Data on Retention

Ward van Wanrooij and Aiko Pras

University of Twente, PO Box 217, 7500 AE, Enschede, The Netherlands
w.a.h.c.vanwanrooij@student.utwente.nl pras@cs.utwente.nl

Abstract. Proposed EU regulations on data retention could require every provider to keep accounting logs of its customers' Internet usage. Although the technical consequences of these requirements have been investigated by consultancy companies, this paper investigates what this accounting data could be, how it can be obtained and how much data storage is needed. This research shows that every gigabyte of network traffic results in approximately 400 kilobyte of accounting data when using our refinements to existing methods for storing accounting data – less by a factor twenty than previously assumed.

1 Introduction

Recently the discussion regarding the controversial European proposal for a framework decision on the retention of telephone and Internet data [1] stirred emotions across Europe. If enacted this could require every Internet service provider to keep detailed logs of the Internet usage of its customers. Incomprehension about the supposed effectiveness of the proposals and lack of knowledge about the technical implications of the requirements cause an opaque debate dominated by emotions [2] and not by facts.

One major source of confusion appears to be the uneducated guess made by studies regarding the technical consequences [3, 4] of the volume and cost of storing so-called "IP accounting data", a term coined in these papers. The amount of and way to store all other relevant traffic data, namely authentication and e-mail logs, is fairly straightforward; however the report estimates that a typical, large access provider needs to store 72 terabytes of IP accounting data yearly. This paper focuses on the definition of this IP accounting data, the way it can be obtained and making an educated guess towards establishing a relation between network data and IP accounting data. Finally, this data is used to establish the expected technical implications for several providers. Note that this paper only documents research into technical aspects of the storage of accounting data and does not give any indication of the effectiveness or efficiency of the proposal and neither approves nor endorses it.

Although much research has been done on traffic analysis, this paper presents a new challenge in storing specific logical connection information while minimizing storage requirements without sampling packets. The resulting process is based on and validated using reliable, real world data. The study "Storage and bandwidth requirements for passive Internet header traces" [5] is based on sampling; the study by KPMG [4] is based on the extrapolation of one unspecified 2 Mbit connection and likewise unspecified "accounting data" to a 25 Gbit load and Cisco [6] specifies a reduction rate for full "netflow data" of 98.5%.

The results of this paper are not only important to law-makers but also to network managers, both policy-makers and administrators, because the enactment of the proposed law can require serious network infrastructure changes. The information in this paper can be used to determine its impact.

The structure of this paper is as follows. Section 2 phrases the research questions, followed by a discussion of the definition of IP accounting data (Sect. 3). Section 4 addresses the way accounting data can be obtained and the results of this process are commented on in Sect. 5. Finally the outcomes of this research are compared to the conclusions of the aforementioned KPMG paper in our conclusion.

2 Research Questions

The three research questions can be summed up as: what is IP accounting data, how can it be obtained and how much storage capacity is needed. More formally, these are the questions:

1. What is, for the purpose of data retention, the description of IP accounting data?
2. In what ways can accounting data be extracted from network traffic?
3. How much accounting data has to be stored ?

3 What is Accounting Data?

Conceptually, IP accounting data in the context of the data retention initiative has been defined as data pertaining to a connection using "subsets of Internet Protocol numbers" and satisfying one or more of the following criteria [1]:

a Data necessary to trace and identify the source of a communication which includes personal details, contact information and information identifying services subscribed to.
b Data necessary to identify the routing and destination of a communication.
c Data necessary to identify the time and date and duration of a communication.
d Data necessary to identify the telecommunication.
e Data necessary to identify the communication device or what purports to be the device.
f Data necessary to identify the location at the start and throughout the duration of the communication.

Concrete storage proposals based on these requirements for each connection (referred to as communication in the initiative) have been recorded in [7] and include: user, IP addresses, port numbers, date and time and "type of service" (fulfilling conditions a, b and c). For the purpose of this research we assume [3] that all access providers save network authentication logs and consequently are able to map a local IP address to a unique user, based on the combination of local address and time of the connection. We do not expand on this aspect of accounting data in this work.

Now the recordable properties of a connection have been defined, however the definition of a connection has not been. Technically, a connection exists after e.g. a three-way handshake has been made using TCP. For other Internet protocols, like UDP, a similar notion does not exist – therefore a more logical than technical definition has to be used. For this purpose, a modified version of a NetFlow [6] flow has been adapted: *a network flow is defined as a bidirectional stream of packets between a given source and destination within a certain time frame. A flow is defined by its recordable properties.* To avoid confusion, in this document connection refers to a single IP session (TCP connection, UDP message and reply) and flow refers to a stream of packets, possibly spanning multiple connections.

4 How can Accounting Data be Obtained?

Several possibilities exist to obtain and store the required accounting data. Three evident options exist:

1. Capturing packet dumps by configuring a special monitoring device on a network link. Because these dumps contain a copy of all network traffic the size is equal to amount of used network bandwidth. Compressing these files results in an average size reduction of 42% [5], nonetheless practically not manageable due to their size.
2. Capturing packet dumps and saving only the first 68 bytes of each Ethernet frame. Depending on the protocol used, these trimmed packets contain at least the full header containing the necessary accounting data and possibly even some payload. A major advantage over plain packet dumps is the reduction in the amount of stored data, compressed a scaling down of up to 90% may be achieved [5]. Such a reduction still results in a 100 MB file for a continuous loaded 2 Mbit/s link for one hour.
3. Saving NetFlow data. Cisco's NetFlow technology, included in many high-end routers and switching devices, generates a flow record (47 bytes) for each unique connection through the network device. The definition of this NetFlow is a superset of the accounting flow definition of Sect. 3. Because the latest incarnation of NetFlow technology has been selected as the basis for IETF's IPFIX (Internet Protocol Flow Information eXtract) [8] and some other vendors[1] already include NetFlow compatible technology in their

[1] The competitor sflow can currently not be used for gathering accounting data because sflow is based on sampling.

routers and switches [9], the usage of this technology is a viable choice. In conjunction with a NetFlow Collector, it allows for realtime network traffic overview on a relatively small scale. However, Cisco's estimate is that the amount of log data is approximately 1.5% of the network traffic and that is undeniably too much for very large scale deployments like the EU data retention initiative.

In the next subsection we introduce a method of processing the input data, be it a (partial) packet dump or NetFlow log, to a format that minimizes long term storage requirements while still conforming to the requirements of accounting data as set forth in Sect. 3. This method extends on the NetFlow data and principles and is referred to as accounting flows for the remainder of this document.

4.1 Accounting Flows Data Definition

First we establish what traffic data we choose to save based on the definition of accounting data and the most efficient way of storing it.

The source port number is typically semi-random and conveys no information so it can be omitted[2] from stored records. The destination port number (when available) expresses important information about the supposed type of service that has been provided (e.g. TCP/80 means HTTP, UDP/53 means DNS). This information is also the only reasonably reliable indicator about the type of service that can be obtained without inspecting, analyzing and possibly decrypting each packet.

Many other NetFlow fields, like number of packets, number of bytes transferred and AS numbers, are irrelevant to this application and can be dropped.

We need to to store two temporal characteristics for each flow: the start and end date. The proposal prescribes a one second resolution for all time related data, but a full timestamp (e.g. seconds since the epoch) requires 4 bytes, each. We opt to instead file the data in units (files, tables, folders) spanning 12 hours of traffic so we can store 2 byte offsets: the offset since the start of the unit for the start date and the duration for the end date.

Summarized, each accounting flow is an instance of this tuple (15 bytes):

Source address IPv4 address of the host that initiated the flow (4 bytes).

Destination address IPv4 address of the host that received or rejected the flow (4 bytes).

Destination port number Port number of this flow on the destination host (2 bytes).

Flow start Seconds since the epoch when the flow was initiated (2 bytes).

Flow duration Duration of the flow in seconds (2 bytes).

Protocol (1 byte) Protocol number used for this flow (e.g. TCP is 6, UDP is 17, GRE is 47).

[2] In rare cases, the source port number is used for authentication; it can also be used for operation system fingerprinting.

Further minimization of this data can be obtained by using compression techniques (instead of removing information), e.g. dictionary coding the IP addresses.

No special provisions have been made for IPv6 since its use in connecting end-users is negligible at the moment. However, the above tuple layout is also applicable to IPv6 without changing the size: assuming that the number of connections does not increase when IPv6 is deployed then the total practical number of possible source and destination IP addresses also does not change nor does its storage size. Using a translation table for each unit, an IPv6 address can be mapped to a virtual IPv4 address solely used in the unit for storing the data.

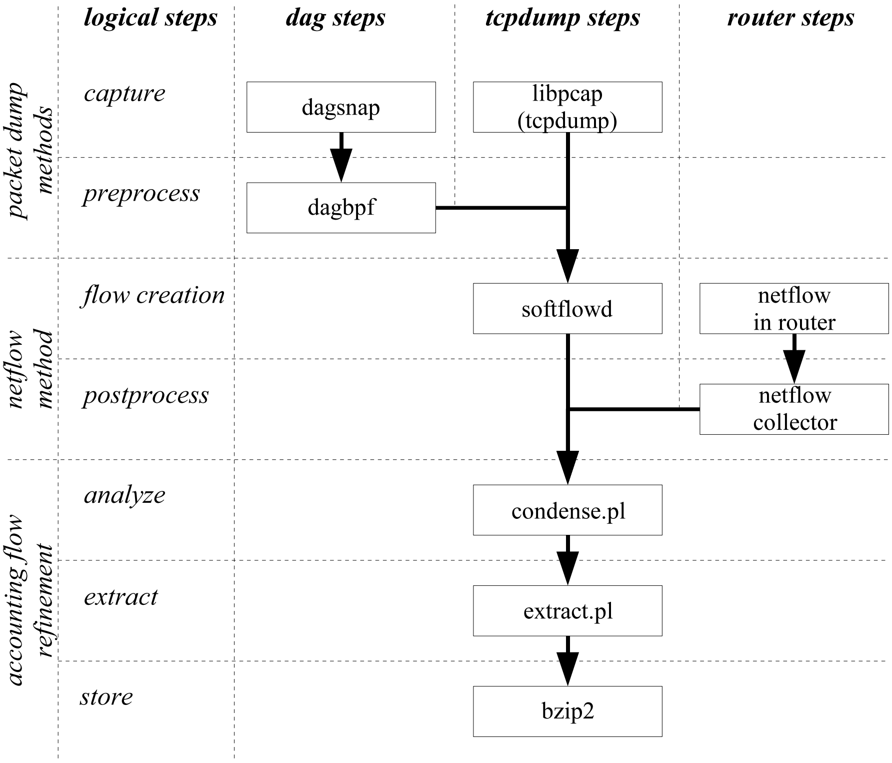


Fig. 1. Steps to obtain accounting flows

4.2 Obtaining Accounting Flows

Having defined our target data, we need to extend the NetFlow process to output this data and allow two types of input data: realtime data from e.g. NetFlow collectors or a monitoring device and offline data to assist us in testing its performance.

If deployed in production use, the input data will most probably be NetFlow logs because this is more efficient, scalable and reliable than using full or partial packet traces.

The procedure to go from network traffic to accounting flows can be characterized as a seven step process (Fig. 1):

Capture The action of capturing the packets, we use dagsnap and libpcap based tools.

Preprocess Processing the captured data to provide suitable input to the next step (e.g. collecting, assembling, converting). In this paper, data captured using DAG [12] cards needs to be converted to libpcap format.

Flow creation Creating flows from the preprocessed data. The algorithm has been specified in [6]: in essence NetFlow groups related packets together in a flow (logical connection) based on, among other things, IP addresses and port numbers. After the end of the connection or a certain period of inactivity the flow is expired and exported. We utilized the package softflowd[3] [10] for the conversion of libpcap dumps. The parameters for flow expiration are 5 minutes of inactivity on a flow or a maximum life of 12 hours or a maximum traffic of 2 gigabytes. NetFlow enabled routers natively export NetFlow data and combines this and the previous two steps.

Post process (optional) Processing the flows to provide storable output. When using NetFlow, the flows are gathered by a NetFlow collector in this step.

Analyze Transforming the netflows into accounting flows. For this paper the step is achieved by running the data through a custom Perl program that removes and regroups data based on the data definition. Its main effect is achieved by removing the source port and grouping tuples of equal properties (source address, destination address, destination port, protocol) together while still considering the other (time, space) constraints.

Extract Extracting the relevant data from the analyzed data and saving it in binary format. We use a simple script to perform these tasks.

Store Storing the extracted accounting flows in compressed form for archival purposes. For this evaluation we make use of the general purpose program bzip2 [11] to pack the data.

5 How much Accounting Data has to be Stored?

To accurately measure the storage requirements to save the accounting data and substantiate the refined method we need representative input data of several types of locations. Further, this input data needs to be available in packet dumps and not NetFlow logs because dumps offer greater insight during development and are more prevalent than NetFlow logs. For our research we have selected

[3] Several patches (file output, report aggregates, distinguish between source and destination of a flow) have been made to enhance softflowd. The complete set is available at http://wwwhome.cs.utwente.nl/~wanrooij/papers/dataonretention/-softflowd.patch

over 30 traces from six different locations (table 1), together representative for most public networks.

Table 1. Samplepoints

name	type	date	#traces	#days	time (hr:mn)	traffic (GB)
m2c-loc1 [13]	student dorms	2002Q2	4	4	01:00	51
m2c-loc3 [13]	college	2003Q3	8	5	02:00	16
m2c-loc4 [13]	broadband homes	2004Q1	15	8	03:45	184
sigcomm-01 [14]	conference	2001Q3	1	3	52:00	4
nzix-II [15]	Internet exchange	2000Q3	3	3	14:30	37
mawi-b [16]	transatlantic link	2005Q1	12	1	03:00	23

These six locations can be classified into three categories of samplepoints:

1. End-users network uplink: m2c-loc4 monitors basic broadband (ADSL, varying speeds from 256 Kbit/s to 8 Mbit/s) connections, m2c-loc1 captures packets from heavy broadband users (Ethernet, 100 Mbit/s), m2c-loc3 is a college allowing 9 to 5 usage of its network by students (comparable to an office setting) and sigcomm-01 is a trace of a wireless 802.11b conference network.
2. Inter-network point: nzix-II is a set of traces of the NZIX when it served as a peering point for six major New Zealand ISPs. These files, captured by a DAG [12] card, need to be converted to libpcap format before use.
3. Intra-network link: mawi-b is a transpacific (Japan - United States) 100 Mbit/s line.

The last category of intra-network link serves as verification whether possible conclusions about the amount of accounting data might also be applicable to other links.

5.1 Results

For each of the categories of Sect. 5 the amount of stored date for each of the methods in Sect. 4 and the new accounting flows refinement has been determined using the steps of Sect. 4.2. These amounts have been visualized in figures 2 and 3; when applicable, charts have separate data series for different times of the day. Along the X axis in Fig. 2 are the different methods: full dump (traffic), header dump (header), netflow, accounting flows (accflow) and the compressed version of accounting flows (bzip2). Finally each graph contains an indicator on the optimality of the accounting flow algorithm: this calculated value is based on the principle that every combination of source and destination address should at least appear once in the flows. Each combination of two addresses that appears more than once in the accounting flows, e.g. due to different port number or expiration, is only present once in this indicator.

Table 2 shows the ratio (compressed accounting flows)/(traffic data) for each of the locations.

Table 2. Unweighed ratio compressed accounting data/traffic

name	category	ratio	name	category	ratio
m2c-loc1	end-user	0.00367%	nzix-II	inter-network	0.04116%
m2c-loc3-night	end-user	0.07485%	sigcomm-01	end-user	0.03670%
m2c-loc3-morning	end-user	0.00892%	mawi-b-03h	intra-network	0.17977&
m2c-loc3-afternoon	end-user	0.03019%	mawi-b-15h	intra-network	0.26824%
m2c-loc4	end-user	0.02646%	mawi-b-21h	intra-network	0.01137%

The four end-user locations share the common impression of a steep descending line in the graphs, but some variation in the resulting ratios still exist. The differences can be explained by looking at the usage of the connection at the respective location:

- Student dorms (Fig. 2(a)): Probably the most notable aspect are the small ratios for m2c-loc1 in comparison with all other locations. The explanation for this phenomenon is straightforward: when connections are stable (same characteristics) and relatively large amounts of data are transferred using these connections then a smaller set of flows is produced than when the connections are used for e.g. browsing or messaging. This location connects student dorms using 100 Mbit/s endpoints: an adequate interpretation of this data, also glancing over the used port numbers and direction of connections, is up and downloading of large files (sharing).
- College (Fig. 2(b)): The traffic dump for this location has been preprocessed before flow creation was attempted due to large inexplicable amounts of ICMP traffic. Because "normal" traffic data does not exhibit this pattern this ICMP data has been filtered out; all figures (e.g. amount of traffic data) have also been corrected for this filtering. The diverging ratios for different times of the day are striking. However, considering the type of location (college, no student dorms) and the observations of m2c-loc1 the explanation is obvious: at night the Internet traffic is limited to e.g. DNS lookups, some mails and some background traffic (all activities resulting in very high ratios); in the morning students come in and start sharing files (low ratio), most of them already left the building by the end of the afternoon (higher ratio).
- Broadband homes (Fig. 2(c)) and sigcomm-01 (Fig. 2(c)) represent typical web activities and share similar results. Sigcomm being a conference engages in more casual, quick browsing and net accessing while m2c-loc4 (ADSL network) has more transfer of files.

The inter-network point shows a declining curve for nzix-II (Fig. 2(d)) that approximately matches the results of the end-users connections. The free fall in the graph near the indicator is caused by the fact that the traces used cover a period of six hours, so expired flows, e.g. POP3 sessions and news websites, are more prevalent than in a short trace. This also applies to the sigcomm-01 trace of several days.

The intra-network samplepoint differs from the others in it being an arbitrary link in a network that connects two large IP networks in different time zones. Because we used parts of a 24 hour long trace and initial analysis revealed wildly varying traffic patterns throughout the day, the graph has been set-up in a different way: the x-axis is now time measurement and the data series are the amount of data of the respective steps (Fig. 3). Because the link primarily connects Japan and USA, time has been marked in local (Tokyo) and Central Standard Time. Upon investigation (destination port, traffic patterns), a lot of virus related activity seems to be going on on the link (scans for targets by worms), especially during the spikes around 03:00 and 16:00. Now the value of the two timelines comes around: when intentional traffic is only unidirectional, because the other side is mostly asleep, traffic can become distorted because worms and viruses don't ever sleep. These malignant programs usually scan complete networks just by sending a few UDP or TCP packets, thereby creating a different flow entry for each target, hence the amount of accounting flows substantially increases per 100 MB traffic. The fact that the second spike is higher than the first one does not conclusively prove anything; this may be caused by:

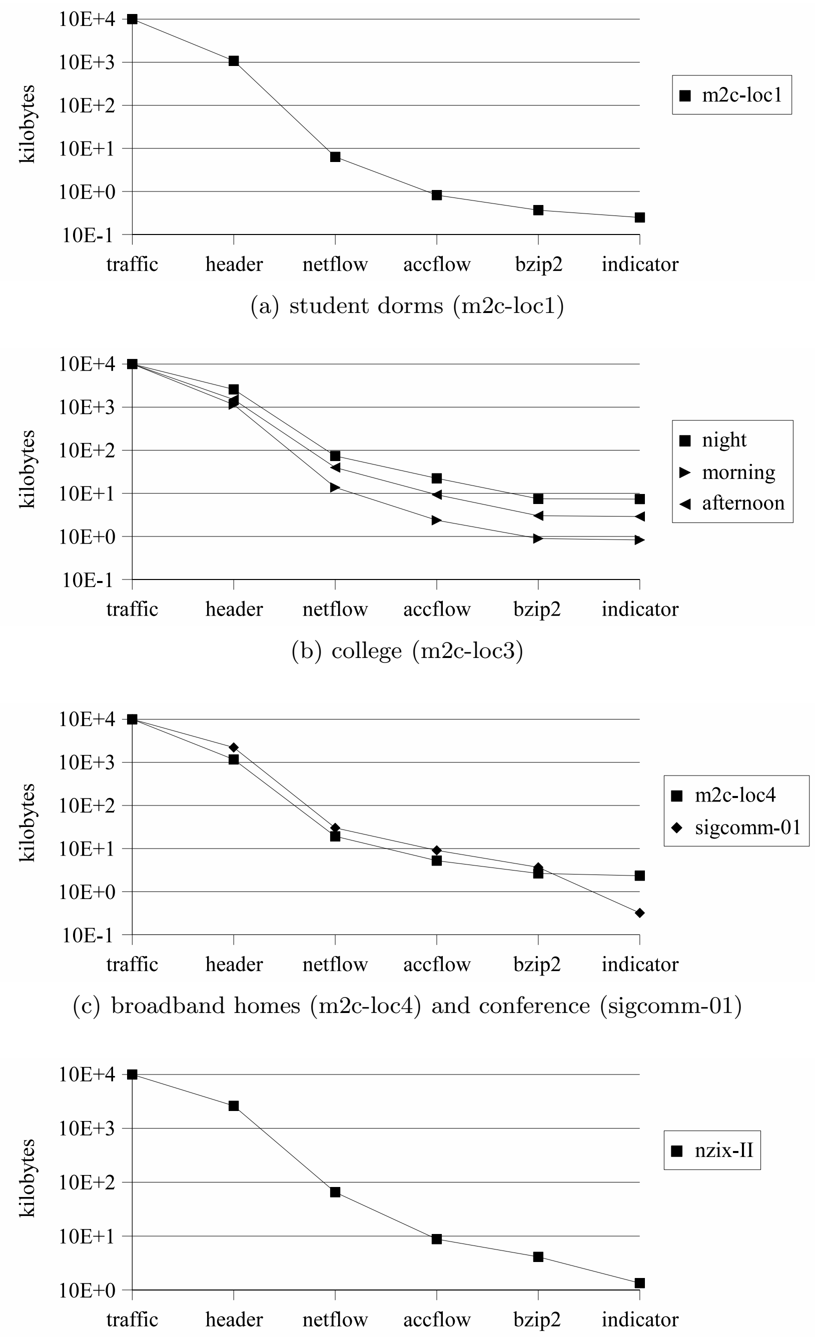
- Larger IPv4 network in Japan reachable through the link (more targets: more traffic).
- Larger IPv4 network in USA reachable through the link (more perpetrators: more traffic).
- Larger percentage of infected workstations in USA
- Different browsing habits in Japan and USA

Aside from these interesting observations, the traffic patterns, although always distorted by worms due to the nature of the link, confirm that our other observations are approximately on target, even though these results are higher. This is easily explained due to the nature of the link: bidirectional, not only connecting end users to servers but also servers and worms to end users.

5.2 Comparison

Based on the results in tables 1 and 2, we can make the cautious, but reasonable estimate for the ratio (compressed accounting data in bytes)/(network traffic in bytes) of 0.04%. This figure is applicable to both traffic on network uplinks as well as traffic on network exchanges; graph 3 established that this figure could also be applied to other links, except for special circumstances.

These results differ from previously published findings by KPMG. This report [4] asserted that one provider stated that one hour of 2 Mbit/s traffic generates 8 megabyte accounting data (ratio of 0.89%). Based on the average throughput



(a) student dorms (m2c-loc1)

(b) college (m2c-loc3)

(c) broadband homes (m2c-loc4) and conference (sigcomm-01)

(d) internet exchange (nzix-II)

Fig. 2. Accounting data per 100 MB traffic

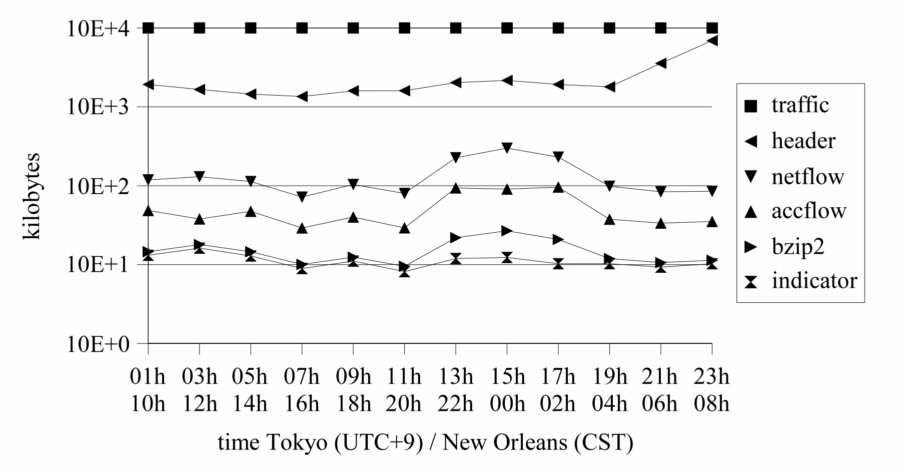


Fig. 3. Accounting data for mawi-b per 100 MB traffic

of AMSIX [17], the major Internet exchange in the Netherlands, of 25 Gbit/s, an estimate of 60 terabyte accounting data per month (0.76%) is reasoned out in the report. Although not specified in the report, we assume that the format for saving data is compressed NetFlow.

Using our refinements to the NetFlow technology for storing accounting flows requires just 0.04% disk space of the original traffic, a twenty fold increase in efficiency. Instead of 60 terabyte of storage, now only an estimated 3 terabyte is required monthly to store all the logs.

6 Conclusion

Motivated by the confusion surrounding the storage requirements of the possible EU regulations on data retention [1], we have investigated the definition of accounting data and ways to extract this data from network traffic.

During our research we have refined the NetFlow method to allow for efficient storage of the information required by the data retention proposal. The testing of these enhancements on several representative data sets shows a reduction in storage data of about 99.96% – equal to approximately 400 kilobyte of accounting flows for every gigabyte of bandwidth used. Previous research into this area by the consultancy company KPMG [4] documents a ratio of about 0.8% percent; this specialized method performs better by a factor twenty. For the Netherlands this represents a reduction of 57 terabyte of total monthly storage needed to comply with the data retention initiative.

The results of this research can and should be used to make the debate on the EU data retention laws more transparent and factually correct. Because the findings on the definition, process and ratios are universal, they can also be used in any debate on or application of data retention – whether in public law by policy makers or private venue by network managers.

References

1. Presidency of Council of the European Union: Draft Framework Decision on the retention of data processed and stored in connection with the provision of publicly available electronic communications services or data on public communications networks for the purpose of prevention, investigation, detection and prosecution of crime and criminal offences including terrorism, Brussels, Belgium, November 2004, http://register.consilium.eu.int/pdf/en/04/st14/st14190.en04.pdf
2. Persson, M., Trommelen, J.: Ten aanval, Volkskrant 12 April 2005, PCM Uitgevers, Amsterdam, The Netherlands, April 2005
3. Stratix: Onderzoek "Bewaren Verkeersgegevens door Telecommunicatieaanbieders", Schiphol, The Netherlands, August 2003, http://www.bof.nl/docs/-stratix_verkeersgegevens_eindrapport.pdf
4. KPMG Information Risk Management: Onderzoek naar de opslag van historische verkeersgegevens van telecommunicatieaanbieders, Amstelveen, The Netherlands, November 2004, http://www.bof.nl/docs/bewaarplicht_KPMG.pdf
5. Micheel, J., Braun, H.-W., Graham, I.: Storage and Bandwidth Requirements for Passive Internet Header Traces, Workshop on Network-Related Data Management, in conjunction with ACM SIGMOD/PODS 2001, Santa Barbara, California, USA, May 2001, http://moat.nlanr.net/Papers/nrdm2001.pdf
6. Cisco: NetFlow Services Solutions Guide, San Jose, USA, October 2001, http://www.cisco.com/univercd/cc/td/doc/cisintwk/intsolns/netflsol/nfwhite.pdf
7. Working party on co-operation on criminal matters: Non paper data retention, Leiden, The Netherlands, September 2004, http://www.bof.nl/docs/non-paper.pdf
8. IETF Secretariat: IP Flow Information Export (ipfix) Charter, May 2005, http://www.ietf.org/html.charters/ipfix-charter.html
9. Kretchmar, J.: Open Source Network Administration, Prentice Hall PTR, Upper Sadle River, New Jersey, USA, September 2003, Section 5.1
10. Miller, D.: Software NetFlow probe, May 2005, http://www.mindrot.org/-softflowd.html
11. Seward, J.: bzip2, May 2005, http://www.bzip.org/
12. Endace Measurement Systems: Network Monitoring Cards, May 2005, http://-www.endace.com/networkMCards.htm
13. Van de Meent, R.: M2C Measurement Data Repository, Enschede, The Netherlands, December 2003, http://arch.cs.utwente.nl/projects/m2c/m2c-D15.pdf
14. Balachandran, A.: Wireless LAN Traces from ACM SIGCOMM'01, San Diego, California, USA, August 2001, http://ramp.ucsd.edu/pawn/sigcomm-trace/
15. WAND Research Group: NLANR MOAT NZIX-II trace archive, May 2005, http://pma.nlanr.net/Traces/long/nzix2.html
16. WIDE MAWI Working Group: MAWI Working Group Traffic Archive, May 2005, http://tracer.csl.sony.co.jp/mawi/
17. AMS-IX B.V.: AMS-IX Homepage, May 2005, http://www.ams-ix.net