a strong relationship between this work, and the work presented earlier that day by Vladislav Marinov.

6 Conclusions

This first workshop on the use of NetFlow/IPFIX for network management was generally seen as very successful. The number of attendees was high, and the various presentations resulted into interesting discussions. It was decided to organize a special journal issue on this topic, and a second workshop, which will take place October 6, 2009 in Bremen. For further information on that workshop, please contact the authors of this report.

We would like to thank Heinz-Gerd Hegering and Helmut Reiser (Leibniz-Rechenzentrum) for hosting this workshop. This workshop was supported in part by the IST Network of Excellence EMANICS, funded by the European Union under contract number FP6-2004-IST-026854-NoE.

References

[1] Cisco Systems, "NetFlow Services Solution Guide", 2007

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

[3] Homepage IRTF/NMRG: *http://www.ibr.cs.tu-bs.de/projects/nmrg/*

[4] Emanics Network of Excellence: *http://www.emanics.org/*

[5] Agenda of the 25th NMRG meeting, October 2008, Munich, Germany: *http://www.ibr.cs.tu-bs.de/projects/nmrg/meetings/2008/munich/*

[6] Minutes of the 25th NMRG meeting, October 2008, Munich, Germany: *http://www.ibr.cs.tu-bs.de/projects/nmrg/minutes/minutes-025.txt*

[7] Homepage of the EU Scampi project: *http://www.ist-scampi.org/*