Monitoring an academic network with Netflow

David López, Jorge E. López-de-Vergara, Luis Bellido, David Fernández

Abstract: Quality of Service delivery in network services has been a mayor issue for both customers and providers, but it is only recently that the possibility to integrate network performance metrics into a global Network Management system has become a reality thanks to modern analysis tools and computing power improvement.

In order to define and characterize quality of service as the basis to provide reliable network services, several initiatives like IPPM and IPFIX have been arranged.

This paper describes the work performed to monitor an academic network combining both SNMP and IPFIX technologies.

Index terms-- **Network management, Network monitoring, Netflow, Application QoS, IPPM, IPFIX.**

I.Introduction

Network management is a keystone in today networks: they monitor and control network resources to deliver a reliable 24x7 operation. In this context, monitors and probes have been used to measure their performance among other management functionalities; there are however, some relevant aspects related to quality of service not yet addressed, with the advent of voice over IP there is a growing interest on both provider's and customer's side for enhanced network services capable of fulfilling Service Level Agreements (SLA's). With regard to this matter, flow accounting architectures seems to have a promising future as a performance auditing technology.

IP Flow Information Export (IPFIX Charter) [1] appears as a standard way to obtain network information about flows. These flows are defined as sets of IP packets passing an observation point in the network during a certain time interval. Flow measurements are performed by routers and probes. The information about these flows can be exported from these devices to be processed in other management systems.

This paper describes an architecture based on IPFIX specifications pointing out a set of new modules that have been developed in order to enhance the usability of the system. Finally, after discussing some of the results obtained from the measurement experience, the last section presents some conclusions and future work.

II.IPFIX and Netflow

Netflow [2] is a proprietary technology developed by CISCO systems in 1996. Its main purpose was the performance improvement of former Layer 3 Switching Technologies. For that, it identified traffic flows between internetworking hosts, switching later every frame belonging to that flow on a connection-oriented basis.

Netflow, being targeted at first time as a switching path, is nowadays emerging as a useful network accounting technology, capable of characterizing most of the properties that the traversing IP packets possess; in order to do so, seven attributes (termed as flow-keys) are defined to identify each flow:

- *Source IP address*
- *Destination IP address*
- *Source port*
- *Destination port*
- *Layer 3 protocol type*
- *TOS byte (DSCP)*
- *Input logical interface (ifIndex)*

A stream of packets with an equal combination of keys is uniquely identified as a flow as seen by the router in a limited timeframe. Once a new flow has been detected, the router timestamps the relevant information like number of packets, TCP Flags, octets, as well as interfaces by which the traffic was routed. Therefore, it provides a useful insight into many layers of the TCP/IP stack.

Netflow version 9 has been recently proposed by the IPFIX working group as the standard exporting protocol for data measurement and flow information export [3]. This version of Netflow, supersedes former versions with extended functionalities like Multicast support, MPLS Label accounting, BGP Next Hop information as well as a template based protocol model ensuring future backwards compatible implementations.

III.Monitoring an academic network with Netflow

As a proof of concept of how flow accounting can effectively contribute to manage and provision a campus network, an experimental monitoring infrastructure has been deployed at the School of Telecommunications Engineering (ETSIT-UPM). So far the results obtained have provided a valuable feedback for both ongoing and

Authors can be contacted at the Telematic Systems Engineering Department, Technical University of Madrid (DIT-UPM) - Av. Complutense, s/n - 28040 Madrid, Spain. Their email addresses are: {lopezber,jlopez,lbt,david}@dit.upm.es. David López and Jorge E. López de Vergara are currently working at Terra Networks and Universidad Autónoma de Madrid respectively.

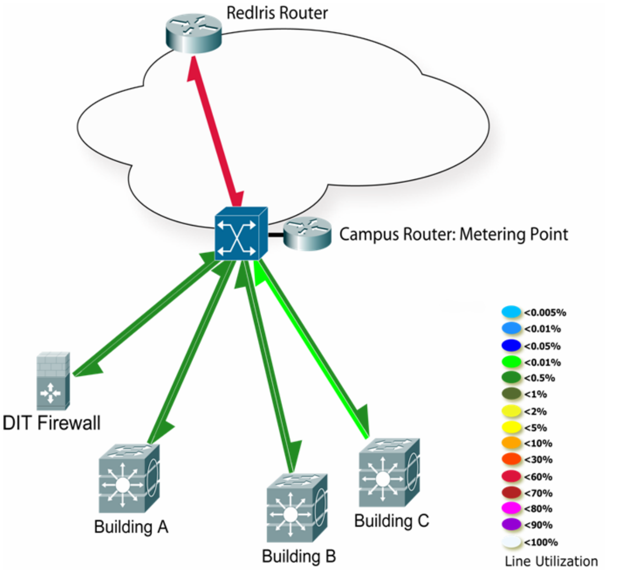

Figure 1 Campus Network

future research activities related to QoS provisioning in IP networks.

ETSIT-UPM Campus Network supports the labs and research activities of different groups providing not only internal connectivity but also Internet access through the Spanish Research Network RedIRIS.

As **Figure 1** shows, the network is composed of several buildings interconnected by means of an optical Gigabit Network, following a multilayer approach based on VLAN's and a central layer 3 routing device (router on a stick configuration).

A. IPFIX Monitoring Architecture

According to IPFIX specifications [4], the monitoring infrastructure relies on the following elements (see **Figure 2**):
IPFIX Device: Campus Router, model CISCO 7201.
Observation Points: VLAN's associated to each IP range.
Metering Process: Netflow.
Exporting Process: Netflow.
Collector: Linux Box.
Collecting Process: flow-tools.

Despite the large amount of traffic being switched on a daily basis, the information exported to the collector is handled with commodity hardware, a dual Intel processor server suffices to retrieve, store and process the data, (around 200 Mbytes daily). Obtaining the same information with packet sniffing tools would require a lot more CPU and storage resources, although the level of granularity could be much higher.

B. Additional Monitoring Modules

No matter how useful flow accounting is on production environments, it lacks the capability to guarantee some critical factors like hardware malfunction or physical link outages. SNMP agents on the contrary, address some of those important functions supporting the Reliability, Availability and Survivability (RAS) of the global system.

One of the most widespread open source software found in network environments is the round robin database tool RRDTOOL [5] which, combined with SNMP, enables an efficient way to store data fetched from SNMP agents. By means of standard MIBS like IF-MIB and proprietary ones like CISCO-PROCESS-MIB it is fairly simple to monitor network device operation on a real-time basis.

C. Information Modules & Applications

To enhance the quality and usability of the information presented to network management staff, the following modules have been developed: (see **Figure 2**)

Traffic Statistics Module: Programmed in Perl, it produces Layer 2 and 3 related information, like number of bytes routed on each VLAN and most accessed ip addresses.

Service Statistics Module: Programmed in Perl, it handles upper Layers information like protocol distribution by organizational units or service distribution by IP prefixes.

Security Module: Still a work in progress, it monitors traffic patterns according to previously defined rule sets to detect possible attacks or worm activities.

Visual Module: Based on XML Front-end software developed by the Apache software foundation group [6], it allows network administrators to have a unified framework of the actual campus network state.

QoS Modules: Still to be completed, its purpose is to report information relative to performance issues, based

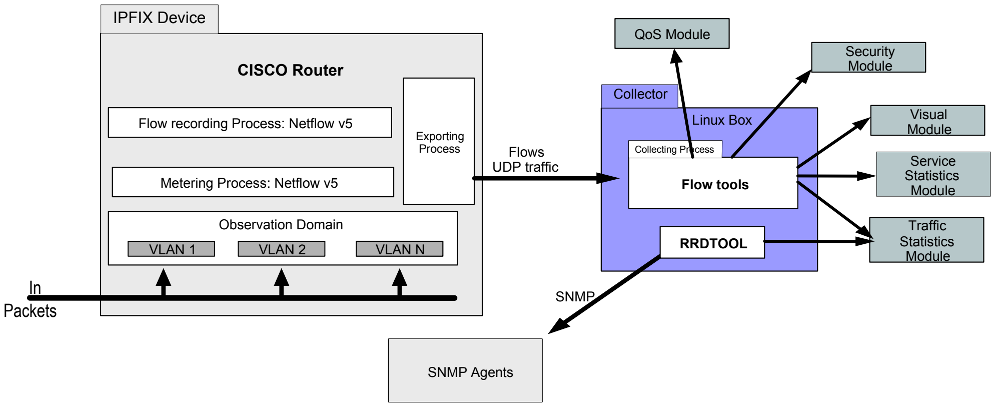

Figure 2: Monitoring Infrastructure

on the IPPM framework.

IV.OBTAINED RESULTS

Based on the above mentioned agents some revealing aspects related to systems and users behaviour are concluded:

- Traffic patterns are highly dependant on the group researching activities: for those groups not closely related to networking research, TCP is the main constituent of the traffic with about 99,5 percent of the overall distribution. On the other hand, groups directly involved with networking reveal a different pattern of protocol distribution: TCP: 63,917 %, GRE: 12,65 %, IPv6 7,305 %, UDP 15,9 %, ICMP 0,193 %.

- One of the motivations for an integrated network management system is to automate as much as possible network troubleshooting and present the information to network managers in a way that can be easily understood. The more visual the applications become, the better the information they provide, especially when this data needs human intervention. For that reason a special emphasis was set into developing graphics oriented software. Although not previously intended, XML has played an important role in the development of visual applications, like the one shown in figure 1, this image, generated in Scalable Vector Graphics format (SVG) informs about the current state of the campus network, leveraging network administration effectiveness.

- A passive method to estimate Bulk transfer capacity metrics according to RFC 3148 [7] has been tested; based on flow timestamps obtained at the IPFIX device it is fairly simple to know the effective bandwidth for that specific connection. As depicted in the following table, several tools are compared, Iperf [8]: an active TCP bandwidth estimation software, Ethereal [9] a commodity sniffer and finally Netflow. The results obtained with the different tools are very similar, though Netflow tends to be more optimistic. One of the main advantages of Netflow estimation is that no extra traffic needs to be produced. On the other hand Iperf floods the link to measure the maximum achievable bandwidth.

Bulk Transfer Capacity Estimation

Kbytes **	Netflow *	Ethereal *	Iperf *
1135	223.549	216	215
16.451	222.953	216	216
5170	207.636	214	205

*** Total Traffic amount*

** Bulk Transfer Capacity Estimation (Kbits/s)*

- An alternative method to infer the response time of the campus services has been developed. By time-stamping the moment at which the first incoming flow was noticed and the time the response left the router we can estimate how long it took the campus network to deliver back the request with a precision on the order of milliseconds. This approach is consistent with RFC 2681 [10], giving useful information related to campus service performance delivery (please refer to **Figure 3**). Questions like: when our services are most heavily loaded, who is accessing these services and how the services are performing can be answered with this type of metrics.

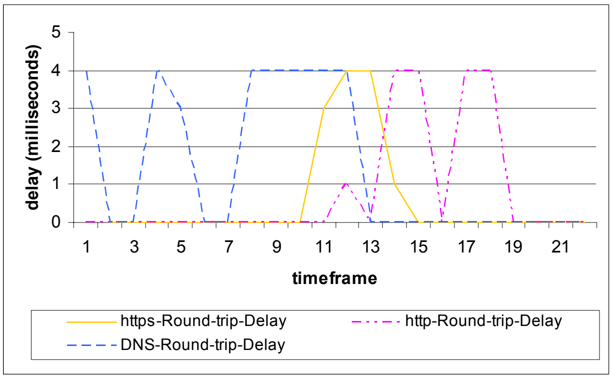

Figure 3 Campus Round Trip delay Metrics

V.CONCLUSIONS AND FURTHER WORK

On the Internet community there is an increasing interest on how to deliver services according to some previously agreed contract (SLA’s). IPPM has set the general framework in which both providers and customers are, by means of some well known metrics, able to reach a quality of service consensus.

At this point, IPFIX efforts are oriented towards the definition of a standard protocol for the retrieval and export of traffic flows. This information will fill the existing gap between raw traffic as seen by the Routers and useful metrics like the ones defined at IPPM.

A management infrastructure is at present time being deployed on a production environment, based on IPFIX and IPPM recommendations, it shall constitute the basis for future experiments related to QoS in IP networks.

Current works include the development of security and QoS modules to detect possible attacks or worm activities, and to calculate relevant IPPM metrics. However, there are some open issues to be accomplished. For instance, flows belonging to secured communications like IPSEC cannot be directly monitored with Netflow, as they encrypt relevant Layer 3 and 4 information.

VI.REFERENCES

[1] IP Flow Information Export (ipfix), IETF.

[2] Cisco Sytems “NetFlow Services and Applications”, White Paper, July 2002.

[3] Simon Leinen “Evaluation of Candidate Protocols for IP Flow Information Export” , Internet Draft, January 2004.

[4] Ganesh Sadasivan, Nevil Brownlee “Architecture Model for IP Flow Information Export”, Internet Draft, October 2003.

[5] T. Oetiker, RRDTOOL,. http://people.ee.ethz.ch/~oetiker/webtool/

[6] Apache software foundation, Forrest, http://xml.apache.org/forrest/

[7] M. Mathis, M. Allman “A Framework for Defining Empirical BTC”, RFC 3148, July 2001.

[8] NLANR, IPERF, http://dast.nlanr.net/Projects/Iperf/

[9] Ethereal, http://www.ethereal.com

[10] G.Almes, S. Kalidindi, M. Zekauskas, A Round-trip Delay Metric for IPPM, RFC 2681, September 1999.