Internet Access Traffic Measurement and Analysis

Steffen Gebert[1], Rastin Pries[1], Daniel Schlosser[1], and Klaus Heck[2]

[1] University of Würzburg, Institute of Computer Science, Germany
{steffen.gebert|pries|schlosser}@informatik.uni-wuerzburg.de

[2] Hotzone GmbH
heck@hotzone.de

Abstract. The fast changing application types and their behavior require consecutive measurements of access networks. In this paper, we present the results of a 14-day measurement in an access network connecting 600 users with the Internet. Our application classification reveals a trend back to HTTP traffic, underlines the immense usage of flash videos, and unveils a participant of a Botnet. In addition, flow and user statistics are presented, which resulting traffic models can be used for simulation and emulation of access networks.

Key words: traffic measurement, traffic classification, flow statistics, broadband wireless access network

1 Introduction & Background

As the web evolves, Internet usage patterns change heavily. For the simulation and development of new protocols and routing algorithms, it is important to have accurate and up-to-date Internet traffic models that describe the user behavior and network usage. For network administrators, it is crucial to have knowledge of what is happening in their network. Additionally, consecutive measurements help to identify bottlenecks, network misconfiguration, and also misuse.

A few years ago, Peer-to-Peer (P2P) file sharing was with about 40% the dominating service in the Internet [1]. Since the movie industry is pursuing users of file sharing software distributing copyright protected content, file download sites like RapidShare started attracting more people. Besides file download sites, YouTube's popularity accounts for an additional enormous amount of traffic that is transferred using HTTP. These observations were already made by Wamser et al. [1] back in 2008.

In this paper, we evaluate the same access network as in [1]. However, in the meantime, the observed network has grown and more than doubled its size, connecting about 600 users with the Internet. The increased size of the network required a new design of our measurement architecture to be able to collect and process the increased amount of data in real-time. We evaluate whether the high share of P2P still applies or if any other trend can be observed. In contrast to the measurements published in [1], we also evaluate flow statistics and present a user statistic.

The rest of this paper is organized as follows. Section 2 presents the work related to this paper. In Section 3, we describe the observed network and the measurement scenario. Section 4 shows the results of the 14 days lasting measurement. Finally, conclusions are drawn in Section 5.

2 Related Work

An analysis of 20,000 residential Digital Subscriber Line (DSL) customers is described by Maier et al. [2], which took place in 2008/09. Analyzing DSL session information, available through the ISP's RADIUS authentication server, revealed an average duration of 20-30 min per online-session. Observing the transfer times of IP packets exposes that most of the delay, in median 46 ms, is created between the user's device and the DSL Access Multiplexer (DSLAM), compared to 17 ms to transfer it to the connection end point (mostly a server accessed through the core network). HTTP dominated the traffic volume with a share of almost 60%, compared to 14% of P2P traffic. Furthermore, a distribution of the different MIME types of HTTP transfers is presented.

Classifying flows by only looking at connection patterns between hosts (host behavior-based classification) is done by BLINC [3]. By setting up connection graphs, so called *graphlets*, and matching observed traffic against them, a categorization of single hosts into different classes is made.

Szabó et al. [4] applied multiple classification methods to minimize the amount of unknown traffic and to compare the results of the different methods. The outcome was that every classification type and engine has its strengths and weaknesses. The final decision takes the different methods into account. If no majority decision can be made, classification results from the methods with the highest precision (signature- or port-based) are preferred over results achieved with vague heuristics.

Finamore et al. [5, 6] present their experience of developing and using their traffic measurement software *Tstat* during the past 10 years. From the technical perspective, they highlight design goals of measurement software coping with Gigabit speed and present statistics, where hardware capture cards cause an immense release of CPU resources.[3] By combining several classification techniques, including different DPI variants and a behavioral classifier, classification rates of more than 80% of data value were reached. Outgoing from four vantage points at end-customer ISPs with up to 15,000 connected users located in Italy, Poland, and Hungary they demonstrate that application usage is subject to variation by the observed country. Presented statistics range from May 2009 to December of 2010, thus also cover our measurement time frame of June/July 2010. At this time, in their measurement, eMule file sharing accounts for 15-35% of total Italian traffic, its share in Poland was around 1-2%, and not even visible in the Hungarian statistics. In contrast, Bittorrent was very popular in Hungary with around 35% traffic share respectively 20% in Poland. The majority of Polish traffic was observed to be caused by HTTP-based traffic, proportioned into each 15% of video streaming and downloads of file hosting services and 40% of the total volume general HTTP traffic. In contrast, its share in Italian and Hungarian is only half as high with 30-40%. At this particular time in June 2010 they observe - contrary to previous trends - a turn in P2P traffic share, which started increasing again.

Our study fits very well to the work of Finamore et al. [5], which presents data of other European countries, while we now provide numbers for German Internet usage. In addition to general application statistics, we focus on flow characteristics in the second part of this paper.

[3] According to the figures from 100% load to less than 10% at a link utilization of 300 Mbps

3 Measurement Scenario & Methodology

In this section, we describe the measurement scenario as well as our measurement framework with its software components, which used during this evaluation.

3.1 Setup

In cooperation with a German broadband wireless access provider, we observed an access network to which about 600 households are connected. The network connects two housing complexes with the Internet through another carrier ISP. Complex 1 provides wireless access for about 250 households through numerous Cisco Access Points. Complex 2 provides wired access for about 350 households using the in-house LAN installation. Complex 2 is connected to Complex 1 using an 802.11n WiFi link.

The connection to the Internet carrier is established using another 802.11n WiFi link, which is limited to 30 Mbps per direction. Although this seems to be a quite slow connection, it is according to the provider completely sufficient to support all 600 households. All three links are connected in an exchange network, located in Complex 1. Figure 1 illustrates the established links and the position of our measurement point in the exchange network, right before the WiFi link to the carrier.

Besides artificial limitation of the WiFi link to the carrier, a class-based traffic shaping from Cisco [7] is applied. Traffic shaping can limit the bandwidth available for different services. Contrary to dropping packets, shaping limits the bit rate more softly. Through the implementation of a token bucket filter, packets are sent only at a specified maximum rate. Downside is an increased delay, as packets have to wait for some time, in times of non empty queues. When the incoming rate is below the shaping limit, no delay is added, as packets are forwarded immediately. Queued shaping can only limit

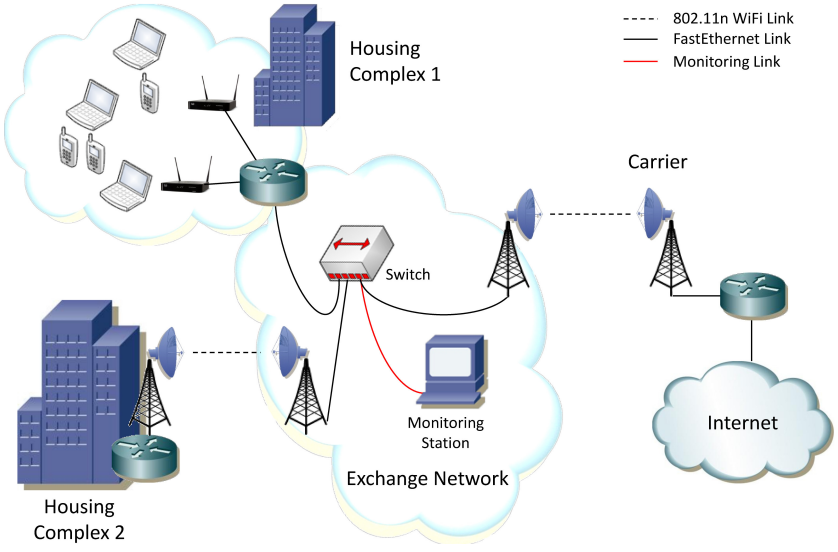


Fig. 1. Network and measurement setup.

the average rate, not the peak rate. If the queue is empty, the rate can be way higher for a short time frame, until the buffer is filled up.

The ISP's shaping configuration is as follows. Based on a white-list, traffic is added to a *high-priority class*. This class is allowed to consume up to 22 Mbps per direction and includes the most frequently used and "known-good" protocols like HTTP, HTTPS, email, SSH, Skype, or video streaming applications. They are identified using Cisco's content filtering engine. Furthermore, on user request, ports are added to the high-priority list, to cope with limitations of the classification engine. All other protocols that are not identified to be high-priority, are put into a *default class*. This especially matches P2P file sharing software like Bittorrent. The default class is limited to an average bandwidth of 1 Mbps per direction. This way, excessive use of file sharing and negative influence on real-time services is prevented according to the provider's opinion. However, during our measurements, we observed that the limitation with Cisco's content filtering engine does not work properly.

Clients get public IP addresses with a short DHCP lease timeout of 30 minutes assigned. Therefore, IPs are no valid identifier to track a particular user, as a reassignment of the address after an inactivity of at least 30 minutes is very likely. As the ISP firewall allows incoming connections, people are directly exposed to the Internet and able to run publicly reachable services.

3.2 Measurement Software

Our measurements are done using our custom measurement tool PALM, which is targeted to run on commodity hardware. PALM stands for *PAcket Level Measurements* and is designed to generate packet and flow level statistics. Although packet level statistics are available, we focus in this publication on flow level statistics. We designed PALM in such a way that a throughput rate of at least 50 Mbps symmetric with an average packet rate of more than 50k packets per second can be measured, classified, and stored.

We could have used community-oriented measurement software such as TIE [8], but decided to write our own software because we were able to reuse parts of our previous software applied in [1].

Figure 2 depicts the architecture of PALM. Its three major components *Collector*, *Processor*, and *Store* are separated into different threads for use of multiple CPU cores. All components are highly decoupled to stay flexible for exchanging single implementations and for further expansion, e.g. through additional classification engines.

Packet processing starts with a *Collector*, which reads packets either from the network interface or a PCAP file. We make use of the *SharpPcap* library, which provides

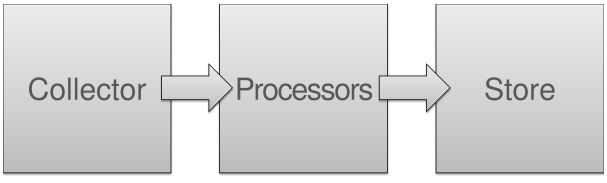


Fig. 2. Measurement software architecture.

a C# interface to the *libpcap* library. In the next step, several *Processors* are executed on a per packet basis, from which we elaborate only on the most important ones.

OpenDPI: This processor makes use of OpenDPI, an open-source Deep Packet Inspection (DPI) library, for classification of packets. Instead of enforcing a classification rate of near to 100% bringing the risk of an immense number of false classifications, we prefer a more defensive strategy and relinquish vague heuristics. Table 1 lists the most important applications and their arrangement in different categories.

Table 1. Application categories of the packet classification.

Category	Protocol
HTTP	HTTP
	HTTPS
	DirectDownloadLink
IM	Jabber
	Yahoo
	ICQ
	MSN
	IRC
MAIL	IMAP
	POP3
	SMTP
P2P	Bittorrent
	Gnutella
	PANDO
MULTIMEDIA	Flash
	RTP
	SHOUTcast
	PPStream
	MPEG
	Windowsmedia
OTHER	IPSec
	DHCP
	SNMP
	NETBIOS
	NTP
	SSH
	DNS
	ICMP
UNKNOWN	*unknown*

IpAnonymizer: Very important while dealing with user data is anonymization. We run a basic *IpAnonymizer*, which scrambles IP addresses. In case of higher demands, e.g. prefix-preserving anonymization, we think of extending PALM by making use of external libraries like *Crypto-PAn* or *PktAnon* [9].

FlowManager: Through this processor, packets having the same quintuple (source/destination IP/port and protocol) are aggregated into flows. We treat a flow as unidirectional data transfer. Thus, in most cases an opponent flow with exchanged source/destination IP/port exists. As suggested by CAIDA, we use an inactive-timeout of 64 seconds for flow termination [10]. By executing flow aging every 60 seconds in a background thread, longer inactivity of a flow (up to 124 seconds) could prevent termination of the first and starting of a second one with the same quintuple and merge them to one flow. However, we do not count this as a disadvantage for our measurements.

After real-time analysis and anonymization, packet and flow data is saved by a *Store*. We made good experience with our *MySQL* store, which is able to save data at a very high rate of more than 100k records (packets or flows) per second by using bulk inserts[4]. The

[4] Bulk inserts can be done by using the *LOAD DATA INFILE* command

advantage of using a Relational Database Management System (RDBMS) like MySQL is that we are able to directly gain statistics by executing SQL queries, also via software like Matlab[5], without any intermediate conversion steps.

3.3 Measurement Duration

Our measurements took place in June and July of 2010. For the evaluation presented in this work, we use data of 14 days. Table 2 lists statistics about the collected data.

Table 2. Measurement statistics.

Beginning of measurement	2010-06-28
End of measurement	2010-07-12
Number of packets	4,946,071,829
Number of flows	202,429,622
Amount of data observed	3.297 TB
Number of households (acc. to ISP)	ca. 600
Carrier link speed	30 Mbps

4 Measurement Results

In the following, we present our measurement results retrieved from a 14 days measurement period. During these two weeks, 3.3 TByte were observed. This section is divided into daytime statistics, application usage, flow statistics, and usage statistics.

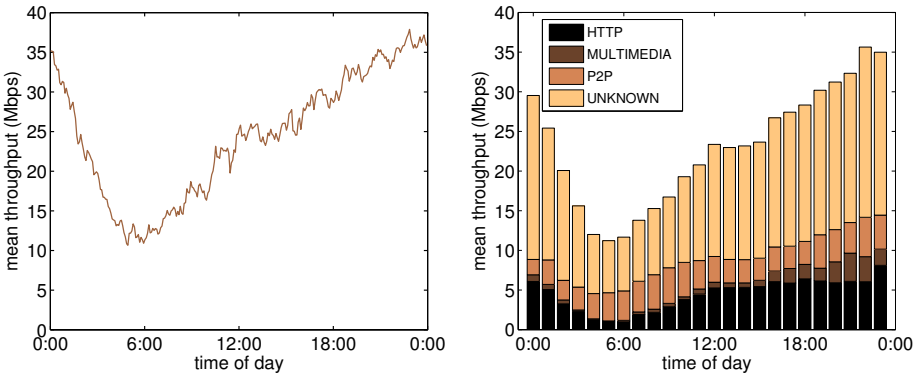
4.1 Daytime Statistics

We start with investigating the variation of Internet usage during the day. Observed traffic patterns dependent on the ISP's customers type, such as home or business users.

The network, which we are observing, mostly supplies smaller households. Figure 3(a) shows the daytime utilization of the network link in a 5-minute interval averaged over the 14 days of measurement. The minimum usage is observed around 5:00 to 6:00 in the morning, when most people are sleeping. The throughput increases steadily over the whole day and reaches its maximum at about 22:00. Such observations of a three- to fourfold throughput during late hours compared to early mornings are perfectly in line with the values reported for home users in [6].

To get an insight from which applications the traffic is originating, Figure 3(b) presents the daily traffic fluctuation, grouped by application type, omitting less important categories. Peer-to-Peer applications cause an almost constant bit rate of 3-4 Mbps. This reveals two facts: On the one hand, the ISP's traffic shaping is not working that perfect. This can be caused either by P2P applications using ports prioritized by the traffic shaping and thus circumventing the 1 Mbps per direction limit, or the configured

[5] Using the *mYm* extension for Matlab



(a) Total network throughput. (b) Throughput per application category.

Fig. 3. Mean daily traffic fluctuations.

”average” limit of Cisco’s shaping functionality is really very inaccurate. The second fact shown by this statistics is that users of file sharing software often do not switch off their computers during night.

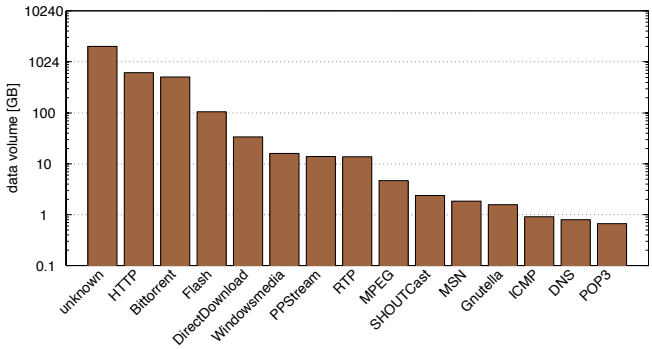
Contrary to P2P traffic, HTTP is heavily influenced by the day-night fluctuation. Except for large file downloads, HTTP is interactive traffic, caused by the user accessing web pages. In line with the expectations, the minimum usage is observed between 5:00 and 6:00, when only a few people are actively using their PCs. HTTP throughput increases steadily between 7:00 and 12:00 from below 1 Mbps to 5-6 Mbps. For the rest of the day, the throughput stays almost constant. An absolute maximum is reached in the hour between 23:00 and 0:00, where its average traffic is at 8.1 Mbps. The factor between the throughput at 6:00 in the morning and during the day at 12:00 and 18:00 is 1:5.4 respectively 1:6.5. The characteristic that HTTP traffic is mostly interactive traffic plays a major role in the day-night fluctuation of the overall network usage.

Streaming and video download traffic is even more subject to variation. Again, the minimum usage is observed at 5:00. The utilization increases, until its maximum is reached at 20:00. Especially in the evening, more intense use of streaming is observed. The throughput relations between 0.18 Mbps at 6:00 compared to 0.7 Mbps at 12:00 and 1.78 Mbps at 18:00 are even more diverse than of HTTP traffic. As pointed out, HTTP stagnates about noon. Increased usage of multimedia applications is mainly responsible for the further increase of the network throughput during the rest of the day.

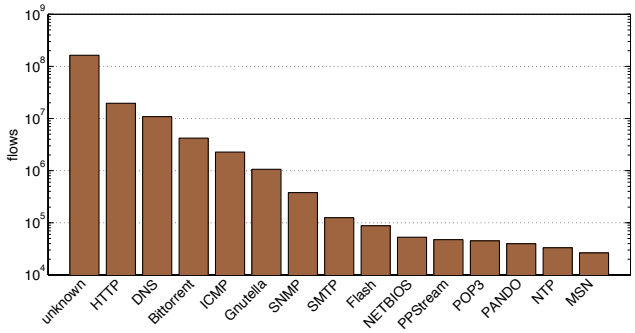
4.2 Application Usage

We continue with presenting the top-15 protocols regarding the amount of transferred data and number of flows in Figure 4. The y-axis show a logarithmic scale.

We observe HTTP traffic as the protocol that is used for most data transfers with a share of 18.4%. OpenDPI can return more fine-grained results for some protocols using HTTP as transport, like Flash or downloads of Windowsmedia (wmv) and MPEG video files. It also comes with detection rules for the most popular DirectDownload hosters, like Rapidshare. Putting them altogether and counting all HTTP transfers increases the share to 22.6%, which makes HTTP the dominating protocol.



(a) Total data volume per application.



(b) Total number of flows per application.

Fig. 4. Total data volume and number of flows per application.

While [6] reports 12-15% traffic share of video streaming for Italian and Hungarian customers and 20% for Polish users, we observe with 3.0% a lot less. The share of Flash on total HTTP traffic is for us with 17% also smaller compared to their observations. Furthermore, [6] reports an increase of 10% of Flash traffic starting from January to July of 2010, which since then stagnates. Very likely, this trend is shifted by a few months for the German users that which we observed.

The second most seen protocol is the P2P file sharing protocol Bittorrent. As explained in Section 3.1, Bittorrent traffic is throttled in the observed network. Although P2P file sharing usage decreased in favor of DirectDownload hosters over the last years [1], the share of 14.8% is immense. On average, more than 35 GB per day are Bittorrent data transfers. Compared to values for other European countries [6], our German users show similar usage characteristics as users from Poland. However, as mentioned before, traffic is throttled in the observed network and also very likely a portion of the unidentified traffic is caused by P2P software, which we well discuss in following.

The observed amount of flows classified as *unknown* is tremendous. OpenDPI was not able to classify 80.1% of the flows, responsible for 64% of the traffic volume. More than 46% of those flows have no inverse flow, which gives a few options of guessing their origin, and will be discussed in following. First, if the connection uses TCP, these flows must have been unsuccessful connection attempts. As the first SYN packet contains no payload, it can not be identified by our signature-based classification. In total,

8.5% of the unsuccessful classified flows without an inverse flow are failed TCP connection attempts. UDP uses no connection establishment and the first packet already contains payload data. It is possible that the one-way UDP connection was arranged using another control connection or was defined by configuration. *SNMP Traps* or other monitoring data are examples, where no feedback is expected and data is sent without receiving any response packets. However, this is very unlikely in the Internet and only possible for sending data to servers, in case of clients behind NAT gateways.

Thus, we assume that most one-way conversations are failing connection attempts. These are most likely P2P applications, which try to find other peers, but fail due to the peer having already disconnected, or the packet being blocked by the target host's firewall. We measure that 91.6% of those one-way UDP flows consist of only one single packet, which proves the theory of being mostly connection attempts. Altogether 98.7% contain less than five packets and even 99.6% of the flows consist of less than ten packets. This leads to the conclusion that almost no serious one-way connections are used. We conclude that the main part of the mentioned 46% of all flows, which have been classified as unknown, is caused by failed connection attempts, which our Deep Packet Inspection is not able to identify. The remaining share of unsuccessfully classified flows are most likely caused by P2P applications and encrypted traffic.
Further exploration of *unknown* traffic based on port numbers reveals the following:

- Traffic originating from TCP port 80 (HTTP) causes the major share of 57.7% of the unknown traffic amount. As we found no obvious mistakes in OpenDPI's rule set, we assume that this is non-HTTP traffic tunneled through port 80. As pointed out in Section 3.1, the ISP's traffic shaping prioritizes traffic based on white-listed ports. We evaluated traffic tunneled through port 80 being classified as non-HTTP protocol and e.g. found 820 MB of Bittorrent and 1.3 GB of MSN messenger traffic.
- Port 13838, both TCP and UDP, is used by exactly one customer IP address for incoming connections during the two weeks. While the UDP flows account for 1.33% of unknown flows and 1.60% of unknown bytes, the TCP flows account for only 0.13% of unknown flows, but 6.13% of unknown traffic. On average, port 13838 TCP flows are 10 times bigger than the UDP flows.We have the feeling that this is a P2P application that uses UDP for signaling and TCP for data transfers.
- TCP Port 443 (HTTPS) accounts for 1.90% of unknown bytes and 0.69% of flows. Although encrypted, SSL connections can be identified during the handshake. OpenDPI ships with a signature, however this either does not support all SSL versions or, again, some applications use the prioritized port 443 for tunneling other traffic.
- Except these ports, we found only TCP ports 554 and 182 as port numbers below 10,000 in the 20 port numbers responsible for most of the unknown traffic. Port 182 is registered port for *Unisys Audit SITP*, 554 for *kshell* (Kerberos). Thus, we assume most of the unidentified traffic being P2P transfers using dynamic port numbers.

We stick to this as explanation for the high amount of unknown traffic and use the numbers gained by successful classifications from OpenDPI.

Besides OpenDPI, we also implemented and evaluated a *L7-filter* Deep-Packet-Inspection processor, which could help lower the amount of unknown traffic. However, we experienced an immense slowdown of the classification by a factor of 5-10. In order to reach our performance goal, we sticked to solely OpenDPI for this measurement.

Another thing, which attracted our attention, was the number of SMTP flows: One outgoing SMTP flow usually means sending one email. Observing 125,000 SMTP flows during the 14 days would mean that each of the 600 users writes more than seven mails per day. As many people prefer Webmail interfaces or use Facebook for their communication, this number seems way too high - given the fact that we the clients are mostly smaller households. Calculating the number of SMTP flows reveals that one IP address is responsible for 98,150 of these 125,000 flows, distributed over the the whole measurement time. This immense number of email sending attempts must be caused by a virus infection of this particular host. It is very likely that this host is part of a Botnet and acts as a sender of spam emails.

4.3 Flow Sizes per Application

In Figure 5, cumulative distribution functions (CDFs) are shown for the most frequently observed categories. OTHER combines all successfully classified flows together that do not fit into one of the other categories. DNS traffic, which is associated with this category, is responsible for 18.7% of all flows and mainly consists of one packet per flow. DNS is one of the reasons for this category containing more smaller flows, so called *mice*, than other categories. Almost 42% of the flows are less than 100 bytes in size and more than 81% are below 310 bytes.

The left part of the HTTP flow size distribution is caused by upstream flows, while the right part, containing the larger flows, is typically caused by downstream flows. In contrast to downstream flows containing the real payload data, upstream flows mostly contain only the HTTP request header and TCP acknowledgments (ACKs). The number of ACKs depend on the retrieved payload size. The size of the request header is influenced by the used browser, transferred cookie data, and some other information, including the length of the requested file name. Thus, the distribution starts very smooth, compared to the other application protocol categories, which show more discrete steps.

4.4 Concurrent Flows

The number of concurrent active flows has an influence on the required processing power of network devices operating on flow level. Besides of statistics collection, e.g.

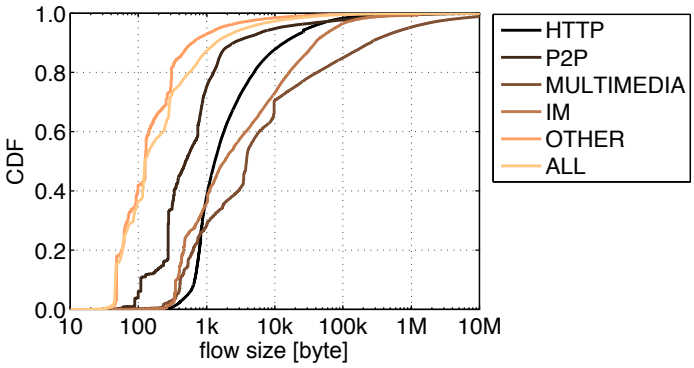
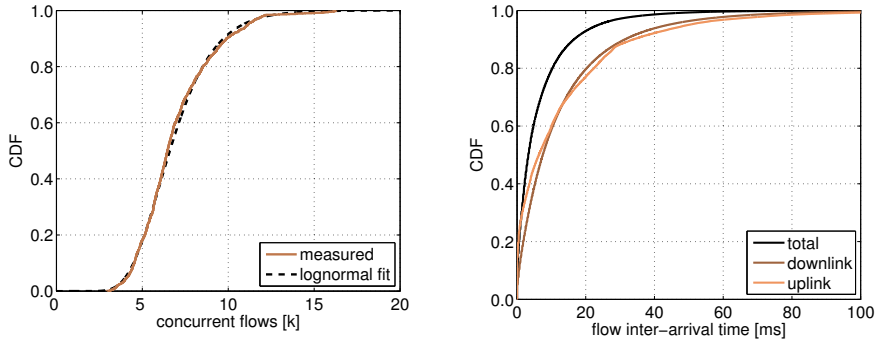


Fig. 5. Flow size distribution per application category.



(a) Distribution of concurrent flows.

(b) Distribution of flow inter-arrival times.

Fig. 6. Distributions of concurrent flows and flow inter-arrival times.

like on a NetFlow probe, the number of concurrent active flows gets an interesting measure with recent work on flow-based routing or switching, like OpenFlow [11].

For this evaluation, we calculate the number of concurrent active flows once every minute during one day. Figure 6(a) shows the resulting CDF. In 90% of the time, between 4,100 and 11,200 flows are simultaneously active. During 9.9% of the observed points of time, 10,000 and more concurrent flows are observed, in 0.9% even more than 15,000. The maximum number of concurrent flows is 16,290. The lognormal distribution with parameters $\mu = 8.79473,\ \sigma = 0.302074$ fits the measured distribution with a mean of 6907.45 concurrent flows best.

4.5 Flow Inter-Arrival Times

For the same day, we present the distribution of flow inter-arrival times (IAT) in Figure 6(b), both for the total traffic and split into uplink/downlink direction.

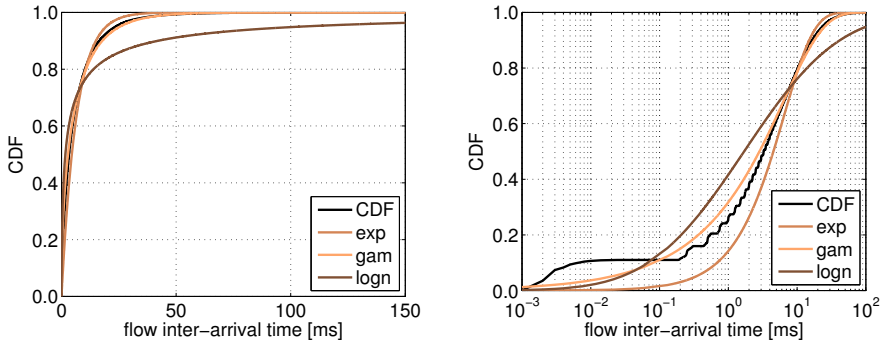
On average the next flow arrives within the next 3.23 ms. For egress traffic, a new flow is created every 6.3 ms and every 7.5 ms for ingress traffic. An inter-arrival time of less than 10 ms is observed for 60.0% of egress, and 59.0% of ingress traffic. For IATs of combined traffic directions, new flows arrive in less than 10 ms in 79.4% of all cases. In less than 10% of the cases, new flows arrive with delays higher than 34.0 ms for upstream, 31.5 ms for downstream, and 16.5 ms for combined traffic. These values are again useful as input for simulations of an access network or while dimensioning devices operating on a per-flow level, like the mentioned OpenFlow devices.

In order to figure out, which distribution the overall flow inter-arrival times are following, we fitted the CDFs with the exponential, gamma, and lognormal distributions. Table 3 shows them together with the optimal parameters. While the lognormal distribution does not reflect the observed IAT distribution very well, the exponential distribution and especially the gamma distribution show an accurate fit.

The exponential distribution function seems to fit pretty well according to Figure 7(a). However, looking at the differences using a logarithmic x-scale in Figure 7(b) reveals that especially in the smaller time distances, its failure is way larger than the mean error of the gamma distribution. As with a probability of 50%, the time distance

Table 3. Applied distribution functions for flow inter-arrival times (in ms).

Distribution	Parameters	Mean	Variance
Exponential	$\lambda = 6.5277$	6.5277	42.6107
Lognormal	$\mu = 0.5295,\ \sigma = 2.5058$	39.2175	818811
Gamma	$k = 0.4754,\ \theta = 13.7300$	6.5277	89.6254
Measurement		6.5277	87.0107



(a) Linear scale. (b) Logarithmic scale.

Fig. 7. Fitting of flow inter-arrival times distribution.

between two new flows is below 3.3 ms, a larger failure of the fitting distribution for smaller IATs makes the disadvantage of the exponential distribution even more appealing. Using a gamma distribution to estimate flow arrivals is also preferred by [12].

4.6 Internet Usage Statistics

As the ISP has a very short DHCP lease times of only 30 minutes configured, it is hardly possible to track after switching their PCs off and on. To get an impression of the client's traffic consumption, we evaluate logging data of the ISP's billing system. Figure 8 shows the distribution of the amount of traffic caused by the users during one month, separated into upstream, downstream and total traffic.

After a steep increase until about 2 GB transfer volume, the curve flattens heavily. This is the transition from users, accessing the Internet mainly for web browsing, email, and very likely social networking, towards the user category, making intense use of video or audio streaming sites and services. Already 63.5% of the users cause a traffic amount of more than 10 GB per month, 17.5% more than 40 GB. Within this region is the fluent transition towards the group of power users, making excessive use of downloading, file sharing, video streaming, and other bandwidth consuming applications. A monthly transfer volume of more than 100 GB is observed for 4% of the ISP's users. The largest observed volume was 135 GB.

An average traffic volume of 12 GB for German households, respectively 19 GB for US households, in 2009 was reported by [13]. The mostly single households, which we observed, make more intense use of their Internet connection and cause a higher traffic

Table 4. Gamma distribution parameters for modeling of monthly user traffic consumption.

Direction	Parameters	Mean
Downstream	$k = 0.622127\ \theta = 29.4298$	18.3091 GB
Upstream	$k = 0.391863\ \theta = 13.1583$	5.15626 GB
Combined	$k = 0.589397\ \theta = 39.8124$	23.4653 GB

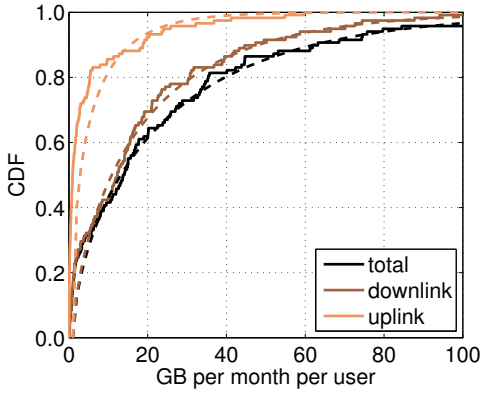


Fig. 8. Total traffic consumption per user.

volume. We measured an average traffic volume of 21.9 GB per month per user. Table 4 lists the parameters of the gamma distribution, which best fits the user statistics.

5 Conclusion

In this paper, we presented the results of a 14 days Internet access network measurement. In contrast to our previous publication [1], we had to completely redesign the measurement infrastructure to be able to cope with the increased size of the ISP. A major finding is that HTTP is the dominant protocol used in today's access networks and has displaced Peer-to-Peer file sharing protocols. One reason for this is the popularity of file hosting sites, which emerged after the movie industry started pursuing users of P2P file sharing software. This trend is underlined by our flow statistics, showing some very large HTTP flows. The only still dominant P2P protocol is Bittorrent, with a share of about 38%, neglecting not classified flows. Flash video traffic is with 8% still the most widespread video container format, but we expect that a decrease with the discontinued development of mobile flash and the trend towards HTML5 video streaming.

Looking at the number of concurrent flows and flow inter-arrival times, network planners have to be aware of how to cope with it as the Internet usage increases drastically. Our results show that on average 22 GB of data is transferred per user per month compared to 12 GB per month measured only one year before. Class-based traffic shaping like used by our provider have to be able to handle this short flow inter-arrival times.

For future research, we want to initiate new measurements, as the provider now offers IPTV, which results in a lot more traffic per user and a changed application distribution.

Acknowledgments

The authors would gratefully thank Antonio Pescapé from the University of Napoli and Phuoc Tran-Gia from the University of Wuerzburg for the fruitful discussions and support on this paper.

References

1. Pries, R., Wamser, F., Staehle, D., Heck, K., Tran-Gia, P.: Traffic Measurement and Analysis of a Broadband Wireless Internet Access. In: IEEE VTC Spring 09, Barcelona, Spain (2009)
2. Maier, G., Feldmann, A., Paxson, V., Allman, M.: On Dominant Characteristics of Residential Broadband Internet Traffic. In: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference. IMC '09, New York, NY, USA, ACM (2009) 90–102
3. Karagiannis, T., Papagiannaki, K., Faloutsos, M.: BLINC: Multilevel Traffic Classification in the Dark. SIGCOMM Comput. Commun. Rev. **35** (2005) 229–240
4. Szabó, G., Szabó, I., Orincsay, D.: Accurate traffic classification. In: Proceedings of IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks. WoWMoM (2007)
5. Finamore, A., Mellia, M., Meo, M., Munafo, M.M., Rossi, D.: Experiences of Internet traffic monitoring with Tstat. **25**(3) (2011) 8–14
6. García-Dorado, J., Finamore, A., Mellia, M., Meo, M., Munafò, M.M.: Characterization of ISP Traffic: Trends, User Habits and Access Technology Impact. Technical Report TR-091811, Telecommunication Networks Group - Politecnico di Torino (2011)
7. Cisco Systems: Regulating Packet Flow on a Per-Class Basis - Using Class-Based Traffic Shaping. Cisco IOS Quality of Service Solutions Configuration Guide. (2010)
8. Dainotti, A., de Donato, W., Pescapé, A.: TIE: A Community-Oriented Traffic Classification Platform. In: International Workshop on Traffic Monitoring and Analysis (TMA'09) @ IFIP Networking 2009, Aachen, Germany (2009) 64–74
9. Gamer, T., Mayer, C., Schöller, M., Gamer, T., Sciences, C.: PktAnon - A Generic Framework for Profile-based Traffic Anonymization. PIK Praxis der Informationsverarbeitung und Kommunikation **2** (2008) 67–81
10. CAIDA: Preliminary measurement specification for internet routers (2004)
11. McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., Peterson, L., Rexford, J., Shenker, S., Turner, J.: OpenFlow: Enabling Innovation in Campus Networks. SIGCOMM Comput. Commun. Rev. **38** (2008) 69–74
12. Loiseau, P., Gonçalves, P., Primet Vicat-Blanc, P.: A Comparative Study of Different Heavy Tail Index Estimators of the Flow Size from Sampled Data. In: MetroGrid Workshop, GridNets, New York, USA, ACM Press (2007)
13. Economist Intelligence Unit and IBM Institute for Business Value: Digital economy rankings 2010 - Beyond e-readiness (2010)