Identification of Peer-to-Peer VoIP Sessions Using Entropy and Codec Properties

João V. Gomes, Pedro R. M. Inácio, Manuela Pereira, Mário M. Freire, and Paulo P. Monteiro

Abstract—Voice over Internet Protocol (VoIP) applications based on peer-to-peer (P2P) communications have been experiencing considerable growth in terms of number of users. To overcome filtering policies or protect the privacy of their users, most of these applications implement mechanisms such as protocol obfuscation or payload encryption that avoid the inspection of their traffic, making it difficult to identify its nature. The incapacity to determine the application that is responsible for a certain flow raises challenges for the effective management of the network. In this article, a new method for the identification of VoIP sessions is presented. The proposed mechanism classifies the flows, in real-time, based on the speech codec used in the session. In order to make the classification lightweight, the behavioral signatures for each analyzed codec were created using only the lengths of the packets. Unlike most previous approaches, the classifier does not use the lengths of the packets individually. Instead, it explores their level of heterogeneity in real-time, using entropy to emphasize such feature. The results of the performance evaluation show that the proposed method is able to identify VoIP sessions accurately and simultaneously recognize the used speech codec.

Index Terms—Data communications, distributed applications, network communications, network management, network monitoring, packet-switching networks.

1 I

THE popularity of Voice over Internet Protocol (VoIP) applications relying on the peer-to-peer (P2P) paradigm has been growing in the last few years. The simplicity of these solutions, as well as their economic benefits over the traditional telephony, make them an increasingly common choice for long distance calls and voice conferences. Furthermore, the possibility to integrate them in mobile devices, like smartphones and tablets, make them more flexible and easy to use. When implemented over P2P systems, VoIP applications benefit from the scalable and reliable properties of the distributed nature of the P2P model, which puts the intelligence at the network edges.

Over the years, many of these applications have started to adopt measures to disguise their traffic and avoid the inspection of their contents. Protocol obfuscation, payload encryption, and the use of random port numbers are now common features in the majority of the popular VoIP software clients. *Skype* is the most demonstrative example of this trend: it is based on a closed code and proprietary P2P protocol, its communications are encrypted, and it has a large number of users. Nevertheless, there are also other VoIP solutions based on P2P communications that use different protocols. The Session Initiation Protocol (SIP), used by several VoIP applications, or an extension of the Extensible Messaging and Presence Protocol (XMPP), used by *Google Talk*, are good examples of such VoIP systems.

In most of these applications, the implementation of techniques to avoid the inspection of traffic has primarily the intention of protecting the privacy of the data of the VoIP sessions. However, it also makes it more difficult to correctly and effectively manage computer networks. Understanding what kind of data is being transmitted in each flow is of critical importance to organize the network and its traffic, distribute the available bandwidth fairly, or guarantee the Quality of Service (QoS) needed by distinct classes of traffic [1], [2], [3]. Besides the impact that VoIP applications may have in the network performance, they also raise a few security concerns. Several authors [4], [5], [6] and security institutes or companies [7], [8], [9] have exposed the potential vulnerabilities associated with VoIP systems and suggested a few guidelines to avoid security flaws.

For these reasons, traffic classification based on the application protocol has been a very active research field. The identification of VoIP, especially *Skype* related traffic, has attracted the attention of many researchers who have addressed this topic in several articles [1], [2], [3], [10], [11]. In the majority of the cases, whether the classification is made by resorting to payload inspection, flow-level heuristics, statistical analysis, or machine learning algorithms, the goal is to identify the whole data generated by the VoIP application. These flows are generated by a signaling protocol that initiates, controls, and terminates the session and by a transport protocol responsible for delivering the data from one peer to the other. The signaling data, as well as the flows used for authentication and other operations, have little impact

- *J. Gomes, P. Inácio, M. Pereira, and M. Freire are with Instituto de Telecomunicações, Department of Computer Science, University of Beira Interior, Portugal.*
  E-mail: jgomes@penhas.di.ubi.pt, {inacio, mpereira, mario}@di.ubi.pt
- *P. Monteiro is with Nokia Siemens Networks Portugal, S. A., with University of Aveiro, and with Instituto de Telecomunicações.*
  E-mail: paulo.1.monteiro@nsn.com

Manuscript received 8 May 2011.

on the network performance when compared with the data from the VoIP session. Hence, a distinct approach is followed in this work. Instead of aiming for the identification of the whole traffic from a certain VoIP application, the intention of this work is to identify the traffic from the actual VoIP session. The data transported within each packet of the session flow depends more on the speech codec used to codify the voice than on the signaling protocol or client application. In fact, the data from VoIP sessions, made using distinct applications and even distinct signaling protocols, has similar characteristics when the same codec is used. From the traffic management perspective, it may be more useful to identify the VoIP traffic with similar characteristics regardless of the application or protocol that was used than to base the classification on the specific application that has generated it, which may include flows with different properties and purposes. Moreover, the QoS requirements are different for each speech codec [12]. Hence, instead of prioritizing any VoIP traffic, it may be interesting in some network scenarios to prioritize the VoIP sessions where specific speech codecs are used or even to allow only sessions whose speech codecs require less available resources.

This article presents a VoIP classifier that is suitable for real-time analysis and does not rely on the payload data, being therefore applicable for encrypted traffic. Unlike most previous works, the goal of the classifier described herein is to identify the traffic flows that are related with a VoIP session in which a specific codec was used. Moreover, it is our intention to minimize the number of packet-level or flow-level characteristics required to identify VoIP sessions, so as to make the whole classification process lightweight. The length of the packets was the only traffic feature used in the identification of VoIP flows. Instead of looking at the lengths individually or calculating their mean, we focused on the relation between the different lengths and explored their level of heterogeneity using entropy. The entropy is invariant to the particular values and can be updated with a point-by-point algorithm, which favors the robustness of the classifier without jeopardizing performance. The characteristics of the packets from VoIP sessions using different codecs were carefully analyzed. Several distinct applications and speech codecs were considered in the study. Based on this analysis, a set of behavioral signatures for each codec is proposed. Each of them is formed by an interval for the entropy and another one for the lengths of the packets. Additionally, a sliding window with a constant size of N packets was implemented to assess the heterogeneity in real-time and to avoid losing the sensitivity to the local changes in the values. To the best of our knowledge, the level of heterogeneity was used for the purpose of traffic classification only in our previous works [13], [14], with the exception of a recent study that has followed a similar approach [15] only for offline analysis and for complete flows.

The performance of the classification mechanism was evaluated using datasets containing traffic from VoIP sessions as well as from multiple P2P and non-P2P applications or services. The results show that the method identified the flows from VoIP sessions with very good accuracy and it was also able to recognize the speech codec with a good sensitivity rate. Moreover, the analysis of the computational resource consumption showed that it grows linearly with the size of the input data.

The remainder of the paper is structured as follows. Section 2 describes the previously published related work. The analysis of speech codecs considered in the scope of this work is included in section 3. Section 4 presents the classifier and the evaluations of its performance is discussed in section 5. The last section summarizes the most important conclusions.

2 R W

The classification of traffic from VoIP applications has already been studied by several authors. A few studies relied on the data carried within the payload to create signatures to identify *Skype* packets [16]. In some cases, the inspection of bytes in the payload is combined with statistical data, behavioral patterns, or heuristics [11], [17], [18], [19], [20]. A different approach followed by a few authors is based on the fact that the payload data from packets generated by applications that encrypt the traffic is more random. Bonfiglio et al. [10] explored the randomness of the payload data by using the *Chi Square* test and applied the method to *Skype* traffic. Additionally, they proposed a statistical classifier based on inter-arrival times and packets lengths. In [21], [22], the authors resorted to entropy to analyze the randomness of the bytes within the packet payloads in encrypted traffic.

Methods based on heuristics are proposed in some articles [1], [23], [24], [25], as well as statistical methods that analyze flow or packet-level features to identify VoIP traffic [2], [26]. The use of machine learning algorithms has also been applied to the traffic classification, and specifically, to the VoIP traffic identification. Jun et al. [27] proposed a method to identify *Skype* traffic based on the *Random Forest* classifier, while Branch et al. [28] relied on the *C4.5* decision tree algorithm. In [29], symbiotic bid-based genetic programming was used to identify *Skype* encrypted traffic and the performance was compared with *C4.5* and *AdaBoost* algorithms. Wu et al. [30] explored characteristics of the human behavior, as the speech period, and used a Naïve Bayes classifier to identify VoIP traffic. Zhang et al. [31] proposed a method based on Support Vector Machines (SVMs), that uses a set of traffic features to identify *Skype* communications.

The approach followed in this article resorts to the characteristics of the lengths of the packets. Several previous works have already used the lengths of the packets as one of the features employed in the traffic classification. Nonetheless, they analyzed them mostly through statistics that use the actual value of the packet lengths, such as the mean [20] or the standard deviation [21], or through the use of intervals [32] or probabilistic models [10]. On the contrary, instead of focusing on the lengths of the packets *per se*, the method described herein explores the relation between the different lengths by analyzing how heterogeneous these values are. Many of the previous works that explore flow level properties, through statistical measures [25] or machine learning algorithms [31], separate the traffic into flows offline and then apply the classification approach, making these methods difficult to adapt (or even unsuitable) for the real-time analysis of the traffic. The approach followed herein implements a sliding window with size of N packets that produces information about the traffic characteristics in every iteration, during all the duration of the flow since its beginning.

Furthermore, besides identifying the VoIP related data, the proposed classifier also tries to give a strong prediction on the speech codec used in a VoIP session, instead of identifying the VoIP application, which is the goal of most studies. In fact, although packet or flow properties, like the length of the packet or the inter-arrival time, differ when distinct codecs are used, most studies seem to use them without considering the speech codec. Besides that, some of the properties identified as being specific for a certain VoIP application may also apply to other applications that use similar codecs. Nonetheless, a few authors have considered the influence of distinct codecs when proposing a classification method. Branch et al. [28] analyzed the traffic from the Sinusoidal Voice Over Packet Coder (SVOPC) codec, while Molnár and Perényi [25] focused on the Internet Speech Audio Codec (iSAC). Chen et al. [26] considered iSAC and the Internet Low Bit Rate Codec (iLBC) and Yildirim et al. [32] analyzed three Constant Bit Rate (CBR) codecs, *G.711*, *G.723*, and *G.729*. Xu et al. [33] proposed a traffic classification method based on a finite state machine and applied it to identification of *Skype* traffic generated using SVOPC, Adaptive Multi-Rate Wideband (AMR-WB), *G.729*, and Pulse-Code Modulation (PCM). In the statistical analysis of *Skype* VoIP flows described in [34], the authors considered iSAC. A more comprehensive set of codecs, which includes iSAC, iLBC, *G.729*, Internet Pulse Code Modulation wideband (iPCMwb), Enhanced *G.711* (EG711) A/U, PCM A/U, and SVOPC, was analyzed by Bonfiglio et al. in a study of *Skype* traffic [3] and used in the classifier described in [10]. Nevertheless, none of these works presented a method to identify the codec used in a VoIP session, nor proposed signatures for each codec. Moreover, the analyzed codecs are mostly codecs used by older versions of the *Skype* software.

We proposed the analysis of the level of heterogeneity of the lengths of the packets from P2P applications and its quantification through entropy for the first time on a previous article [13]. The work described herein elaborates on that method, and evolves to the identification of VoIP traffic from different speech codecs. The most comparable work was published recently by Li et al. [15] who used a similar approach, in conjunction with an analysis of the inter-arrival times, to identify CBR and Variable Bit Rate (VBR) codecs. Their method is based on the idea that CBR codecs produce packets with constant lengths and VBR codecs produces packets with different lengths. They did not analyze the behavior of different codecs, nor try to identify the specific codec used in a session. Moreover, even the different VBR codecs may produce packets whose lengths can be more or less heterogeneous depending on the specific codec. The algorithm proposed by Li et al. is also based on the offline analysis of the traffic. The heterogeneity of the traffic is analyzed for complete flows. Besides preventing the method from being applied to real-time monitoring, their approach also raises a few problems. If the characteristics of the traffic change or occasional occurrences of different lengths appear in the middle of the flow, the results of the analysis for the whole flow may be compromised. The work from Li et al. appears to be based on [35]. Liu et al. [36] explored the ratio between small packets and large packets and used that value, together with a few heuristics, to identify P2P traffic offline. Wright et al. [37] used the packet lengths for a different purpose. Instead of identifying the application or codec that generated the data, they analyzed the lengths of the packets generated by VBR codecs to try to recognize spoken phrases in encrypted VoIP sessions.

TABLE 1
Applications and codecs considered in the study.

Application	Codecs
Blink	PCM A/U, *G.722*, iLBC, GSM, *Speex*
Ekiga	PCM A/U, *G.722*, iLBC, GSM, *Speex*
Linphone	PCM A/U, GSM, *Speex*
QuteCom	PCM A/U, *G.722*, GSM, *Speex*
SIP Communicator	PCM A/U, *G.722*, GSM, *Speex*
Skype	iPCMwb, iSAC, EG711 A/U, PCM A/U, iLBC, *G.729*, AMR-WB, SVOPC, NWC, *SILK*
X-Lite	PCM A/U, iLBC, GSM, *Speex*

3 A S C

The proposed method is based on the properties of the lengths of the packets for different codecs, regardless of the VoIP application. To understand and study the behavior of the traffic from each codec, it was necessary to collect traffic from VoIP sessions using different speech codecs. A set of applications was used to perform the calls so as to consider any possible influence of the application in the characteristics of the traffic. With the exception of *Skype*, the used applications resort to SIP for signaling. Table 1 presents a summary of the applications and codecs considered in this article. The details about the speech codecs considered in this article are described in appendix A of the supplemental material.

To allow the capturing of experimental data from specific codecs, we included only VoIP applications that offer the possibility of choosing the codec in a preferences menu. Moreover, since it was our intention to use *Microsoft Windows* and *Linux* platforms in the experiences, only applications that have versions for both operating systems were selected. The datasets used in the traffic analysis, which are not the same used in the performance evaluation, are described in subsection C.1 of the supplemental material.

3.1 Properties of the Codecs

The goal of this work is to identify the traffic from VoIP sessions and, as such, it is reasonable to focus the observation on the packets of each flow separately. The concept of flow used herein coincides with the Transmission Control Protocol (TCP) notion of connection. In the case of User Datagram Protocol (UDP) traffic, a flow includes all the packets traveling between two *(host, port)* pairs, in both directions, with inter-arrival times inferior to 64 seconds, as suggested in [38].

Nonetheless, since the properties of the traffic in each VoIP session are similar in both directions, the classifier described in section 4 analyzes each unidirectional flow separately so as to distinguish the cases where the traffic properties may match VoIP traffic characteristics in only one direction. Furthermore, in some situations, *Skype* uses hosts, called relay nodes, that act as middle nodes mainly to overcome connection problems from users that are behind Network Address Translation (NAT) systems. We observed that, in some of these cases, it is possible to have a host receiving the incoming VoIP data from a relay node, and sending the outgoing data to a different node. The only common properties in this situation are the Internet Protocol (IP) address of the monitored host and the port used for the *Skype* session. Hence, in order to identify these VoIP connections, besides the flow perspective, the traffic was also analyzed from the point of view of the *(host, port)* pair. This approach enables an observation level that includes all the traffic sent and received by the application process responsible for the VoIP session, even if relay nodes are used. Likewise, the analysis examples presented in this subsection concern all the traffic generated by a VoIP session, whether relay nodes are used or not.

In order to identify properties of the packet lengths from each codec, we analyzed the traffic of the VoIP sessions included in the data described in subsection C.1 of the supplemental material. We observed that distinct codecs produce packets whose lengths present different levels of heterogeneity, which we measured by resorting to the entropy. As explained before, the experimental data contained sessions generated with different VoIP applications, transport protocols, and operating systems. Nevertheless, the obtained results were similar for each codec, regardless of those factors. Hence, the entropy values calculated for the different codecs were used to identify patterns. In appendix B of the supplemental material, we described, with detail, the process used to assess the entropy and the implementation of the entropy computation in real-time by resorting to sliding windows. The value of the entropy depends also on the considered window size, increasing moderately when the size of the window increases. Nevertheless, the most noticeable consequence of increasing the size the window is the stabilization of the entropy values throughout the iterations of the window, which creates a smoothing effect (resulting from the Law of Large Numbers). The analysis of the packet lengths included in this article considers only the length of the data carried within the transport payload and excludes the small packets whose payload has length less or equal to 5 bytes. More details regarding the effect of the sliding window size and about the packet lengths are provided, respectively, in subsections C.2 and C.3 of the supplemental material.

The following subsections describe the properties identified for CBR and VBR codecs. AMR-WB is a multi-rate codec, formed by nine source codecs with distinct constant bit rates. The bit rate it uses may change every 20 milliseconds. Therefore, in spite of being a CBR codec, AMR-WB will be analyzed along with the VBR codecs. *Speex* supports CBR and VBR and, thus, examples of VoIP sessions using both modes will be analyzed with the remaining CBR and VBR codecs. The presented examples refer to analyses of the lengths of the transport-level payload, filtering out the packets whose payload is less or equal to 5 bytes and using sliding windows with size of 500 packets.

3.2 Constant Bit Rate Codecs

The traffic from VoIP sessions that use CBR codecs is constituted mostly by packets with the same length and, as such, the entropy level is extremely low. For the applications that use SIP, the entropy is almost always equal to 0 as the payloads of most packets have the same length. Although the traffic is also very homogeneous when using *Skype*, there are a few occurrences with different lengths. Because *Skype* uses its own proprietary protocol, it is difficult to understand why this happens.

In Fig. 1, one can observe a comparison between the first three minutes of VoIP sessions that used PCMA, PCMU, and iLBC, made using *Skype* and SIP clients. Although in both sessions the payloads have the same length, in the case of *Skype* there are also a few packets whose payloads have a different length. This behavior was observed in all of the analyzed VoIP sessions in which CBR codecs were used.

Due to the limitations of space in the main article and to the large number of charts that would be needed to represent every session example, we included Table 1 in the supplemental material to give a general view of the values obtained for the datasets described in subsection C.1 of the supplemental material. The table contains the mean of the entropy for all the VoIP sessions that used each CBR codec, as well as the most frequent lengths of the transport-level payload that were

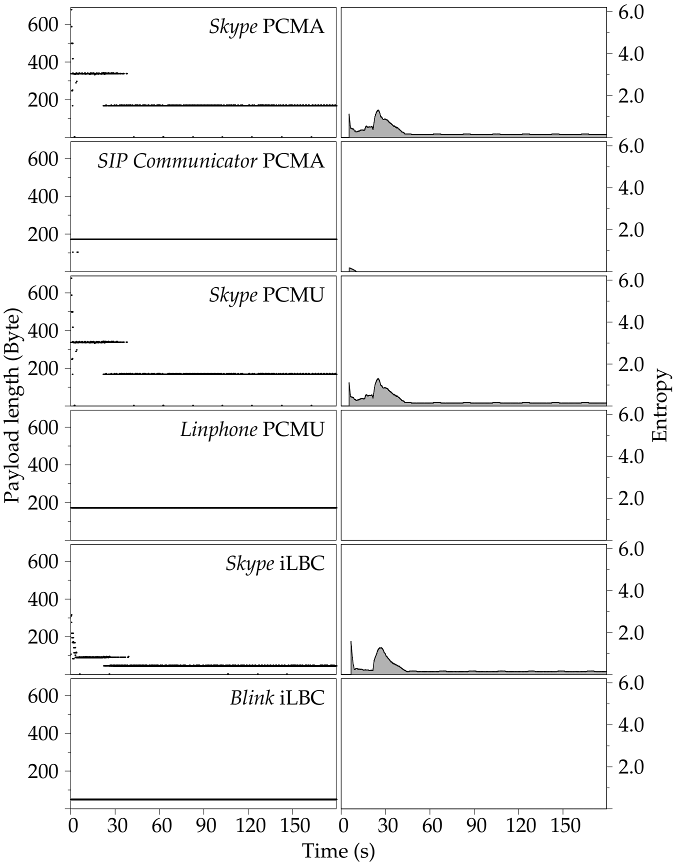


Fig. 1. Comparison of the lengths of the payloads and of the entropy between VoIP sessions using *Skype* and SIP applications with CBR codecs.

observed. For each VoIP session, the mean of the entropy in all the iterations of the window was calculated, which results in one entropy value for session. The mean of the values obtained for all the sessions in which the same codec was used was computed and included in the table.

Entropy was analyzed separately for the incoming and the outgoing data. VoIP flows usually have similar properties in both directions, which is also an important feature to distinguish VoIP flows from traffic of other applications. Table 1 of the supplemental material shows this similarity between the traffic in both directions.

The *G.722* codec uses the baseline of PCM and, therefore, the behavior of the traffic from both codecs is similar. During the traffic analysis, we observed that the packet payloads from sessions where *G.722* and PCM were used have similar lengths. Although it was not possible to find any details regarding NWC (because *Skype* does not provide that information), the packets from VoIP sessions based on NWC also have lengths similar to the ones based on PCM. Hence, NWC seems to also be using the PCM baseline.

3.3 Variable Bit Rate Codecs

Unlike the CBR codecs, the traffic from each VoIP session in which a VBR codec is used is formed by packets whose payloads have very heterogeneous lengths. Fig. 2 depicts the lengths of the transport-level payloads and the corresponding entropy of the first three minutes of several VoIP sessions, each of them using a different VBR codec. In all the cases, the variety of distinct lengths form a strip of values. The different levels of heterogeneity of the payload lengths are put in evidence by the distinct levels of entropy depicted in the charts. In a few cases, however, different codecs presented a similar level of entropy, e.g., the Global IP Solutions (GIPS) VBR codecs (EG711 A/U, iSAC, and iPCMwb). Nevertheless, in those cases, the payload lengths are included in different ranges of lengths.

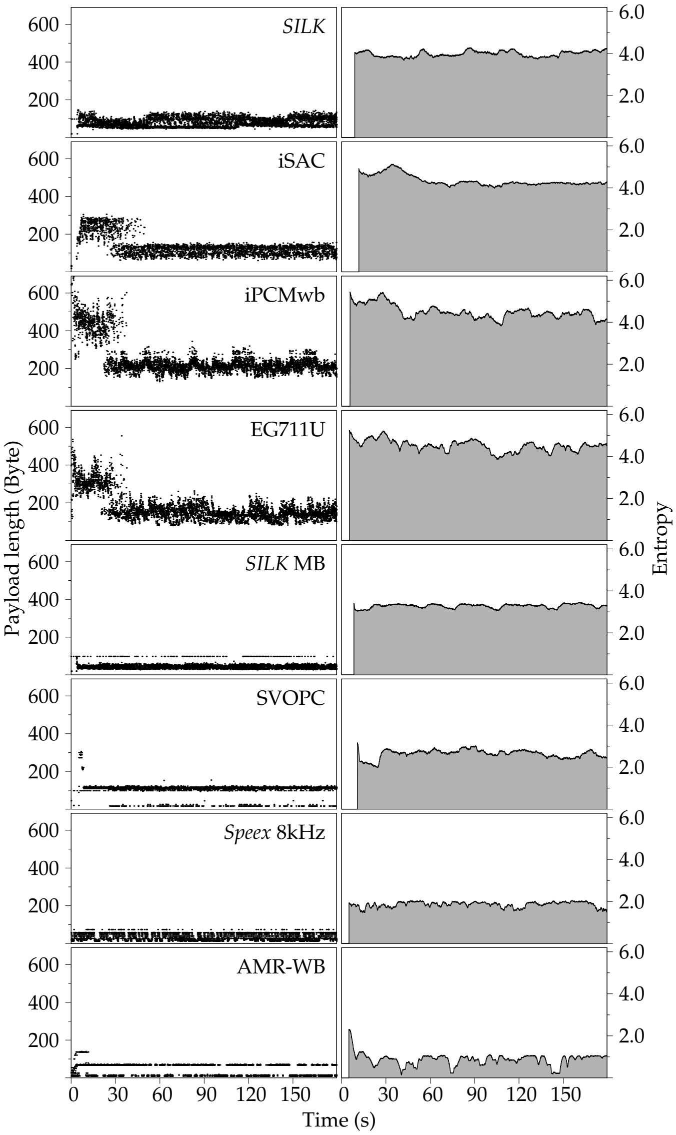


Fig. 2. Representation of the lengths of the payloads and of the entropy of the first three minutes of VoIP sessions using different VBR codecs.

There also other details that are visible in the charts. In the beginning of the VoIP sessions in which the GIPS VBR codec were used, the lengths of the payloads appear on a strip of higher values. Before the 30 seconds mark, they stabilize on lower values. This behavior was also observed in other VoIP sessions in which *Skype* was used, even with CBR codecs as shown in Fig. 1. In the case of *Speex*, the lengths vary within a small range of values, resulting in a lower level of entropy. This is even more evident in the case of AMR-WB, in which the entropy is even lower and less stable.

A summary of the results obtained for all the VoIP sessions from the datasets described in subsection C.1 of the supplemental material in which VBR codecs were used is presented in Table 2 of the supplemental material. The values were obtained in the same way as it was done for the CBR codecs. In the case of the VBR codecs, the table includes ranges of frequent length instead of individual values as the lengths are heterogeneous. The GIPS VBR codecs generate lengths with higher entropy. *SILK* and *SILK* wideband (WB) seem to have similar properties, as well as *SILK* mediumband (MB) and *SILK* narrowband (NB), and *Speex* 32 kHz and 16 kHz. AMR-WB presents a very low entropy when compared with the VBR codecs, which was expectable since it is not a truly VBR codec.

4 T V IP C

The classifier proposed herein is based on the properties described in the previous section. The following subsections provide a list of the proposed signatures and describe the classification mechanism and its operation.

4.1 Behavioral Signatures for the Codecs

A set of behavioral signatures was defined to model the properties described in section 3.1, which result from the observation of the datasets described in subsection C.1 of the supplemental material. The signatures are formed by the codec description, an interval in which the entropy should be contained, an interval in which the payload length should be contained, and a minimum number of occurrences matching these conditions so that a tuple can be classified as a VoIP session. In addition to preventing occasional matches from resulting in immediate classifications, the minimum number of matches also reduces the possibility of having dual classifications due to the overlap of parts of the intervals in distinct signatures.

Table 2 lists the signatures proposed in this work and used by the classifier to identify VoIP sessions. The values defined for the intervals and for the minimum matches were optimized for sliding windows with size of 500 packets and result from the analysis of the experimental datasets and from testing the classifier with those traffic samples. The number of minimum matches is different for each codec as it depends on the existence of other codecs with similar properties or with overlapping intervals in the signatures. Similarly to the other signature components, the number of minimum matches is defined based on the study of traffic samples generated using each of the codecs. Three different levels of signatures were defined. Most of them are signatures created to identify specific speech codecs. Nevertheless, signatures to simply identify VoIP sessions based on CBR, VBR codecs, and VBR codecs with low variation, or other groups of codecs, were also created. Separating the classification into a smaller number of categories improves its accuracy and makes the process faster.

TABLE 2
List of the behavioral signatures, formed by intervals of packet lengths and of entropy, and by a minimum number of matches, for sliding windows with size of 500 packets, used to identify the VoIP sessions.

Signature Description	Interval of Lengths (Byte)		Interval of Entropy		Minimum Matches
		Bit rate level			
CBR	15	400	0.00	1.00	400
VBR (low variation)	10	400	1.25	3.25	450
VBR	15	800	2.80	6.00	450
		Group level			
GIPS VBR	75	700	3.50	5.50	400
PCM based	160	190	0.00	1.00	400
Skype CBR	25	190	0.10	1.00	400
Skype proprietary VBR	20	120	1.50	4.50	400
		Codec level			
G.729	25	30	0.10	0.95	400
GSM	44	45	0.00	0.10	450
iLBC	49	51	0.00	0.10	450
iLBC	87	90	0.00	0.10	450
iLBC *Skype*	46	50	0.05	1.00	400
iLBC *Skype*	86	90	0.05	1.00	400
PCM	165	185	0.00	0.10	450
PCMA *Skype*	160	171	0.10	1.00	450
PCMU *Skype*	170	185	0.10	1.00	450
G.722	171	175	0.00	0.10	450
NWC	160	171	0.10	1.00	450
Speex 32 kHz	85	87	0.00	0.10	450
Speex 32 kHz	45	50	0.00	0.10	450
Speex 16 kHz	80	85	0.00	0.10	450
Speex 16 kHz	40	45	0.00	0.10	450
Speex 8 kHz	50	52	0.00	0.10	450
Speex 8 kHz	30	35	0.00	0.10	450
AMR-WB	45	80	0.15	1.75	250
EG711	200	550	3.00	5.50	400
EG711	75	250	3.50	5.50	400
iPCMwb	250	700	3.00	5.50	400
iPCMwb	150	300	3.50	5.50	400
iSAC	100	300	3.00	5.50	400
iSAC	60	200	3.50	5.50	400
SILK	40	120	2.75	4.50	250
SILK MB/NB	20	60	2.00	3.50	400
SVOPC	80	120	1.50	3.00	400
Speex	20	100	2.00	2.50	400
Speex	20	100	1.50	2.00	400

4.2 Architecture of the Classifier

The implementation of the classifier includes two alternative levels of observation, as explained in section 3.1: per flow or per *(host, port)*. In order to individually identify each flow or *(host, port)* pair, the classifier uses an identification tuple. When the flow perspective is used, the tuple is formed by the source IP address and port number, by the destination IP address and port number, and by the transport protocol (UDP or TCP). For the *(host, port)* pair perspective, the tuple is formed by the host IP address and the port number. In this section, we will use the term *tuple* to designate a generic flow or *(host, port)* pair, depending on which perspective is used.

The network point where the classifier should be deployed depends on factors like the classification purposes, the amount of data that the complete system is able to process, or the typical traffic classes in the network. In most cases, it would be advantageous to place the classifier in the distribution layer of the network so that the results of the classification could be used in the traffic routing and segregation.

During the analysis described in section 3, we observed that the heterogeneity of the packet lengths is similar in both directions for VoIP. On the other hand, for other types of application that may use speech codecs, like audio streaming, the packet lengths may present the heterogeneity associated with a speech codec only in one direction, while the traffic in the opposite direction is mainly formed by acknowledgement messages. Hence, to avoid these cases, the classifier separately analyzes the VoIP session traffic in each direction and only if the packet lengths in both directions have similar properties, the session is classified.

Furthermore, as described in section 3.1, the traffic analysis showed more than one frequent length for some codecs, which results in more than one signature for the same codec in Table 2 (e.g., iLBC codec). We observed that the traffic from some VoIP sessions that use one of those codecs has distinct packet lengths in both directions. For example, in some sessions using the iLBC codec, the packets in one direction had 46 bytes, while in the opposite direction, the packets had 86 bytes. Hence, one of the two iLBC signatures included in Table 2 matches one direction of the VoIP session traffic, while the other signature matches the traffic in the opposite direction. Therefore, when separately analyzing the traffic in each direction, the classifier tries to classify the traffic in both directions with signatures for the same codec, even if the signatures are distinct signatures for the same codec.

The proposed classifier is formed by three modules: one responsible for processing the packets, other for calculating the entropy level, and a third one for identifying the VoIP data. The modular operation of the classifier, illustrated by Fig. 3, is described in appendix D of the supplemental material. In the following subsection, we explain the process of identification of a VoIP session. which is made in the *classification decision* module.

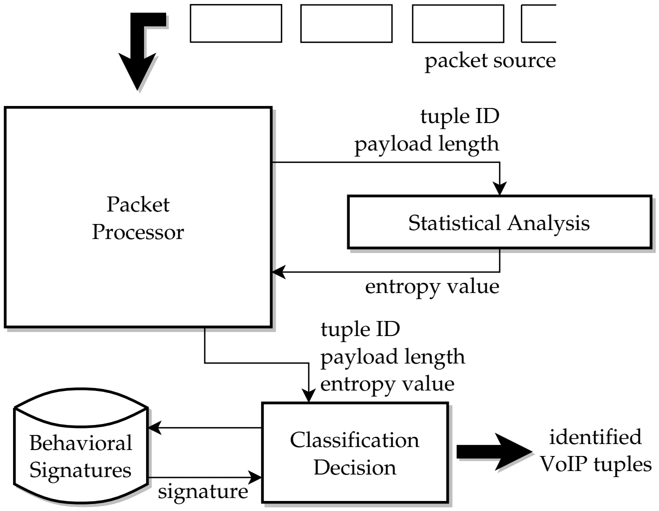


Fig. 3. Architecture of the proposed classifier formed by three modules.

4.3 The *Classification Decision* Module

The *classification decision* module receives, from the *packet processor*, the payload length and the entropy value along with the identification of the corresponding tuple and produces a classification result. The classification process is formed by two main procedures, the signature matching and the classification based on the matched signatures, and is repeated for every processed packet.

During the signature matching process, the module tries to match all the behavioral signatures in the repository. As explained in section 4.1, each signature *S*, associated with a codec *Cod*, is formed by two intervals *E* and *L* to which the entropy and the packet length should belong, respectively, and by the required minimum number of matches *minM* in the latest *W* (size of the sliding window) packets so that the tuple can be classified as traffic generated by *Cod*. For each pair formed by *S* and the analyzed tuple *T*, there is an individual counter *C* of the number of matches in the last *W*, whose value is always between 0 and *W*. The classifier tries to match each signature *S* in the repository, as depicted in Fig. 6 of the supplemental material. Depending on the result of the signature matching, the *C* counter associated with *S* and *T* is decremented or incremented, unless it is already 0 or *W*, respectively.

After all the signatures are checked, each counter *C* associated with each signature *S* contains the number of matches of *S* for tuple *T*, allowing the classifier to make a decision using the method represented in Fig. 7 of the supplemental material. The classifier goes through the signatures repository and checks if, for each codec *Cod*, there is a signature *S1* with required minimum number of matches *minM1* so that the corresponding counter *C1* is greater than *minM1*. If it does, the classifier has to check if the signature for the same codec also matches the traffic in the opposite direction, identified by the inverse tuple *invT*. Since there are different signatures for the same codec, as explained in the beginning of section 4.2, the classifier has to check if there is a signature *S2* (which can be the same as *S1*) for the same codec *Cod* with a minimum number of matches *minM2* so that the corresponding counter *C2* for *invT* is greater than *minM2*. In case both conditions are true, *T* and *invT* are classified as traffic from codec *Cod*, if they have not been before. Otherwise, *T* and *invT* are *unclassified* as traffic from *Cod* if they have been classified before, meaning that they do not present characteristics of *Cod* anymore, because the VoIP session may have finished. Although it was not common during the traffic analysis and the classifier evaluation, if two signatures match the flow, the first to reach the required number of matches determines the classification until the corresponding counter drops below the minimum number of matches. If they reach the minimum required matches in the same iteration of the sliding window, their order in the list of signatures will determine the classification, with the preference decreasing from the first signature to the last one.

5 P E

The evaluation of the classifier was made by resorting to offline data so that the procedure could be repeated and compared against other classifiers. The details about the datasets and the testbed are described in subsection E.1 of the supplemental material. The used metrics are *sensitivity* and *specificity*, which are explained in subsection E.2 of the supplemental material.

The accuracy of the classifier was evaluated separately for the behavioral signatures of the bit rate, group, and codec levels, and the results are listed in Table 3. The proposed method continually analyzes every packet since the beginning of the flow and it makes a classification as soon as the properties of the packet lengths are matched by one of the signatures. Nevertheless, especially in the case of *Skype* traffic, those properties are sometimes distinct at the beginning of the connection. Hence, although the traffic is initially matched by a signature, returning a first classification, seconds later the properties of the packet lengths are more stable and slightly different and are thus matched by a different signature. Since the classifier continues to analyze every packet, it modifies the classification when the traffic is matched by a different signature, resulting in a second classification. This usually happens for similar signatures, such as the ones for *SILK* and *SILK* MB/NB or VBR and VBR (low variation), and it is observable, e.g., in dataset 2. For this reason, the evaluation of the sensitivity for the second classification was also included in Table 3. Nonetheless, if for a certain VoIP flow, the classifier changes the classification several times during the life time of the flow without being able to maintain a stable classification, we considered it as a false negative case, even if one of the classifications was correct. The average time since the beginning of the flow until the classifier reaches the first classification, and in some cases the second, is presented in Table 6 of the supplemental material. This includes the time that the classifier takes to fill the sliding window and the time it takes to reach the minimum number of matches.

Generally, the sensitivity decreases from the bit rate level to the codec level as the signatures are less broad on the latter. For the same reasons, the specificity decreases in the opposite direction. Nevertheless, there are a few exceptions. In the case of the bit rate level signatures, the traffic from low variation VBR codecs was, for a few sessions, classified initially as VBR and only a few seconds later as VBR (low variation). Also, the traffic from VBR *Speex*, which is not covered by any group level signature, was sometimes classified by the signature for *Skype* proprietary VBR, especially in dataset 3, which pulled down the sensitivity rate. Furthermore, *G.722* and NWC VoIP sessions were classified as PCM since these codecs are based on PCM baseline, whose signature appeared first in the list of signatures. Nonetheless, we still chose to consider as false positives the VoIP sessions based on those codecs and classified as PCM. The results of sensitivity and specificity for each speech codec were included in Table 8 of the supplemental material. Additionally, since the dataset of the Politecnico di Torino contains only the flows that result from the VoIP sessions, it does not have any negative case. Hence, it does not make sense to calculate the specificity for this dataset. The percentage of the traffic in datasets 1, 2, 3, and 4, from each class of applications, that caused false positive cases is presented in Table 7 of the supplemental material, showing that they were caused by streaming, P2P file-sharing, and P2P streaming traffic.

The results show that the method is capable of classifying the traffic from VoIP sessions and identifying the used speech codec with interesting accuracy. Moreover, we obtained similar results for the same speech codec despite the fact that the datasets used for the performance evaluation contained traffic from VoIP sessions generated with different applications, transport protocols, and operating systems, showing the independence of the classifier from these factors.

The performance of the proposed classifier was also compared with the performance of three available classifiers: *l7-filter* [39], *l7-netpdlclassifier* [40], and *Tstat* [41]. The results and details of the performance evaluation of these three classifiers are described in subsection E.2 of the supplemental material. The results demonstrate that most classifiers have difficulty to identify the specific flows related with conversations, even when they are able to identify other flows of the VoIP application like signaling data. The three mechanisms seem to have more problems to identify traffic from VoIP sessions over TCP, as shown by the low sensitivity rates for dataset 4. The specificity rates for the same dataset are also lower, mostly due to the larger share of traffic from other P2P applications. *Tstat* seems to be more conservative in the identification of VoIP traffic, which also helps it to perform better in terms of false positive cases. The Polito dataset contains only 10 VoIP sessions, and therefore any misclassified flow has a strong negative impact in the sensitivity. Since the packets in this dataset do not contain the payload data, *l7-filter* was unable to identify any VoIP flow. Generally, the classifier proposed herein presents a better accuracy when used to distinguish

TABLE 3
Results of the performance evaluation of the VoIP classifier for the different levels of signatures.

Dataset	**Bit rate level**			**Group level**			**Codec level**		
	Sensitivity		Specificity	Sensitivity		Specificity	Sensitivity		Specificity
	first	second		first	second		first	second	
Dataset 1	92.31%	100.00%	99.97%	100.00%	100.00%	100.00%	84.62%	92.31%	100.00%
Dataset 2	92.86%	100.00%	99.99%	100.00%	100.00%	100.00%	78.57%	100.00%	100.00%
Dataset 3	100.00%	100.00%	99.98%	80.00%	80.00%	100.00%	93.34%	93.34%	100.00%
Dataset 4	96.97%	96.97%	99.51%	96.97%	96.97%	99.56%	84.85%	84.85%	99.99%
Polito	100.00%	100.00%	not applicable	100.00%	100.00%	not applicable	70.00%	100.00%	not applicable

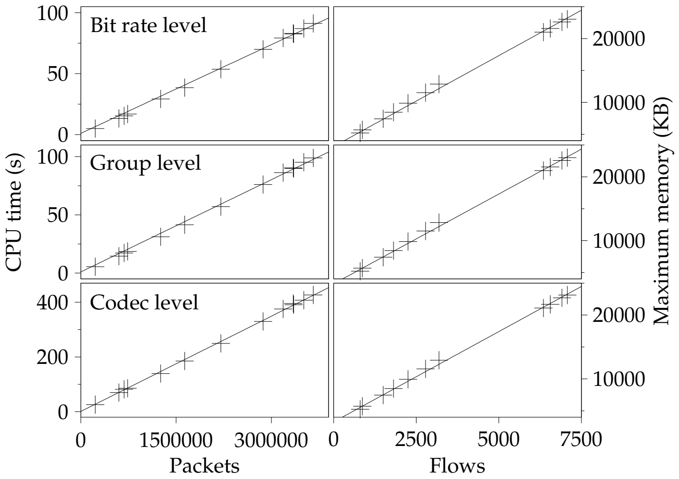


Fig. 4. Representation of the CPU time and memory consumption growing and the number of packets and flows for 13 trace files, considering packets with payload larger than 5 bytes.

between CBR, VBR, and VBR with low variation, and it is also able to make an accurate prediction of the codec used in each session. Furthermore, the accuracy of the identification of the flow of the real conversation is much higher than for the other classifiers.

As it happens with most traffic classifiers, the method used by the classifier may be bypassed if the target applications are able to modify the behavior matched by the signatures. Nonetheless, since the packet lengths depend on the speech codec used in a VoIP session, the pattern may not be so easily modified by the target applications. This subject is further discussed in subsection E.2 of the supplemental material.

In addition to the performance evaluation, an analysis of the computational resources used by the classifier has been made showing that the consumption of resources grows linearly with the analyzed data, as illustrated by Fig. 4. The details of the resource evaluation are presented in subsection E.3 of the supplemental material.

6 C

In this article, a new method for the identification of P2P VoIP traffic was described. The proposed mechanism is focused on the properties of the speech codec used in the VoIP session instead of the application and it aims to identify the flow used for the conversation rather than the signaling data. The traffic from several VoIP sessions, using many codecs and made using different applications was collected and analyzed to identify properties that could be used in the classification process. The lengths of the payloads presented different levels of heterogeneity for distinct codecs. Although the lengths of the packets have already been used in different ways, its level of heterogeneity has never been used for the classification of traffic in real-time. To the best of our knowledge, this is the first behavioral method capable of identifying the codecs used on a VoIP session. In order to quantify the level of heterogeneity and use it to identify traffic, an approach based on entropy was used. Its value was calculated by resorting to sliding windows with size of a constant number of packets. By doing so, it is possible to monitor the value of the entropy, in real-time, from the beginning of the flow to its end. The identification of VoIP sessions is made by using a set of behavioral signatures formed by an interval for the entropy, an interval for the length of the payload, and a minimum number of matches that should be reached for the traffic to be classified by the corresponding signature.

The performance of the proposed classifier was evaluated by resorting to aggregated traffic from multiple VoIP sessions, using different codecs and applications, and several P2P and non-P2P applications. The results showed that the classifier was capable of identifying the VoIP sessions with very good accuracy, performing better that the remaining analyzed tools. Furthermore, the mechanism was able to recognize the specific speech codec that was used with a sensitivity rate between 70.00% and 93.34%. Additionally, the resource consumption was also analyzed, showing that the resource usage grows linearly with the amount of the input data.

In the future, we plan to address the classification of traffic from other types of P2P applications. For P2P media streaming and file-sharing, the differences in the entropy are not so clear in the traffic from each flow, which prevents the use of the same approach for these types of P2P traffic. Furthermore, we expect to study the challenges inherent to the optimization of the classifier, to build an optimized prototype of the proposed classifier, and to test it in high-speed network scenarios.

A

This work was partially supported by University of Beira Interior, by *Instituto de Telecomunicações*, and by the portuguese *Fundação para a Ciência e a Tecnologia*, through the grant contract SFRH/BD/60654/2009 and the project TRAMANET: Traffic and Trust Management in Peer-to-Peer Networks with contracts PTDC/EIA/73072/2006 and FCOMP-01-0124-FEDER-007253.

R

[1] K. Suh, D. R. Figueiredo, J. Kurose, and D. Towsley, "Characterizing and detecting Skype-relayed traffic," in *Proc. 25th IEEE Int. Conf. Computer Communications (INFOCOM 2006)*, Barcelona, Spain, Apr. 2006, pp. 1–12.

[2] E. P. Freire, A. Ziviani, and R. M. Salles, "Detecting VoIP calls hidden in web traffic," *IEEE Trans. Netw. Service Manag.*, vol. 5, no. 4, pp. 204–214, Dec. 2008.

[3] D. Bonfiglio, M. Mellia, M. Meo, and D. Rossi, "Detailed analysis of Skype traffic," *IEEE Trans. Multimedia*, vol. 11, no. 1, pp. 117–127, Jan. 2009.

[4] J. F. Ransome and J. W. Rittinghouse, *Voice over Internet Protocol (VoIP) Security*. Digital Press, Nov. 2004, ch. VoIP Security Risks, pp. 181–233.

[5] J. Seedorf, "Security challenges for peer-to-peer SIP," *IEEE Netw.*, vol. 20, no. 5, pp. 38–45, Sep./Oct. 2006.

[6] R. Dantu, S. Fahmy, H. Schulzrinne, and J. Cangussu, "Issues and challenges in securing VoIP," *Elsevier Comput. Security*, vol. 28, no. 8, pp. 743–753, Nov. 2009.

[7] D. R. Kuhn, T. J. Walsh, and S. Fries, "Security considerations for voice over IP systems," National Institute of Standards and Technology, Gaithersburg, MA, USA, Tech. Rep. 800-58, Jan. 2005.

[8] T. Berson, "Skype security evaluation," Anagram Laboratories, Tech. Rep. ALR-2005-031, October 2005.

[9] J. Xin, "Security issues and countermeasure for VoIP," *White Paper, SANS Institute, Information Security Reading Room*, 2007.

[10] D. Bonfiglio, M. Mellia, M. Meo, D. Rossi, and P. Tofanelli, "Revealing Skype traffic: When randomness plays with you," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 37–48, Oct. 2007.

[11] D. Adami, C. Callegari, S. Giordano, M. Pagano, and T. Pepe, "Skype-Hunter: A real-time system for the detection and classification of Skype traffic," *Int. J. Commun. Syst.*, 2011.

[12] A. A. Khuther, "Performance analysis of voice codec for VoIP," Master's thesis, Universiti Teknologi Malaysia, Oct. 2008.

[13] J. V. P. Gomes, P. R. M. Inácio, M. M. Freire, M. Pereira, and P. P. Monteiro, "Analysis of peer-to-peer traffic using a behavioural method based on entropy," in *Proc. 27th IEEE Int. Performance Computing and Communications Conf. (IPCCC 2008)*, Austin, TX, USA, Dec. 2008, pp. 201–208.

[14] J. V. Gomes, P. R. M. Inácio, M. Pereira, M. M. Freire, and P. P. Monteiro, "Exploring behavioral patterns through entropy in multimedia peer-to-peer traffic," *The Computer Journal*, vol. 55, no. 6, pp. 740–755, Jun. 2012.

[15] B. Li, M. Ma, and Z. Jin, "A VoIP traffic identification scheme based on host and flow behavior analysis," *J. Netw. Syst. Manag.*, vol. 19, no. 1, pp. 111–129, Mar. 2011.

[16] Y. Yu, D. Liu, J. Li, and C. Shen, "Traffic identification and overlay measurement of Skype," in *Proc. Int. Conf. Computational Intelligence and Security*, Guangzhou, China, Nov. 2006, pp. 1043–1048.

[17] S. Ehlert and S. Petgang, "Analysis and signature of Skype VoIP session traffic," Fraunhofer FOKUS, Berlin, Germany, Tech. Rep. NGNI-SKYPE-06b, Jul. 2006.

[18] P. Svoboda, E. Hyytiä, F. Ricciato, M. Rupp, and M. Karner, "Detection and tracking of Skype by exploiting cross layer information in a live 3G network," in *Proc. 1st Int. Workshop Traffic Monitoring and Analysis (TMA '09)*, ser. LNCS, vol. 5537, Aachen, Germany, May 2009, pp. 93–100.

[19] F. Lu, X.-L. Liu, and Z.-N. Ma, "Research on the characteristics and blocking realization of Skype protocol," in *Proc. Int. Conf. Electrical and Control Engineering (ICECE 2010)*, Wuhan, China, Jun. 2010, pp. 2964–2967.

[20] D. Zhang, C. Zheng, H. Zhang, and H. Yu, "Identification and analysis of Skype peer-to-peer traffic," in *Proc. 5th Int. Conf. Internet and Web Applications and Services (ICIW 2010)*, Barcelona, Spain, May 2010, pp. 200–206.

[21] R. Dhamankar and R. King, "Protocol identification via statistical analysis (PISA)," *White Paper, Tipping Point*, 2007.

[22] P. Dorfinger, G. Panholzer, B. Trammell, and T. Pepe, "Entropy-based traffic filtering to support real-time Skype detection," in *Proc. 6th Int. Wireless Communications and Mobile Computing Conf. (IWCMC '10)*, Caen, France, Jun./Jul. 2010, pp. 747–751.

[23] J.-L. Costeux, F. Guyard, and A.-M. Bustos, "Detection and comparison of RTP and Skype traffic and performance," in *Proc. IEEE Global Telecommunications Conf. (GLOBECOM 2006)*, San Francisco, CA, USA, Dec. 2006, pp. 1–5.

[24] L. Lu, J. Horton, R. Safavi-Naini, and W. Susilo, "Transport layer identification of Skype traffic," in *Proc. Int. Conf. Information Networking (ICOIN 2007)*, ser. LNCS, vol. 5200, Estoril, Portugal, Jan. 2007, pp. 465–481.

[25] S. Molnár and M. Perényi, "On the identification and analysis of Skype traffic," *Int. J. Commun. Syst.*, vol. 24, no. 1, pp. 94–117, Jan. 2011.

[26] K.-T. Chen and J.-K. Lou, "Rapid detection of constant-packet-rate flows," in *Proc. 3rd Int. Conf. Availability, Reliability and Security (ARES 08)*, Barcelona, Spain, Mar. 2008, pp. 212–220.

[27] L. Jun, Z. Shunyi, X. Ye, and S. Yanfei, "Identifying Skype traffic by random forest," in *Proc. Int. Conf. Wireless Communications, Networking and Mobile Computing (WiCom 2007)*, Shanghai, China, Sep. 2007, pp. 2841–2844.

[28] P. A. Branch, A. Heyde, and G. J. Armitage, "Rapid identification of Skype traffic flows," in *Proc. 18th Int. Workshop Network and Operating System Support for Digital Audio and Video (NOSSDAV '09)*, Williamsburg, VA, USA, Jun. 2009, pp. 91–96.

[29] R. Alshammari and A. N. Zincir-Heywood, "Unveiling Skype encrypted tunnels using GP," in *Proc. IEEE Congress on Evolutionary Computation (CEC 2010)*, Barcelona, Spain, Jul. 2010, pp. 1–8.

[30] C.-C. Wu, K.-T. Chen, Y.-C. Chang, and C.-L. Lei, "Detecting VoIP traffic based on human conversation patterns," in *Proc. Principles, Systems and Applications of IP Telecommunications (IPTComm 2008)*, ser. LNCS, vol. 5310, Heidelberg, Germany, Jul. 2008, pp. 280–295.

[31] H. Zhang, Z. Gu, and Z. Tian, "Skype traffic identification based SVM using optimized feature set," in *Proc. Int. Conf. Information, Networking and Automation (ICINA 2010)*, vol. 2, Kunming, China, Oct. 2010, pp. 431–435.

[32] T. Yildirim and P. J. Radcliffe, "VoIP traffic classification in IPSec tunnels," in *Proc. Int. Conf. Electronics and Information Engineering (ICEIE 2010)*, vol. 1, Kyoto, Japan, Aug. 2010, pp. 151–157.

[33] B. Xu, M. Chen, C. Xing, and G. Zhang, "A network traffic identification method based on finite state machine," in *Proc. 5th Int. Conf. Wireless Communications, Networking and Mobile Computing (WiCom 2009)*, Beijing, China, Sep. 2009, pp. 1–4.

[34] N. M. Markovich and U. R. Krieger, "Statistical analysis and modeling of Skype VoIP flows," *Elsevier Comput. Commun.*, vol. 33, no. S1, pp. S11–S21, Nov. 2010.

[35] T. Okabe, T. Kitamura, and T. Shizuno, "Statistical traffic identification method based on flow-level behavior for fair VoIP service," in *Proc. 1st IEEE Workshop VoIP Management and Security (VoIP MaSe 2006)*, Vancouver, Canada, Apr. 2006, pp. 35–40.

[36] F. Liu, Z. Li, and J. Yu, "P2P applications identification based on the statistics analysis of packet length," in *Proc. Int. Symp. Information Engineering and Electronic Commerce (IEEC 2009)*, Ternopil, Ukraine, May 2009, pp. 160–163.

[37] C. V. Wright, L. Ballard, S. E. Coull, F. Monrose, and G. M. Masson, "Spot me if you can: Uncovering spoken phrases in encrypted VoIP conversations," in *Proc. IEEE Symp. Security and Privacy (SP 2008)*, Oakland, CA, USA, May 2008, pp. 35–49.

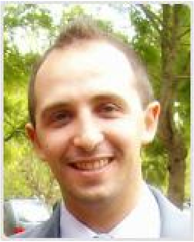
[38] K. C. Claffy, H.-W. Braun, and G. C. Polyzos, "A parameterizable methodology for Internet traffic flow profiling," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 8, pp. 1481–1494, Oct. 1995.

[39] L7-filter, application layer packet classifier for Linux. [Online]. Available: http://l7-filter.sourceforge.net

[40] Tools for L2-L7 traffic classification. [Online]. Available: http://netgroup.polito.it/research-projects/l7-traffic-classification/

[41] Tstat: TCP statistic and analysis tool. [Online]. Available: http://tstat.tlc.polito.it

João V. Gomes received the 5 years BSc degree in Computer Science and Engineering from University of Beira Interior, Portugal, in 2006. He worked as researcher in the research group of Nokia Siemens Networks Portugal, where he started pursuing the PhD degree in Computer Science and Engineering, in enterprise environment, and also as R&D engineer in the Operations & Business Software unit of the same company. In 2012, he received the PhD degree in Computer Science and Engineering from University of Beira Interior. Currently, he is a researcher in the Multimedia Signal Processing Group of *Instituto de Telecomunicações*. His main research interests include traffic monitoring and analysis, traffic classification, peer-to-peer computing, and cloud computing.



Pedro R. M. Inácio was born in Covilhã, Portugal, on February 24, 1982. He received the degree in Mathematics / Computer Science from the University of Beira Interior, Portugal, in 2005, with an overall classification of Good with Distinction (average of 17 in 20). In the same year, he was accepted for a Ph.D. programme at the Department of Computer Science of that university, but the Ph.D. work was performed in the enterprise environment of Nokia Siemens Networks Portugal S.A., under the umbrella of a collaboration protocol between the two entities and the portuguese Foundation for the Science and Technology. He was also a System Architect in the Broadband Access Business Unit at that company. He got the Ph.D. degree in Computer Science and Engineering in December 2009 with the defense of the thesis entitled “Study of the Impact of Intensive Attacks in the Self-Similarity Degree of the Network Traffic in Intra-Domain Aggregation Points”.

He is an Invited Assistant Professor at the University of Beira Interior since 2010, where he lectures subjects related with computer security, databases, software engineering, computer based simulation and computer networks to the Computer Science and Engineering and Technology and Information Systems courses. He is also a researcher of the *Instituto de Telecomunicações* (IT). His main research interests include algorithm development and optimization, network traffic monitoring and simulation, security mechanisms for information networks (namely key agreement protocols and cryptography), anomaly detection and software security.

Manuela Pereira received the 5-year B. S. degree in Mathematics and Computer Science in 1994 and the M. Sc. degree in Computational Mathematics in 1999, both from the University of Minho, Portugal. She received the Ph. D. degree in Signal and Image Processing in 2004 from the University of Nice Sophia Antipolis, France. She is an Assistant Professor at the Department of Computer Science of the University of Beira Interior, Portugal. Her main research interests include: multiple description coding, joint source/channel coding, image and video coding, wavelet analysis, information theory, image segmentation and real-time video streaming.

She has been the editor of 2 books and has authored or co-authored around 40 papers in international refereed journals and conferences. She served as a technical and program committee member for several IEEE journals as and conferences. She is also a member of the editorial review board of several International Journals.

Mário M. Freire received the five-year BS degree in Electrical Engineering and the two-year MS degree in Systems and Automatics in 1992 and 1994, respectively, from the University of Coimbra, Portugal. He received the PhD degree in Electrical Engineering in 2000 and the Habilitation title in Computer Science in 2007 from the University of Beira Interior, Portugal. He is a full professor of Computer Science at University of Beira Interior, which he joined in the fall of 1994. When he was a MS student at University of Coimbra, he was also a trainee researcher for a short period in 1993 in the Research Centre of Alcatel-SEL (now Alcatel-Lucent) in Stuttgart, Germany. His main research interests include multimedia networking and peer-to-peer systems, multimedia traffic analysis and synthesis, high-speed networks, and network security. He is the co-author of seven international patents, co-editor of eight books published in the Springer Lecture Notes in Computer Science book series, and author or co-author of about 120 papers in refereed international journals and conferences. He serves as a member of the editorial board of the ACM SIGAPP Applied Computing Review, serves as associate editor of the Wiley Journal on Security and Communication Networks, and served as editor of the IEEE Communications Surveys and Tutorials. He also served as a guest editor of two feature topics in IEEE Communications Magazine and of a special issue of Wiley International Journal of Communication Systems. He served as a technical program committee member for several IEEE international conferences and is co-chair of the Track on Networking of ACM SAC 2013. Dr. Mario Freire is a chartered engineer by the Portuguese Order of Engineers and is a member of the IEEE Computer Society and the IEEE Communications Society, and a member of the Association for Computing Machinery.

Paulo P. Monteiro received the diploma “Licenciatura” in Electronics and Telecommunications Engineering from the University of Aveiro in 1988, the M.Sc. in Electronic Engineering, from the University of Wales UK, in 1990 and the Ph.D. in Electrical Engineering, from the University of Aveiro, in 1999. Presently, he is Associate Professor at the University of Aveiro and Researcher at the *Instituto de Telecomunicações*. He has been at Nokia Siemens Networks since 2007 has division manager of Network Optimization unit (2012-2010) and Research Manager at Transport, Aggregation and Fixed Access (2009-2007). In 2002, he joined the Siemens S.A., Portugal, as Head of Research of Optical Networks. In October 2006, the department was transferred to Siemens Networks, which then has become part of the joint-venture Nokia Siemens Networks (NSN) in April 2007. In 1992, he joined the Department of Electronic and Telecommunications Engineering of University of Aveiro and the Optical Communications Group of Institute for Telecommunications as an Assistant Professor and Researcher, respectively. In 1999, he became an Auxiliary Professor at the University of Aveiro and he was promoted to Associate Professor in 2005. His main research interests include High Speed Electronics; Microwave Circuits; Optical Access and Core Networks. He has authored/co-authored of over than 18 patent applications as well as more than 300 refereed papers and conference contributions.