SkyTracer: Towards Fine-Grained Identification for Skype Traffic via Sequence Signatures

Zhenlong Yuan[*§], Cuilan Du[†], Xiaoxian Chen[¶], Dawei Wang[†] and Yibo Xue[‡§**]
[*]Department of Automation, Tsinghua University, Beijing, China
[†] Coordination Center of China (CNCERT/CC), Beijing, China
[‡] Tsinghua National Lab for Information Science and Technology, Beijing, China
[§] Research Institute of Information Technology, Tsinghua University, Beijing, China
[¶] School of Computer Science and Technology, Harbin University of Science and Technology, Harbin, China
yuanzl11@mails.tsinghua.edu.cn, dcl@isc.org.cn, {xiaoxianchen73, stonetools2008}@gmail.com, yiboxue@tsinghua.edu.cn

***Abstract*—Skype has been a typical choice for providing VoIP service nowadays and is well-known for its broad range of features, including voice-calls, instant messaging, file transfer and video conferencing, etc. Considering its wide application, from the viewpoint of ISPs, it is essential to identify Skype flows and thus optimize network performance and forecast future needs. However, in general, a host is likely to run multiple network applications simultaneously, which makes it much harder to classify each and every Skype flow from mixed traffic exactly. Especially, current techniques usually focus on host-level identification and do not have the ability to identify Skype traffic at the flow-level. In this paper, we first reveal the unique *sequence signatures* of Skype UDP flows and then implement a practical online system named *SkyTracer* for precise Skype traffic identification. To the best of our knowledge, this is the first time to utilize the strong sequence signatures to carry out early identification of Skype traffic. The experimental results show that *SkyTracer* can achieve very high accuracy at fine-grained level in identifying Skype traffic.**

***Index Terms*—Skype, Sequence Signature, Correlation-based Approach, Flow-level Identification**

I. Introduction

During the past decade, Skype has gained a tremendous popularity because of its good voice quality and secure communication mechanism. Due to its extensive application, network managers has an urgent requirement to identify Skype traffic accurately. However, researchers normally face severe difficulties in practical Skype traffic identification due to the following reasons: i) Topological complexity. Skype is a world-wide P2P VoIP network that consists of ordinary nodes (client), supernodes (SN), login servers, update servers and buddy-list servers. As a result, Skype would adopt different kinds of communication models and dynamic ports for information transmission, which leads to the complexities of Skype traffic. ii) Protocol complexity. Due to the broad range of features in Skype (e.g. voice-calls, instant messaging, file transfer and video conferencing), Skype would use different communication mechanisms for data transport, which leads to the complexities of Skype traffic as well. iii) Privacy complexity. Skype uses a variety of techniques to prevent its private protocol from being reverse-engineered. For this reason, traditional traffic identification techniques based on individual packet payload would not work any more.

Looking back upon the existing work, although a great number of researchers have put forward various methods for identifying Skype traffic, most of them are coarse-grained solutions and do not satisfy the following essential requirements: i) Fine-grained classification. A host is likely to run multiple network applications simultaneously, for instance, a user might use both Skype and MSN to chat with friends while downloading resources through Bittorrent. Specially, there are inevitably some other applications running in background on the same host as well, which makes it much harder to precisely classify applications from aggregated-traffic (i.e. at host-level). Therefore, the methods that make direct analysis on aggregated-traffic are not appropriate and it is crucial to identify Skype traffic at individual 5-tuple (i.e. protocol, source IP, source port, destination IP and destination port) flow level. ii) Early identification. As our purpose is to classify traffic for network management and performance optimization, it is essential to identify every single Skype flow with an early detection and thus Skype traffic could be immediately optimized for quality of service (QoS) improvement. iii) High performance. It is generally known that both fast classification speed and high identification accuracy are actually needed for a practical traffic classification system, especially for large-scale traffic classification. For example, for a backbone router at 230 Gbps [1], 1% of classification error rate will result in incorrect classification of 2.30 Gbps. This may heavily affect network administrators' actions and seriously degrade user experience.

In order to meet the above requirements, this paper first reveals the effective sequence signatures of Skype traffic, and thereby proposes a novel system to identify Skype traffic at the flow-level based on those sequence signatures. Unlike the traditional signatures, the sequence signatures are constructed by multiple one-bytes in a sequence of packets. Particularly, this novel system not only can gain very high identification accuracy at a fine-grained level but also can achieve an early identification with the first few packets in a flow. Main contributions of this paper include:

1) *Reveal the Effective Sequence Signatures*. Though in a

[**]Corresponding author. E-mail: yiboxue@tsinghua.edu.cn

number of studies [2], [3], researchers have claimed that there is no valid payload information that can be utilized for effective Skype traffic identification, this paper revealed that there are strong sequence signatures existing in the payloads of Skype traffic. To the best of our knowledge, this is the first time that sequence signatures are utilized for Skype traffic identification.

2) *Design a Practical Classification System*. Our goal is to achieve fine-grained identification for Skype traffic in practice. Starting with the sequence signatures matching and adopting the correlation-based approaches, we designed and implemented a practical system named *SkyTracer* for online Skype traffic classification. Particularly, *SkyTracer* contains two modules: Skype-node identification and fine-grained identification.
3) *Perform an Experimental Evaluation*. In order to validate the efficiency of *SkyTracer*, this paper carried out a comprehensive experiment on a variety of traces. Experimental results showed that 98.93% *precision* and 99.54% *recall* at flow-level are achieved. Besides, the experiments as well demonstrated that *SkyTracer* can gain a very high accuracy with just the first few packets in a flow, which proved the superiority of *SkyTracer*.

The rest of this paper is organized as follows. The related work is introduced in Section II. Section III presents the sequence signatures of Skype traffic. Section IV describes the design and implementation of *SkyTracer*. Experimental evaluation and results are shown in section V. Section VI concludes the paper.

II. Related Work

Traditional port-based approaches that rely on well-known port numbers are no longer valid for Skype traffic identification. Therefore, along with the wide spread of Skype application, various kinds of identification techniques have been proposed. In this section, the previous methods for identifying Skype traffic are divided into two parts: signature-based methods [4]–[6] and behavior-based methods [2], [3], [7]–[15].

A. Signature-based Methods

Adami et al. [4] proposed a real-time algorithm named Skype-Hunter to detect and classify Skype traffic. In fact, Skype-Hunter was designed based on some signatures and employed part of behavior-based techniques as well. Experimental results showed that Skype-Hunter outperformed the 'classical' statistical traffic classifiers as well as the state-of-the-art *ad hoc* Skype classifier. Yu et al. [5] presented an effective tool named Super Nodes Collecting and Probing Platform (SNCPP), which was designed to measure the overlay of Skype network based on signatures. Ehlert et al. [6] developed traffic signatures that allowed a third party monitoring entity to detect the usage of the Skype application. Note that the signatures applied in the above methods might be the payload-based or feature-based.

However, these current signature-based methods usually take some simple or commona characters (e.g. *0x170301*) as signatures and then use them in combination with other approaches to perform traffic identification. Therefore, these signatures are generally not robust and unique enough, which might result in a high misclassification rate in practical use. Considering such a situation, it is better to discover much more strong signatures for Skype traffic identification.

B. Behavior-based Methods

By analyzing various aspects of the Skype protocol under different network setups, Baset et al. [16] revealed that Skype traffic has many self-behavior characteristics, and provided an overview of Skype's design and functions. In general, the behavior-based methods are grouped into two main categories: host-level identification [2], [3], [7]–[10] and flow-level identification [11]–[15]. Particularly, Bonfiglio et al. [13] devised a method that successfully tackled the problem of Skype voice traffic identification. Following that in [10], they achieved to gather a deeper understanding through dissecting the data and signaling traffic generated by Skype. Similarly, both [2] and [8] proposed approaches for accurately recognizing P2P applications (including Skype) based on behavior characteristics. In order to characterize the nature of relayed traffic, Suh et al. [9] proposed a methodology for detection of Skype-relayed traffic based on thresholds with several defined metrics. In these work [3], [7], [11], [12], [14], [15], note that "packet size" is a major characteristic and really helps a lot.

However, for the host-level identification methods that generally deal with aggregated traffic, on one hand, it is not realistic to assume that a host is always running just one application, on the other hand, due to the prevalence of asymmetric routing, it is hard to obtain the complete traffic data that a host generated or received. Therefore, in a real-life situation, host-level identification methods are often difficult to be applied for practical use. For the flow-level identification methods that work based on individual flow information, they mostly utilize the statistical features such as "packet size" to judge whether a flow belongs to Skype or not. We should be aware that these statistical features are not so robust and unique as payload-based signatures and might be affected by different Skype codecs. What's more, researchers would never know if there is another application possessing the same statistical features as Skype.

III. Sequence Signatures

In this paper, the sequence signatures of Skype UDP flows are discovered with the help of an automated packet-sequence signature construction (APSC) system in our previous work [17]. This system not only automatically generate traditional signatures from individual packet payloads but also construct packet sequence signatures based on payloads or features from a sequence of packets in application flows. As shown in TABLE I, the sequence signatures of Skype UDP flows are represented as regular expressions. In particular, all the bytes in the regular expression are represented for the **third** byte

TABLE I
Sequence Signatures of Skype UDP Flows

$^\wedge\backslash x02 + [\backslash x0d\backslash x1d\backslash x2d\backslash x3d\backslash x4d\backslash x5d\backslash x6d\backslash x7d] * \$$
$^\wedge\backslash x02 + [\backslash x0f\backslash x1f\backslash x2f\backslash x3f\backslash x4f\backslash x5f\backslash x6f\backslash x7f] + [\backslash x0d\backslash x1d\backslash x2d\backslash x3d\backslash x4d\backslash x5d\backslash x6d\backslash x7d] + \$$
$^\wedge\backslash x02 + [\backslash x05\backslash x15\backslash x25\backslash x35\backslash x45\backslash x55\backslash x65\backslash x75] + [\backslash x0d\backslash x1d\backslash x2d\backslash x3d\backslash x4d\backslash x5d\backslash x6d\backslash x7d] + \$$

of payloads in a continuous sequence of packets that belong to a Skype UDP flow. Specially, the symbol “∧” in a regular expression represents for the starting of a flow, that is, to a Skype UDP flow, if we compose all the third bytes of payloads in a continuous sequence of packets into one string according to the packet sequence number, then this string must match with one of the three sequence signatures shown in TABLE I, otherwise it will not be a Skype flow. In the later section V, our experimental evaluation will demonstrate that the three sequence signatures are robust and unique enough for Skype UDP traffic classification.

Here, in order to have a detailed analysis on the special sequence signatures, this paper performs a further statistics about the packet information in Skype UDP traffic. TABLE II shows a Skype traffic trace that was generated under various environments, such as different operating systems and different Skype versions. Additionally, this trace contains a variety kind of Skype service traffic, including voice-calls, instant messaging, file transfer, SkypeOut and video conferencing, etc. The detailed statistics are exhibited in TABLE III, from which we can see that the third byte value of packet payloads in Skype UDP flows has a limited scope (i.e. *0x02*, *0x0D~0x7D*, *0x0F~0x7F* and *0x05~0x75*). Particularly among them, “*0x0D~0x7D*” are nearly uniform distribution while “*0x0F~0x7F*” and “*0x05~0x75*” rarely appear. Besides, we can also see that the last four bits of “*0x0D~0x7D*” are the same and might stand for some special message.

TABLE II
The Attributes of Skype Trace File

Protocol	Payload Size	Flows	Packets
UDP	18089KB	227	65197
TCP	170KB	99	2394
TCP & UDP	18259KB	326	67591

It is generally known that Skype secures the whole communication inside Skype network by virtue of strong encryption. So why is there such sequence signature? In this paper, we attribute this characteristics to the following two reasons: i) The most important of Skype is to offer users a high-quality way to communicate with each other. So an inevitable challenge for Skype is to find a right trade-off between communication security and communication quality. Skype should have been able to encrypt the whole payload information (includes the third byte value), but for some reasons such as speeding the process of parsing packet information and then improving the communication quality, Skype just design a proprietary protocol on the top of transport layer. Naturally, this proprietary protocol will have its own proper structure and this structure will set some specific flags at some positions. For example, the third byte of packet payloads in Skype UDP flows are probably a specific flag. ii) Although a flag in Skype protocol structure might be regular and distinct, it is hardly ever considered as a significant component part of a strong signature. Previous research [13], [18], [19] have investigated the value of the third byte in a Skype UDP flow and discovered the specific flag ‘*0x02*’, ‘*0x0D*’ and ‘*0x07*’. However, they did not compose all the flags in a sequence of packets into a string and then utilized this string (i.e. sequence signature) to identify Skype traffic. Just like the behavior-based techniques that exploit the information of multiple packets for traffic identification, we should concentrate on not only the signatures existed in a single packet but also the sequence signatures existed in a sequence of packets.

TABLE III
Signatures of Skype UDP Packets

Protocol	Signature	Positon	Frequency	Binary
UDP	02	0x03	537	00000010
UDP	0D	0x03	8051	00001101
UDP	1D	0x03	8174	00011101
UDP	2D	0x03	8128	00101101
UDP	3D	0x03	7925	00111101
UDP	4D	0x03	8057	01001101
UDP	5D	0x03	8220	01011101
UDP	6D	0x03	7991	01101101
UDP	7D	0x03	8094	01111101
UDP	0F~7F	0x03	14	0***1101
UDP	05~75	0x03	6	0***0101

IV. SkyTracer

In this section, we design and implement a practical system name *SkyTracer* for online Skype traffic identification. It is obvious that the sequence signatures discussed in Section III have been sufficient for identifying the Skype UDP traffic. So the core module of *SkyTracer* is how to identify the other Skype traffic (i.e. Skype TCP traffic) based on the sequence signatures by adopting the correlation-based approaches. Fig.1 illustrates the architecture of *SkyTracer*. As other typical traffic classification systems, we assume that all ingress/egress traffic for a domain (e.g., an ISP) goes through a gateway router. This gateway mirrors all traffic as the input for our system. Particularly, *SkyTracer* mainly consists of two modules: Skype-node Identification module and Fine-grained Identification module.

A. Skype-node Identification Module

1) *Login Signal Detection*. Based on our observations, a signaling message that indicates the login process will

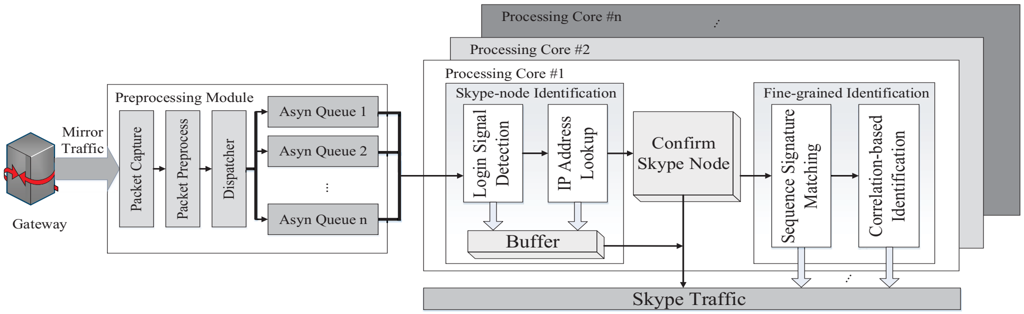

Fig. 1. The Architecture of SkyTracer

be immediately generated while launching Skype. In particular, this signaling message is transmitted as a UDP flow and the third bytes of all the packet payloads in this flow are always ‘*0x02*’. In light of that, once a UDP flow comes, we can examine the third bytes of all the packet payloads to see whether it might be the signaling message of Skype login. Only if there is such a UDP flow, the host could be a Skype user. By this way, *SkyTracer* can filter out a lot of traffic that definitely not belong to Skype application.

2) *IP Addresses Lookup*. After the above process, Skype will perform some other sophisticated communication processes that interact with a number of Skype servers for transmitting information. For example, according to our study, Skype must open an authenticated TCP connection to any one of the 225 IP addresses (the destination port is 33033) shown in TABLE IV. Therefore, we can check the destination IP of traffic flows to see if there is such a connection to Skype server and thus recognize whether a host is launching Skype. Besides that, we have collected a number of official IP addresses which are responsible for providing various kinds of Skype common service. By matching these IP addresses, the related Skype TCP traffic can be identified as well.

B. Fine-grained Identification Module

1) *Sequence Signature Matching*. To classify the Skype UDP flows from the aggregated traffic, the sequence signatures presented in Section III are utilized to match with each UDP flow. For example, if a UDP flow matched with the first one of the three sequence signatures, it means that the string which is composed by the third bytes of all the packet payloads in this flow is starting with some ‘*0x02*’ and ending with ‘$0x0D|0x1D|0x2D|0x3D|0x4D|0x5D|0x6D|0x7D$’. In particular, a UDP flow that matches with any one of the three sequence signatures will be classified as Skype traffic. In practical use, we load a regex engine (DFA) and record the DFA state number for each flow after some packet comes.

2) *Correlation-based Identification*. In our study, Skype uses both TCP (actually it is SSL/TLS) and UDP simultaneously for providing service, including the voice calls, skypeOut, chat, video conferencing, file transfer etc. Particularly, the TCP flows are responsible for transmitting control signals while the UDP flows are responsible for transmitting actual service data. Therefore, while two Skype-nodes are communicating with each other, the related TCP and UDP flows would have the same source and destination IP addresses. As a Skype UDP flow is identified with the strong sequence signatures, then the correlated Skype TCP flow which has the same source and destination IP addresses can be identified as well.

V. Experimental Evaluation

A. Datasets and Metrics

The traffic trace should be as extensive as possible and should generated under various environments. In this sec-

TABLE IV
IP Address List of Skype Authentication

IP Range	Except IP	Count	Organization	Country
64.4.23.140~64.4.23.166	64.4.23.163~64.4.23.164	25	Microsoft	United States
65.55.223.12~65.55.223.38	65.55.223.35~65.55.223.36	25	Microsoft	United States
111.221.74.12~111.221.74.38	111.221.74.35~111.221.74.36	25	Microsoft	Singapore
111.221.77.140~111.221.77.166	111.221.77.163~111.221.77.164	25	Microsoft	Singapore
157.55.56.140~157.55.56.166	157.55.56.163~157.55.56.164	25	Microsoft	United States
157.55.130.140~157.55.130.166	157.55.130.163~157.55.130.164	25	Microsoft	United States
157.55.235.140~157.55.235.166	157.55.235.163~157.55.235.164	25	Microsoft	United States
157.56.52.12~157.56.52.38	157.56.52.35~157.56.52.36	25	Microsoft	United States
213.199.179.140~213.199.179.166	213.199.179.163~213.199.179.164	25	Microsoft	Ireland

tion, the traffic traces are generated under various conditions, such as operating systems (Windows, Linux and MacOS), Skype versions and network conditions (wire and wireless). TABLE V shows the datasets which are used for evaluation. Particularly, the *Dataset2* was gathered with a separate set of traces and contains various types of application traffic except Skype, such as HTTP, SMTP, FTP, SSH, MSN, Gtalk, Bittorrent, eMule, Thunder, TencentQQ and so on.

TABLE V
Datasets for Evaluation

Trace File	Trace Size	UDP Flows	TCP Flows	Traffic
Dataset1	625MB	4191	1729	Skype
Dataset2	597GB	8134186	6477721	No Skype

This paper uses three standard metrics to quantify the performance of classification: *Precision*, *Recall*, and *F-Measure*. *Precision*: Percentage of truly Skype flows among those classified as Skype. *Recall*: Percentage of Skype flows that are correctly classified as Skype. *F-Measure*: An evenly weighted combination between precision and recall, which is defined as:

$$F\text{-}Measure = \frac{2 * Precision * Recall}{Precision + Recall}$$

B. Results

TABLE VI
Identification Results of the Mixed Traffic

Protocol	Precision %	Recall %	F-Measure %
TCP	97.37	98.61	97.99
UDP	99.57	99.93	99.75
ALL	98.93	99.54	99.23

After labeling the traffic flows in *Dataset1* and *Dataset2* respectively, we mixed them together for performance evaluation. From TABLE VI, we can see that a high accuracy for Skype traffic identification is achieved successfully. Besides that, in order to satisfy the early identification requirement, we perform an evaluation on the correlation between number of packets (matched with sequence signatures in a UDP flow) and the two main metrics (*Precision* and *Recall*) as shown in Fig.2. We can see that *SkyTracer* can work very well with matching only the first few packets (e.g. five or six) in a flow.

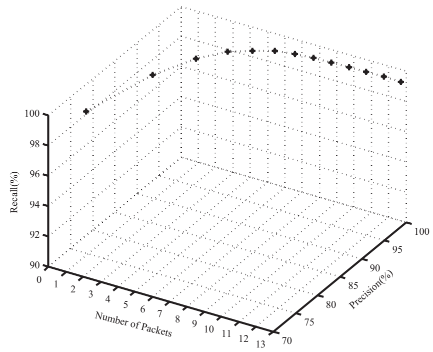

Fig. 2. Number of Packets vs. *Precison* and *Recall*

VI. Conclusion

In this paper, we have first revealed the strong *sequence signatures* in Skype UDP flows and then implemented a practical online system named *SkyTracer* for identifying the whole Skype traffic. Different from other typical traffic classification techniques for Skype application, *SkyTracer* can identify each Skype flow at a fine-grained level with an early detection, which is crucial for network management and performance optimization. The experimental results on *SkyTracer* show that 98.93% *precision* and 99.54% *recall* are achieved while classifying 625MB Skype traffic from 598GB mixed traffic.

Acknowledgement

This work was supported by the National Key Technology R&D Program of China under Grant No.2012BAH46B04.

References

[1] *Statistical Report on Internet Development in China*. [Online]. Available: http://www1.cnnic.cn/IDR/ReportDownloads/

[2] C. Wu, K. Chen, Y. Chang, and C. Lei, "Peer-to-peer application recognition based on signaling activity," in *Proc. of IEEE ICC*, 2009.

[3] M. Perényi, A. Gefferth, T. Dang, and S. Molnár, "Skype traffic identification," in *Proc. of IEEE GLOBECOM*, 2007.

[4] 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," *International Journal of Communication Systems*, vol. 25, no. 3, pp. 386–403, 2012.

[5] Y. Yu, D. Liu, J. Li, and C. Shen, "Traffic identification and overlay measurement of skype," in *Proc. of IEEE International Conference on Computational Intelligence and Security*, 2006.

[6] S. Ehlert, S. Petgang, T. Magedanz, and D. Sisalem, "Analysis and signature of skype voip session traffic," in *Proc. of CIIT*, 2006.

[7] J. Costeux, F. Guyard, and A. Bustos, "Detection and comparison of rtp and skype traffic and performance," in *Proc. of IEEE GLOBECOM*, 2006.

[8] H. Wu, N. Huang, and G. Lin, "Identifying the use of data/voice/video-based p2p traffic by dns-query behavior," in *Proc. of IEEE ICC*, 2009.

[9] K. Suh, D. Figueiredo, J. Kurose, and D. Towsley, "Characterizing and detecting relayed traffic: A case study using skype," in *Proc. of IEEE INFOCOM*, 2006.

[10] D. Bonfiglio, M. Mellia, M. Meo, N. Ritacca, and D. Rossi, "Tracking down skype traffic," in *Proc. of IEEE INFOCOM*, 2008.

[11] P. Santiago del Rio, J. Ramos, J. Garcia-Dorado, J. Aracil, A. Cuadra-Sanchez, and M. Cutanda-Rodriguez, "On the processing time for detection of skype traffic," in *Proc. of IEEE IWCMC*, 2011.

[12] P. Branch, A. Heyde, and G. Armitage, "Rapid identification of skype traffic flows," in *Proc. of ACM NOSSDAV*, 2009.

[13] D. Bonfiglio, M. Mellia, M. Meo, D. Rossi, and P. Tofanelli, "Revealing skype traffic: when randomness plays with you," in *Proc. of ACM SIGCOMM*, 2007.

[14] J. Gomes, M. Pereira, M. Freire, P. Monteiro *et al.*, "Identification of peer-to-peer voip sessions using entropy and codec properties," *IEEE Transactions on Parallel and Distributed Systems*, 2012.

[15] M. Korczynski and A. Duda, "Classifying service flows in the encrypted skype traffic," in *Proc. of IEEE ICC*, 2012.

[16] S. Baset and H. Schulzrinne, "An analysis of the skype peer-to-peer internet telephony protocol," in *Proc. of IEEE INFOCOM*, 2006.

[17] Z. Yuan, Y. Xue, and Y. Dong, "Harvesting unique characteristics in packet sequences for effective application classification," in *Proc. of IEEE CNS*, 2013.

[18] L. Ptácek, "Analysis and detection of skype network traffic," *Masaryk University Faculty of Informatics, Master Thesis, Brno, Czech Republic, Spring*, 2011.

[19] A. Houmansadr, C. Brubaker, and V. Shmatikov, "The parrot is dead: Observing unobservable network communications," in *Proc. of IEEE S&P*, 2013.