A Packet-level Characterization of Network Traffic

Alberto Dainotti, Antonio Pescapé, and Giorgio Ventre
University of Napoli "Federico II" (Italy), {alberto,pescape,giorgio}@unina.it

Abstract— **In this paper we show results from a packet-level traffic characterization aiming at finding spatial and temporal invariances of TCP based applications, such as HTTP and SMTP. We developed a methodology and a software architecture to build packet-level statistical characterization of network traffic based on large traces captured from Wide Area Networks. In order to show the efficacy of the proposed packet-level approach, we applied our methodology to the traffic generated by applications running over HTTP - a typology of traffic that has been extensively studied in literature even if, as far as we know, no accurate packet-level characterization has been proposed as yet with regard to it - and to SMTP traffic. We analyzed traffic from high-speed access links of two different networks and, contrary to common beliefs, the results show properties of spatial and temporal invariance. The study of SMTP has been proposed to demonstrate the generalization of a packet-level approach. Indeed, results prove the general applicability of the methodology and, at the same time, how - also at packet level - different traffic shows different characterizations. Characterization results can be used in platforms for traffic simulation (like *ns* or *ssfnet*) and traffic generation (like *D-ITG* or *MGEN*). Tools and traces at the basis of this work are publicly available at http://www.grid.unina.it/Traffic .**

I. Introduction

Internet Traffic measurement and characterization is an important and essential task in order to understand and solve performance-related issues of current and future networks. In these years, many efforts have been focused on statistically characterizing traffic generated by sources and related to specific application-level protocols, also with the purpose to conduct realistic network traffic simulations. Network traffic can be viewed at different abstraction levels: (i) session; (ii) conversation; (iii) connection/flow; (iv) packet; (v) byte. In this paper we focus our attention on the packet level. Packet-level traffic characterizations express traffic flows in terms of *inter-packet time* and *packet size*. Simpler and more general than higher level models, apart from allowing to analyze network traffic from a different perspective, they have the important advantage that their application in traffic simulation and generation lets us measure and study network parameters like delay, jitter, packet loss, packet corruption, etc. independently from future protocols evolution. For these reasons (and to fill a gap present in literature) we investigate the possibility of constructing realistic and reliable packet-level characterizations from empirical traffic traces. More precisely, we propose a general methodology and a software platform along with the preliminary results we obtained analyzing different kinds of traffic. In this paper we present some results on HTTP and SMTP traffic. HTTP has been extensively studied in literature and several HTTP traffic models have been proposed, but no accurate packet-level characterization has ever been presented before. For this reason it seems to be the best candidate to show both the features of our approach and the results from a packet-level characterization. A preliminary SMTP traffic characterization has been proposed as well both to show how the methodology is simply generalizable and how - also at packet level - distinct behaviors of different traffic can be captured. We analyze very large traffic traces captured from two WAN access links of a University and a research center, different in load, user population, and user practices. Results look very promising and seem to partially contradict common beliefs on packet-level approaches that have been expressed in past literature, in that they show characteristics of invariance (spatial and temporal). Furthermore they offer the opportunity to propose a first insight on how some characteristics of the application-level protocol and user behavior can directly affect network traffic at a lower level.

The rest of the paper is organized as follows. Section II presents related works, highlighting differences with the approach here presented and justifying our work. In Section III we describe our measurement approach, giving details on traffic traces and used tools. In Section IV we expose our characterization methodology, explaining what variables we studied and showing and analyzing the results of our work. We also give some details on how we filtered traffic traces to study only HTTP traffic and we show preliminary results of the analysis of SMTP traffic. Finally, in Section V we draw conclusions and foresee future works.

II. Motivation and Related Work

While packet-level approaches have been used in the past to study traffic generated by Telnet, FTP, NNTP [1] [2], multiplayer network games [3] [4], and multimedia traffic [5], they were not adopted to analyze the characteristics of applications like HTTP, SMTP, or Peer-to-Peer (P2P) traffic. As for HTTP, the only study present at this time at packet level is [6], where a characterization related to the traffic of only a single PC in an Ethernet LAN is presented. In most of the works on HTTP traffic analysis (see Section IV-B.3) it is said that a packet-level approach is not taken into consideration because of its dependency from network conditions and end-to-end flow control, a thesis advocated in [7] and originally expressed in [1] about applications characterized by bulk transfers. This is true over time scales smaller than typical round-trip-times (RTT). But it is also true that over time scales greater than RTTs a packet-level approach can provide results that present both spatial and temporal independence features. Results on SMTP definitely confirm our belief. As for SMTP in [8] a message-level characterization is presented.

0-7803-9536-0/06/$20.00 ©2006 IEEE.

They found that the e-mail trend follows user activity during the day and that the processes of arrivals and departures at servers are Poissonian in nature. In [9], in the context of the *ns* simulator, the authors propose an email traffic characterization based on SMTP connection arrivals and bytes transferred per SMTP connection. Therefore, again, also for SMTP a packet-level characterization had been not proposed up to now.

In this work we propose an approach based on the decomposition of aggregate traffic into *conversations* and on packet-level traffic characterization of traffic inside *conversations*, that is, models based mainly on two variables: packet size (PS) and inter-packet time (IPT). There are several important advantages in such approach: (I) we do not need to make any assumptions regarding the kind of applications generating traffic, (e.g. when we concentrate on HTTP we do not necessarily need to assume traffic is Web traffic), (II) the same methodology is easily extensible to study other application-level protocols and mixes of them (i.e. we give results on SMTP too), (III) packet-level models are simple and easily applicable to traffic simulations that can be used to study network-related issues (measuring delay, jitter, packet loss etc.) or to test network equipment.

We applied this approach looking for parameters that are invariant with respect to time (*time invariance*) and to the observed network (*space invariance*), believing that sampling a higly heterogeneous and large population of clients and servers could make our results partially independent from both network conditions and end-to-end congestion control. In other words, we based the validation of our approach on the concept of the *search for invariants* adopted and stressed in [7]. Indeed we analyzed traffic from two different stub networks with about 1 million of unique client-server couples and 1 billion packets observed. We compared results from traces of different days and from both sites and they showed to be highly invariant. Preliminary results from other sites confirm this result. Finally, in modeling network traffic it has been found that correlation structures must be considered [10]. This aspect is partially out of the scope of this paper. In [11], stepping from results presented here, we propose a traffic model where we analyze the correlation structure of the traffic traces and characteristics at different time scales. Anyway, in this paper we show how spatial and temporal invariances are maintained for the first order statistics.

III. Traffic Traces, Network Scenario and Tools

a) ***Network Scenario and Traffic Traces****:* Results have been obtained by passively monitoring traffic from two stub networks of academic and research Italian institutions during the period *Jan-Dec 2004*. More precisely, we worked on the networks of (i) *Area della Ricerca di Genova of the Italian National Research Council* and (ii) *University of Napoli "Federico II"*, see Fig. 1. For brevity, we will refer to such sites respectively as ARIGE and UNINA. The observed links represent the only connection of the networks to the Internet and have a maximum throughput of $200Mbps$ and $16Mbps$ respectively.

We divided our traffic traces in weekly samples. Because of the regularity we found in traffic and because of space constraints, in this paper we present results related to two different weeks of the considered period, *June 14-19, 2004* for UNINA and *October 4-9, 2004* for ARIGE. The observation of the links resulted in two traffic traces for each node: a trace related to the traffic generated by clients inside the observed networks reaching HTTP and SMTP servers on the rest of the Internet - we will call such traffic *Egress* - and the other one related to external clients visiting HTTP and SMTP servers hosted at the observed networks - we will call it *Ingress* traffic. As the typology of offered services highly influences network traffic characteristics, it is clear that Egress and Ingress traffic are different in terms of generalization properties. In this work we will only report results related to Egress traffic, which represents the vast majority of the observed traffic and is related to a large population of users reaching all kinds of services. Ingress traffic, on the contrary, is strongly related to the limited typology of services offered by UNINA and ARIGE. Furthermore, as it will be clear in Section IV, we modeled traffic exchanged between couples made of a single client and a single server. For this reason, we filtered out from Egress HTTP traffic all the packets flowing through HTTP proxy servers in service in the observed networks, in order to avoid the error of classifying packets generated by different clients as belonging to a single one. This was not necessary in the case of SMTP.

b) ***Tools****:* The analysis was performed on TCP packets with source or destination port 80 (HTTP) and 25 (SMTP).In order to preserve privacy, for each packet we kept only the IP and TCP headers and we scrambled IP addresses using the wide-tcpdpriv tool from the MAWI-WIDE project [12]. As regards HTTP traffic we also captured the first 3 bytes of payload data exchanged between each host pair (which under normal conditions correspond to the *method* invoked by the client in a HTTP request, see Section IV-B.1.c for more details). Traffic was captured and analyzed using Plab [13], a software platform written in C, partially based on the Libpcap library [14], and freely available at [21]. Plab was planned and written to analyze traffic traces by identifying and storing data related to thousands of *conversations* (see Section IV-A) between hosts and to calculate and dump data ready to be processed by statistical analysis software. Also, more intelligence was introduced into the software, as the ability to decode optional TCP headers as the MSS (Section IV-B.1.b), or to filter packets or entire *conversations* based on several criteria (Sections IV-A and IV-B.1.c). The Plab architecture

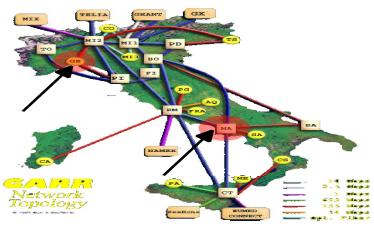

Fig. 1. ARIGE and UNINA nodes in the Italian Research Network

is multi-platform. Our application ran on a FreeBSD 4.x machine at the UNINA site and Linux 2.4.x at the ARIGE site. In the first case, the Berkeley Packet Filter (BPF) kernel driver reported a packet loss of 1 packet out of 10 millions, whereas at the ARIGE site no packet loss was reported by the kernel driver. The packet timestamp resolution provided by the Libpcap library, and by the kernel drivers that it links to, is of $1\mu s$. However, there are several side effects that can affect packet timestamping at kernel level, as buffering on the Ethernet board [15]. For this reason we choose to store and process inter-packet times with a resolution of $10\mu s$.

*c) **Statistical Methodology**:* A statistical analysis of the measured samples in the collected real traffic traces has been provided by setting up a methodology that integrates well-known established techniques separately found in different works, such as distribution estimation, statistical fitting and study of the tails. To perform the statistical fitting, for both IPT and PS, we used the λ^2 discrepancy measure [16] to choose among several well-known analytical distributions [17]. In the choice of the bin width, which is also fundamental to construct density histograms, we used Scott's rule (width w, given by: $w = 3.5 \cdot \sigma \cdot N^{-1/3}$ where σ is the standard deviation) as a base reference, increasing bin width few times when needed [18]. Also, when necessary to improve fitting results, the distributions have been either split in few parts to be separately approximated, or alternatively we adopted the Expectation-Maximization algorithm (EM) applied to parameter estimation of Gaussian Mixtures [19]. Once we obtained analytical distributions, we compared them to the empirical ones by means of graphical representations of the corresponding PDFs, CDFs, and Quantile-Quantile Plot. As regards IPTs, because of the large time scale of samples, ranging from $10\mu s$ to $900s$, we applied a logarithmic transformation to them before analysis. This was necessary to apply an appropriate binning without underestimating the model behavior for small scales and to visualize CDFs and PDFs in a manner more representative of the underlying set. From the analytical models found in the log-transformed domain, it is easy to derive the analytical expressions of the distributions into the original domain. Finally, we also investigated the presence of heavy tails in distributions by estimating the slope in the Log-Log Complementary CDF plot [20]. Statistical software tools developed to obtain the presented results are freely available at [21].

IV. Experimental results

A. Characterization Approach

At the base of our characterization approach there is the decomposition of network traffic into *conversations*, a concept previously introduced in literature [22], defined as the time interval during which two different hosts exchange packets belonging to an association of the kind {*source address, destination address, application-level protocol*}, and separated by a fixed amount of time of silence. We defined as belonging to the same conversation, all packets to and from port TCP 80 (for HTTP) and port TCP 25 (for SMTP), traveling between two hosts, with an inactivity timeout of 15 minutes. As regards HTTP, it is worth to note that such a value is compatible with the HTTP session timeout introduced in past papers related to Web traffic [23] and the distribution of human thinking time modeled by Mah in [24]. They indeed represent an upper bound, being referred to the traffic exchanged between a single client and multiple servers instead of only a single server.

Our approach does not take into account single TCP connections but considers all traffic happening during the conversation as a unique bi-directional flow of data, which we divided into *upstream*, which is traffic from the client to the server (packets with destination port 80 and 25 for HTTP and SMTP respectively) and *downstream*, traffic from the server to the client (the same condition as above but applied to the source port). Both for HTTP and SMTP we separately studied upstream and downstream traffic, building, for each of them, estimates of packet size and inter-packet time distributions. We want to stress that we calculate IPTs by subtracting the timestamps of two consecutive packets belonging to the same conversation and flowing in the same direction (upstream or downstream). An important aspect of our methodology is that in the evaluation of such distributions we did not take into account packets with empty payload. Since we wanted to characterize the traffic generated by the applications, independent as much as possible of TCP itself, we decided to drop all TCP-specific traffic, such as connection establishment packets (SYN-ACK-SYNACK) and pure acknowledgment packets [22]. For the same reason, in the estimation of packet size we measured the byte length of TCP payload. These architectural choices make our results reusable for simulation purposes as an input for TCP state machines, like in D-ITG [25] and TCPlib [1].

B. HTTP Traffic Characterization

Details on HTTP traffic traces are reported in Table I. We started studying observed links with a quantitative analysis of traffic. The three diagrams in Fig. 2 show rates for packets, bytes and opening of new conversations during all the week (from Monday to Saturday) at the UNINA site and related to Egress traffic. Packet and byte rate are separated into upstream and downstream. The data has been sampled with a step of 15 minutes. We note the first characteristic of invariance,

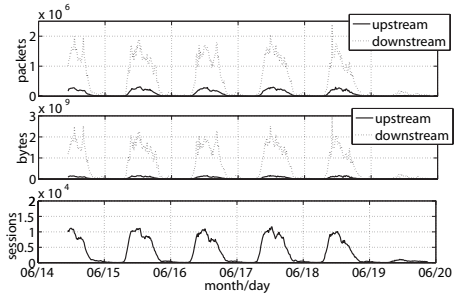

Fig. 2. UNINA packet/byte/conversation rate

diurnal patterns of activity, which is a known phenomenon in network traffic. Indeed Egress traffic is mainly limited to working hours (9:00-17:00). Back to Fig. 2, it is interesting also to observe that bytes, packets and conversations tend

TABLE I

DETAILS OF OBSERVED SITES AND TRAFFIC TRACES

Site	Max Bandwidth	IPs	Date	Size	Pkts	Client-Server pairs	Conversations
UNINA	200Mbps	65000	14-19 Jul 2004	60 GB	830M	1M	2.3M
ARIGE	16Mbps	8000	4-10 Oct 2004	3 GB	43M	56000	125000

to follow the same behavior keeping a proportion among them. Downstream traffic is several times higher than upstream traffic, both for packets (5:1 ratio) and bytes (15:1 ratio). Different ratios for packets and bytes anticipate what we will see later studying packet size distributions: upstream packets are usually smaller than downstream packets. All quantitative values observed at the ARIGE site are in accordance with those from UNINA keeping the same proportions among them but scaled by a factor of 20.

*1) **A packet-level characterization**:* As explained earlier we are interested in accurately characterizing traffic behavior at packet level inside each conversation between a HTTP client and a server. The periodical patterns in traffic induced us to study payload size and IPT distributions separately for each day, from Monday to Friday. For each of the modeled variables (upstream/downstream – payload/inter-packet time), we found almost identical empirical distributions when comparing results for all the days of the week (an example for upstream inter-packet times is shown in Fig. 3). This is an

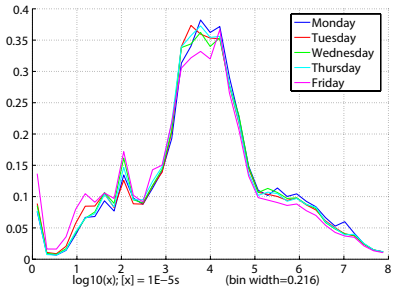

Fig. 3. UNINA upstream inter-packet times PDFs

important result of this study, because it proves time-invariant properties of the characterizations found. For each site we then built average distributions that were representative of the entire considered week. We applied the statistical fitting methodology shortly explained in Section III to such empirical distributions to come up with analytical representations of the sample sets. In Table II the results of statistical fitting related to inter-packet times are summarized. The weekly average empirical distributions were also used to compare PDF and CDF estimates of corresponding variables at each site. In the following, estimates of the distributions of upstream and downstream inter-packet time and payload size, found at both sites, are compared. It can be seen that all the modeled variables show strong properties of invariance when the observed link changes. This is another fundamental finding of our work, which, together with the time-invariant characteristics cited above, makes the obtained results promising in terms of generalization capability and therefore applicable in realistic traffic generation and simulation. However, there are also some differences in the obtained models. Both analogies and dissimilarities, along with an analysis of the empirical and analytical models we obtained, are briefly exposed in the following paragraphs.

*a) **Inter-packet times**:* To properly read the plots in Fig. 4 we remind that packet timestamps have been collected with a precision of $10\mu s$ and that we applied a log_{10} transformation to the samples. Furthermore, all values reported in Table II are referred to the log-transformed domain. We can ideally distinguish among three main overlapping regions in the inter-packet time distribution graphs. The lowest region, which can be roughly considered as starting from the first decade up to half of the third decade, is dominated by back-to-back packets probably due to file-transfers. The next region, which extends until about $1s$, contains samples which are compatible with RTTs found in Wide Area Networks. Finally, consistently with past works [26] [24] [23], we can assume that inter-arrivals directly generated by human behavior (e.g. user clicks) should be located in the last region, that is, beyond $1s$. Such scheme to read inter-packet time distributions has been cross-validated by several considerations and it can be observed that distributions have sudden changes in the transitions between one region and another. As we will also see for payload-size distributions, upstream and downstream traffic behave quite differently. Much of the samples in the upstream traffic is concentrated in the central region, while the majority of downstream inter-packet times are located in the first one. Indeed, while file transfers are the common task of HTTP servers, clients more often issue lot of requests. In the case of Web, for example, the first request of an HTML document is typically followed by more requests for the embedded objects. If such objects are small enough to be sent within one or few packets (as often is the case [27]), requests are sent with intervals close to the RTT from the client to the server. This, not only justifies the predominance of the central region in the upstream distributions, but also explains its presence in the downstream distribution along with a correlation between upstream and downstream inter-packet times. Such correlation can also be seen by looking at the results of statistical fitting, where Lognormal distributions with close *shape* parameters are used to approximate the samples greater than $3ms$ both for upstream and downstream traffic. By plotting the Log-Log

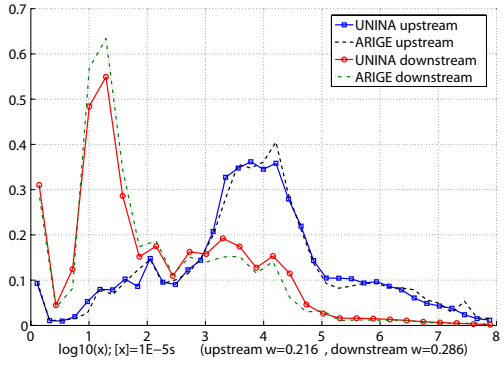

Fig. 4. PDF of UNINA and ARIGE inter-packet times

Complementary CDF of the inter-packet time distributions (see as an example Fig. 5) and estimating the slope of a linear fit (broken red line) to the upper part of the tail we verified that it declined slower than the Exponential function, thus such distributions tend to be heavy-tailed.

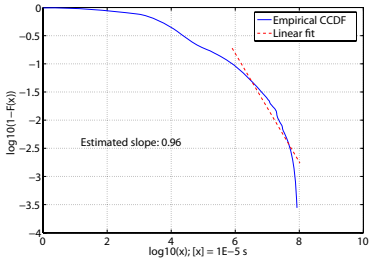

Fig. 5. UNINA upstream inter-packet time Log-Log CCDF

b) ***Payload size****:* Looking at Fig. 6 (in which CDFs are shown instead of PDFs, being that the diagrams are easier to be compared in this case), we can see that corresponding distributions from ARIGE and UNINA networks look very similar. Even though, we noticed that the Maximum Transmission Unit (MTU) of the network interfaces on the hosts, which limits the size of packets that can be sent and received, has a partial effect on them. Indeed, we found several spikes in the distributions. To confirm the hypothesis that they were not related to specific application or user behaviors, but were caused by MTUs set on the interfaces, we instructed Plab to decode, when present, the Maximum Segment Size (MSS) TCP option in SYN segments. We were then able to build separate payload distributions for conversations in which peers negotiated a different MSS (based on the smallest MTU on their interfaces). An analysis of such distributions, and a comparison against the original global distribution revealed that the peaks in the latter were generated by packets with maximum payload from conversations in which the MSS of one of the peers limited the size of the payload. Even if the four most frequent negotiated MSS (1460, 1380, 512, 536 bytes) were found to be associated to more than 95% of the total packets, with the first two covering more than 90%, their distribution was not totally invariant from site to site. Indeed, a larger percentage of conversations with MSS=1380 bytes is the reason of the slight difference between UNINA and ARIGE downstream CDFs.

As we anticipated, upstream and downstream packets have different payload-size distributions. In general, upstream payloads tend to be smaller, with a mean value of 500 bytes; average downstream payload size is 1240 bytes. The downstream distribution is basically dominated by full-length packets. We identify several of the above-mentioned peaks, which together sum up to the 80% of the distribution. Most of them are concentrated near 500 bytes and near 1400 bytes. The remaining 20% of the packets is uniformly distributed over the total range. If we remind that we discarded packets without payload, we see that such distribution is consistent with the trimodal distribution of generic IP packet size found in wide-area traffic studies [28]. For upstream traffic we found totally different results. 85% of the samples reside in a set starting from 120 bytes to 1280 bytes. The spikes present in this region when a frequency histogram is plotted, disappeared when we adopted a bin size of twenty bytes and the distribution could be statistically fitted with a Lognormal distribution of $scale$ 6 and $shape$ 0.4. Smaller payloads represent a 3% of the total and are fitted by an Exponential. The remaining larger samples are due to full-length packets limited by MSS and can be fitted with deterministic and uniform distributions.

TABLE II

SUMMARY OF FITTING FOR INTER-PACKET TIMES

Traffic	Site	Part	Range	Percentage	Distribution	Parameters		λ^2
UP	UNINA	I	0-0.6	3.34%	Gamma	0.22	0.30	9.1E-2
		II	0.6-1.6	6.61%	Extreme	1.33	0.21	8.6E-2
		III	1.6-2.5	9.32%	Weibull	2.16	9.69	7.3E-2
		IV	2.4-max	81.61%	Lognormal	1.43	0.25	6.7E-2
	ARIGE	I	0-0.6	2.53%	Gamma	0.19	0.37	2.4E-1
		II	0.6-1.6	4.7%	Extreme	1.35	0.17	1.2E-1
		III	1.6-2.5	10.6%	Rayleigh	1.44		1.2E-1
		IV	2.4-max	83.16%	Lognormal	1.44	0.26	1.2E-1
DW	UNINA	I	0-0.5	9.73%	Gamma	0.36	0.21	8.7E-2
		II	0.5-2.7	54.64%	Lognormal	0.31	0.33	1.1E-1
		III	2.7-max	35.63%	Lognormal	1.33	0.21	2.1E-1
	ARIGE	I	0-0.6	9.53%	Gamma	0.22	0.34	1.8E-1
		II	0.6-2.7	61.48%	Lognormal	0.34	0.29	2.1E-1
		III	2.4-max	32.26%	Lognormal	1.28	0.24	1.6E-1

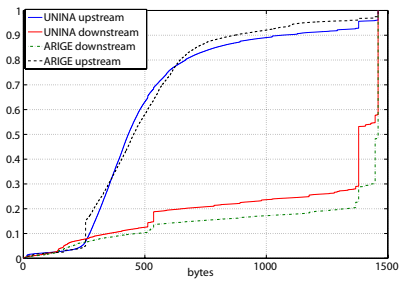

Fig. 6. Upstream and downstream payload size CDFs

c) ***Filtering out non-HTTP traffic****:* To stress the concept of spatial invariance at packet level, in this Section we show how a process of trace sanitization applied to the UNINA traffic makes the traffic profile more regular and more similar to the profile of the ARIGE network traffic. As a side effect, we give some insights about non-HTTP traffic using port TCP 80. Indeed, we know that few applications are sometimes configured to use port TCP 80 to bypass filters on firewalls. Especially P2P applications [29]. For this reason, to enforce network usage policies, network administrators sometimes must resort to application-level firewalls or to more advanced filtering architectures. Latest firewalls and routers, indeed, are moving toward the use of packet inspection, often done in hardware, to enforce traffic filtering by application classification; e.g. the NBAR feature in Cisco routers [30]. An NBAR filter was indeed in action on the UNINA site, which blocked all communications with hosts outside the network using FastTrack and Gnutella protocols. This prevents some popular P2P applications (Kazaa, Grokster, Limewire, Morpheus etc.) to exchange traffic. At the ARIGE site there were no such kind of filters instead. This network, though, is used by a more homogeneous and restricted group of users, mainly researchers, than the large UNINA network. In the latter, different categories of people (e.g., students) have access to the network. Before starting our analysis, we were still concerned with the presence of non-HTTP traffic

in our traces, so we investigated it further. We added a feature in Plab to examine the first 3 bytes of the first packet carrying TCP payload exchanged in each conversation. As reported in Table III, we observed that almost 94% of the conversations started with a GET request, 4% with a POST request etc. Only a small fraction of the sessions presented packets starting with a byte not corresponding to an alphabetic character. Inside this category, 99% of the conversations started with the byte 0xe3. As reported in [31], this is the first byte exchanged by peers opening a communication session based on the eDonkey2000 protocol (used by eDonkey and eMule file-sharing applications). Because our interest was in

TABLE III

PAYLOAD INSPECTION ON FIRST PACKET OPENING A CONVERSATION

Conversation Start	GET	POS	HEA	Downstream	0xe3	PRO
Percentage	93.94	4.23	0.7	0.44	0.27	0.2

characterizing traffic generated only by applications running over HTTP, we instructed Plab to recognize such conversations and to filter them out. Also, 0.44% of the conversations were initiated by the host communicating from port TCP 80 (labeled as "downstream" in Table III). By filtering our traces, we observed that 5.12% of the processed packets were discarded. Therefore, this non-HTTP traffic represents a not negligible portion of the captured traffic. As regards the number of filtered conversations, they account for about 0.7% of the total. This suggests that filtered conversations tend to generate more packets than HTTP conversations. By comparing the results obtained with and without filtering such conversations, we observed that discarded traffic had a consistent impact in terms of payload size and inter-packet time. Comparisons of the obtained distributions for upstream traffic at the UNINA site are shown in Fig. 7. Observing the properties of such

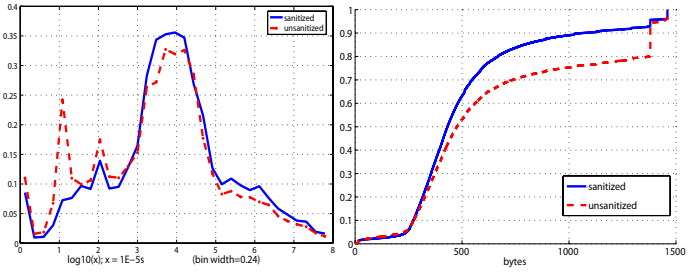

Fig. 7. Filtered UNINA upstream: IPT PDF (left), PS CDF (right)

distributions it is clear that filtered conversations increase the portion of back-to-back packets with full payload, probably due to the presence of file-transfers.

*2) **Conversation-level**:* As said earlier, this work is based on the concept of conversation. To fully understand the traffic dynamics at packet level and to have more evidences regarding results on conversations found after the sanitization stage, in this Section we study the conversations behavior and we link it to dynamics of inter-packet times. For both sites we studied conversation arrivals and duration times. The empirical distribution of conversations durations is quite irregular and

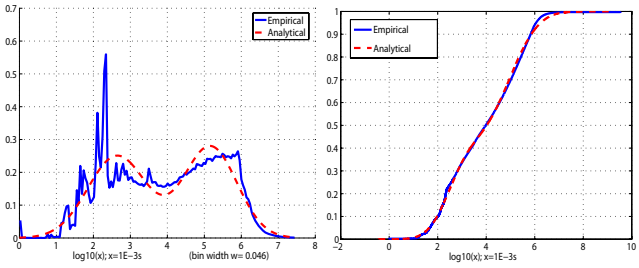

Fig. 8. PDF (left) and CDF (right) of UNINA conversation durations

TABLE IV

STATISTICAL FITTING ON CONVERSATION INTER-ARRIVAL TIMES

Hour	Distribution	λ^2	Scale	Shape
9	Weibull	2.22E-3	100484	0.927
10	Weibull	1.51-3	88769	0.933
11	Weibull	9.53-4	85604	0.942
12	Weibull	1.42-3	82430	0.944

difficult to fit. Also, it spans through a large range of values, from $0ms$ to $8hrs$, so it can be studied more easily after applying a log_{10} transformation. This distribution can be considered at least bi-modal. We obtained the best results by approximating it with a mixture of two Normal distributions, using the EM algorithm. In Fig. 8, by comparing the PDFs, we can observe that the obtained analytical distribution does not fit well all the spikes in the lower part of the distribution. However, looking at the CDFs diagrams we can see that the overall behavior is preserved. We also found this distribution to be heavy-tailed (before moving in the log domain). This result is consistent with the presence of a heavy tail in the IPT distribution.

As regards conversations inter-arrival times, since the rate of new conversations heavily changes during the day (Fig. 2), we split the empirical data into separate segments for each hour of the day. Unlike the conversation duration distribution, such distributions presented rather smooth curves, and in all cases we obtained the best discrepancy results by fitting the empirical data with a Weibull distribution (Fig. 9) with different *shape* and *scale* parameters depending on the corresponding time of the day. To give an example, in Table IV we report fitting results related to the UNINA trace from 9am to 1pm, when there is a constant increase in the conversation rate. We can observe that as this value increases, probably because of a higher number of active users, the *scale* decreases and the *shape* gets closer to 1. A Weibull distribution of *shape* equal to 1 is actually an Exponential distribution. We found, indeed, good λ^2 discrepancy values (of the order of 1E-2 or 1E-3) also fitting the data with the Exponential distribution. This is also visible by plotting the Complementary CDF of the empirical data with the y-axis in logarithmic scale, see Fig. 9, and observing that it approximately follows a straight line, which corresponds to an Exponential distribution. This observation is particularly interesting if associated with the fact that, as said before, we found the conversation duration distribution to be heavy-tailed. Because, if we assume independence between conversation arrivals and durations, we can approximate the entire traffic generation process, at a conversation level, as an

$M/G/\infty$ queue with a heavy-tailed distribution for service times. Such queuing model is known to generate LRD traffic [32], a common property found in network traffic.

*3) **A review of the literature**:* HTTP traffic has been obviously the subject of many research studies. This sub-section reviews the literature on HTTP traffic modeling with the aim to highlight the peculiarities of a packet-level approach. The self-similar nature of HTTP traffic was stressed in [26]. This property was proved to have a significant negative impact on network performance [33]. Self-similarity was explained looking at the distribution of file sizes, transfer times, and inactivity times, which the authors studied by analyzing application-level data, as logs from clients and servers. Using application-level logs as a source to infer traffic characteristics proved to be difficult and poorly scalable. For these reasons, most of the subsequent works have been based on the analysis of packet traces. The authors of [24] expanded the set of variables studied, partially based on heuristics to detect user clicks. Indeed the limit value of $1s$ was chosen to distinguish between *Active* and *Inactive OFF* times, allowing also to ascribe several files to the same Web page. Modeled variables included byte lengths of requests and replies, document size, user think time, etc. The authors of [23] applied a modeling methodology similar to that in [24] but tried to deal with the advent of the new HTTP/1.1 standard. The authors also introduced the concept of user *session*, to be used in the traffic generator they implemented. All of the cited works assumed that the studied HTTP traffic was only related to the Web, but in the last years the number of applications using HTTP rapidly increased [34]. In 2004 the authors of [35] presented a new modeling approach based on TCP connections rather than on Web-page structure, claiming that their models would have been more appropriate for network traffic simulation as opposed to server workload simulation. Also, authors stated that their work was a first result of a general framework to model other TCP applications or mixes of them. They studied parameters as the inter-arrival time between TCP connections and, for each connection, the number of request/response exchanges, the delay between responses and subsequent requests, the number of bytes for each request and each response, etc. A problem with such modeling approach is that it is still dependent on the HTTP implementations currently found on network traffic, being strictly based on the concept of requests and responses alternating inside a TCP connection. For example the pipelining option introduced in HTTP/1.1, which is currently rarely used but could become more frequent in the near future, allows to send requests before the current reply is completely

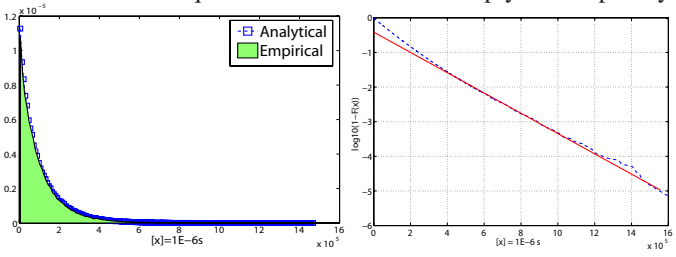

Fig. 9. UNINA conversations inter-arrivals: PDF (left) and Log-CCDF (right)

transmitted. They also added a heuristic, similar to that of [24], to distinguish between requests of files belonging to the same document and requests of different documents. Not only most of the models proposed are based on heuristics, but they are often relatively complex. Therefore, it is quite simple to understand how an HTTP packet-level characterization provides the following benefits when compared with higher level approaches: (I) simple/concise and at the lowest/deepest point of view; (II) network devices (Routers, Switches, Access Points) often operate on a packet-by-packet basis (i.e. buffer management where queues manage packets); (III) network problems (Loss, Delay, Jitter) happen at packet level; (IV) it is independent of future protocol evolution and applicable to different applications/protocols; (V) it is usable in traffic generators/simulators/emulators.

C. SMTP Traffic Characterization

We wanted to look at other type of traffic using the same methodology to show that a different type of traffic exhibits different properties when analyzed at packet level. To this aim we present a short comparison with preliminary results on SMTP. In this case as well, comparing traffic from ARIGE and UNINA networks, we found a spatial and temporal independence. Therefore, as for the SMTP traffic we briefly show preliminary results of the same approach applied to traffic related to port TCP 25, captured at UNINA in the same days as the HTTP traffic trace. The diagrams in Fig. 10 show

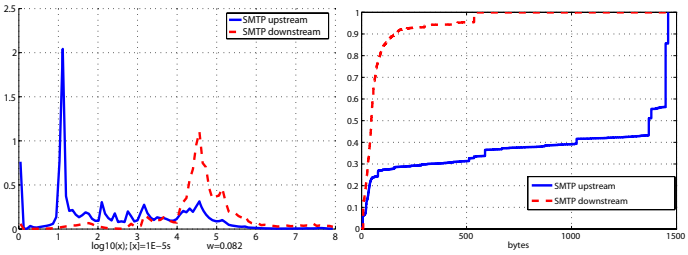

Fig. 10. UNINA port TCP 25: IPT PDFs (left), payload CDFs (right)

indeed a totally different behavior from HTTP traffic, and instead reflect the mechanisms of the SMTP application-level protocol. Servers (mail receivers) tend to send small packets with inter-packet times mostly concentrated around $300ms$. The CDF diagram shows that about 90% of the payloads are smaller than 100 bytes and almost all payloads do not go beyond 300 bytes.

On the contrary, looking at upstream traffic we find that clients (mail senders) behavior is dominated by large back-to-back packets. Also, in the payload-size distribution we found spikes corresponding to negotiated MSS, as explained above. The jump at 1380 bytes is easily recognizable in Fig. 10.

The observed distribution characteristics are strongly related to the application. Indeed in the SMTP protocol the mail receiver answers to commands and data submissions with a rigid syntax in which replies have a numeric code, therefore they are usually very short. Mail senders, after issuing commands, must instead transfer all the mail content, prepended by headers, and often with binary objects attached. On the other side, knowing the application characteristics, it is evident that, unlike for

HTTP, there is not much user interaction involved. However, user influence can be seen at least with the presence of long emails and file attachments. The latter, for example, lead to a higher percentage of maximum-size packets (data transfer) compared to small-size ones (possibly SMTP commands). This aspect can also be observed in conversations data, such as conversation duration and transferred bytes. Once again, this reveals a correlation between these two observation levels.

V. Conclusion and issues for future research

In this paper we proposed an approach to traffic characterization at packet level. We applied the proposed methodology at both HTTP and SMTP traffic. We selected HTTP traffic because it was deeply investigated and a lot of papers about HTTP traffic modeling and characterization are present in literature. Despite this large number of works, few papers are devoted to traffic characterization at packet level. Therefore, the study of HTTP gave us a twofold opportunity. First, we provided a characterization of a well established and largely diffused protocol. Second, thanks to the large number of papers studying HTTP (not at packet level) we compared, contrasted, and understood the results we obtained with a packet-level characterization. Furthermore, in order to show how our approach is simply usable in the case of other protocols and how a packet-level approach can be useful to understand traffic peculiarities we showed some results regarding SMTP traffic. The first results of this work look very encouraging, demonstrating that packet-level characterizations show invariant properties, in terms of time (temporal invariance) and observed link (spatial invariance), when an enough statistically relevant set of samples is chosen. We can summarize what we obtained from the analysis presented in this work in the following points: (I) both conversation-level and packet-level models show invariance in respect to space and time; (II) conversation-level process can be modeled with an $M/G/\infty$ queue; (III) payload size and inter-packet time distributions of upstream traffic are very different from the ones related to downstream; (IV) MSS can influence the payload-size distribution; (V) we found analytical distributions approximating the preliminary empirical ones; (VI) thanks to the process of sanitizing traffic traces we found confirmation that P2P traffic using port TCP 80 to bypass firewalls seems to dramatically change the traffic profile on this port; (VII) finally, in order to assess the validity of our packet-level approach, we showed that preliminary results of the same methodology applied to SMTP traffic look quite different from the models we found for HTTP. Another kind of contribution in the context of this work is the public availability of the analyzed traffic traces and of the software that has been developed [21]. In the future, we look forward to enlarge the number and characteristics of the observed links, to extend the number of studied protocols, and to employ such models in both traffic generators (D-ITG [25]) and simulation environments.

Acknowledgment

This work has been partially supported by the MIUR in the framework of the PRIN 2004 Quasar Project and by the E-Next European project. The authors thank Francesco Palmieri and Maurizio Aiello for providing traffic traces.

References

[1] P. Danzig, S. Jamin, R. Caceres, D. Mitzel and D. Estrin, "An Empirical Workload Model for Driving Wide-Area TCP/IP Network Simulations". *Internetworking: Research and Experience*, Vol. 3, N. 1, pp. 1-26, Mar. 1992.

[2] V. Paxson and S. Floyd, "Wide Area Traffic: The Failure of Poisson Modeling". *IEEE/ACM Trans. Networking*, Vol. 3, N. 3, pp. 226-244, Jun. 1995.

[3] M. S. Borella, "Source Models of Network Game Traffic". *Computer Communications* 23(4), pp. 403-410, Feb. 2000.

[4] T. Lang, G.J. Armitage, P. Branch, H. Choo. "A Synthetic Traffic Model for Half-Life" ATNAC 2003, Melbourne, Dec. 2003

[5] J. van der Merwe, R. Caceres, Y. Chu, C. J. Sreenan, "mmdump: A Tool for Monitoring Internet Multimedia Traffic". *ACM SIGCOMM Computer Communication Review*, Vol. 30, N. 5, pp. 48-59, Oct. 2000

[6] L. Liang, Z. Sun, M. Howarth, "Measurement and Modelling of WWW Traffic in a LAN Environment". *EUROCON 2003*, Vol. 1, pp. 433-437, 22-24 Sep. 2003

[7] S. Floyd, V. Paxson, "Difficulties in simulating the Internet". *IEEE/ACM Trans. Networking*, Vol. 9 , Issue 4, pp. 392-403, Aug. 2001

[8] R. Ohri, E. Chlebus. "Measurement Based E-mail Traffic Characterization". *SPECTS'05*, July 05

[9] S. Luo, G. A. Marin. "Realistic Internet Traffic Simulation through mixture modeling and case study". *2005 Winter Simulation Conference*, pp. 2408-2416, Dec. 05

[10] A. T. Andersen and B. F. Nielsen. "A Markovian Approach for Modeling Packet Traffic with Long-Range Dependence". IEEE Journal On Selected Areas in Communications, vol. 16, no. 5, pp. 719-732, June 1998.

[11] A. Dainotti, G. Iannello, F. Palmieri, A. Pescapé, P. Salvo Rossi, G. Ventre. "An HMM Approach to Internet Traffic Modeling", Submitted to Globecom 2006

[12] http://www.wide.ad.jp/wg/mawi/

[13] A. Dainotti, A. Pescapé, "Plab: a packet capture and analysis software architecture". Dip. di Informatica e Sistemistica, Univ. of Naples Federico II, Italy. Tech. Rep. TR-DIS-122004, Oct 2004 - http://www.grid.unina.it/Traffic/pub/TR-DIS-122004.pdf

[14] S. McCanne, V. Jacobson, "The BSD packet filter: A new architecture for userlevel packet capture". *Winter 1993 USENIX Conference*, pp. 259-269, Jan. 1993

[15] D. Agarwal, J. M. Gonzalez, G. Jin, and B. Tierney "An infrastructure for passive network monitoring of application data streams". *PAM 2003*, Apr. 2003

[16] S. Pederson, M. Johnson, "Estimating Model Discrepancy". *Technometrics*, 32(3), pp. 305-314, Aug. 1990

[17] V. Paxson, "Empirically-Derived Analytical Models of Wide-Area TCP Connections". *IEEE/ACM Trans. Networking*, Vol. 2, N. 4, pp. 316-336, Aug. 1994.

[18] D. W. Scott, "On Optimal and Data-Based Histograms". *Biometrika*, Vol. 66, pp. 605-610, 1979.

[19] J.A. Bilmes, "A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models". Technical Report, ICSI-TR-97-021, University of Berkeley, CA, 1998.

[20] M. Crovella, "Network Traffic Modelling". *ACM SIGCOMM 2004* tutorial, Aug 04

[21] http://www.grid.unina.it/Traffic/

[22] R. Caceres, P. Danzig, S. Jamin, D. Mitzel, "Characteristics of Wide-Area TCP/IP Conversations". *ACM SIGCOMM Computer Communication Review*, Vol. 21 Issue 4, pp. 101-112, Sep. 1991.

[23] H. Abrahamsson, B. Ahlgren, "Using Empirical Distributions to Characterize Web Client Traffic and to Generate Synthetic Traffic". *GLOBECOM '00*, Vol. 1, pp. 428-433, Nov. 2000.

[24] B. A. Mah, "An Empirical Model of HTTP Network Traffic". *INFOCOM '97*, Vol. 2, pp. 592-600, Apr. 1997

[25] http://www.grid.unina.it/software/ITG

[26] M. Crovella, A. Bestavros, "Self-Similarity in World-Wide Web: Evidence and Possible Causes". *IEEE/ACM Trans. Networking*, Vol. 5, N. 6, pp. 835-846, Dec. 1997.

[27] F. D. Smith, F. H. Campos, K. Jeffay, and D. Ott., "What TCP/IP Procotol Headers Can Tell Us About the Web". *ACM SIGMETRICS 2001*, pp. 245-256, 2001.

[28] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, C. Diot, 'Packet-Level Traffic Measurements from the Sprint IP Backbone". *IEEE Network*, Vol. 17, Issue 6, pp. 6-16, Dec. 2003

[29] T. Karagiannis, A.Broido, N.Brownlee, kc claffy, and M.Faloutsos. "Is P2P dying or just hiding?". *GLOBECOM '04*, Vol. 3, pp. 1532-1538, Dec. 2004

[30] Cisco Systems "Blocking Peer-to-Peer File Sharing Programs with the PIX Firewall". http://www.cisco.com/en/US/products/hw/vpndevc/ps2030/products_tech_note09186a00801e419a.shtml

[31] T. Karagiannis, A.Broido, M. Faloutsos, and kc claffy. "Transport layer identification of P2P traffic". *4th ACM SIGCOMM IMC*, pp. 121-134, 2004

[32] Parulekar, M.; Makowski, A.M., "M/G/∞ input processes: a versatile class of models for network traffic". *INFOCOM '97*, Vol.2, pp. 419-426, Apr. 1997

[33] A. Erramilli, O. Narayan, W. Willinger, "Experimental queueing analysis with long-range dependent packet traffic". *IEEE/ACM Trans. Networking*, Vol. 4, N. 2, pp. 209-223, Apr. 1996.

[34] T. Karagiannis, K. Papagiannaki and M. Faloutsos, "BLINC: Multilevel Traffic Classification in the Dark". *ACM SIGCOMM 2005*, pp. 229-240, Aug. 2005 .

[35] J. Cao, W.S. Cleveland, Y. Gao, K. Jeffay, F.D. Smith, M.C. Weigle, "Stochastic Models for Generating Synthetic HTTP Source Traffic". *INFOCOM 2004*, Vol. 3, pp. 1546-1557, Mar. 2004