Digging into HTTPS: Flow-Based Classification of Webmail Traffic

Dominik Schatzmann
ETH Zurich
schatzmann@tik.ee.ethz.ch

Wolfgang Mühlbauer
ETH Zurich
muehlbauer@tik.ee.ethz.ch

Thrasyvoulos Spyropoulos
ETH Zurich
spyropoulos@tik.ee.ethz.ch

Xenofontas Dimitropoulos
ETH Zurich
fontas@tik.ee.ethz.ch

ABSTRACT

Recently, webmail interfaces, e.g., Horde, Outlook Web Access, and webmail platforms such as GMail, Yahoo!, and Hotmail have seen a tremendous boost in popularity. Given the importance of e-mail for personal and business use alike, and its exposure to imminent threats, there exists the need for a comprehensive view of the Internet mail system, including webmail traffic.

In this paper we propose a novel, *passive* approach to identify webmail traffic solely based on *network-level data* in order to obtain a *comprehensive* view of the mail system. Key to our approach is that we leverage *correlations* across protocols and time to introduce novel features for HTTPS webmail classification: First, webmail servers tend to reside close to legacy IMAP and POP mail servers, which are easy to identify. Second, the usage of webmail services results in distinct patterns on sessions' duration and on the diurnal/weekly traffic usage profile. Third, traffic flows to webmail platforms exhibit inherent periodicities since AJAX-based clients periodically check for new messages. We use these features to build a simple classifier and detect webmail traffic on real-world NetFlow traces from a medium-sized backbone network.

We believe that the major contribution of this paper – exploring a set of new features that could classify applications that run over HTTPS ports solely based on NetFlow data – will stimulate more general advance in the field of traffic classification.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations

General Terms

Measurement, Algorithms

Keywords

Traffic classification, HTTPS traffic, webmail, flow-level data

IMC'10, November 1–3, 2010, Melbourne, Australia.

1. INTRODUCTION

Speculations that web browsers could gradually supplant traditional operating systems as the default platform for user applications [21] go back to the first browser war in the mid 90's. According to recent reports [6] many Internet users prefer to access their e-mail via a web-based (webmail) interface such as Horde [11], Outlook Web Access (OWA) [25], or a webmail platform as provided by GMail, Yahoo!, or Hotmail. Similar trends also hold for many other applications, such as video, VoIP, and instant messaging, presently making browsers more than ever before the new operating system.

This trend however is detrimental for the field of Internet traffic classification, which over the past years has seen a plethora of proposals [16]. Evidently, *port-based* approaches are insufficient since many of today's applications use port 80 or HTTPS port 443 [23]. *Signature-based* solutions [5,9,14,23,30], which generally assume packet-level traces, fail if payload encryption is used, raise privacy or legal issues, and may not scale well enough to obtain a network-wide characterization of traffic. Even with hard to obtain, unencrypted full-packet traces, it is very difficult to identify the applications in use [8]. *Statistics-based* (e.g., [1, 2, 7, 19, 20, 22, 27]) and *host-behavior-based* [13–15] techniques avoid payload inspection. The former rely mainly on flow-level features such as flow duration, number and size of packets per flow and classify traffic flows. In our extensive experiments, we have found that flow-based features are not sufficient to classify individual HTTPS flows as webmail or not. Finally, host-based approaches (e.g., BLINC [14]) attempt to classify a host directly based on its communication pattern, and are the closest in spirit to our approach. Yet, to our best knowledge, host-based approaches have not been applied before to HTTP(S) traffic classification.

As a first step towards deciphering HTTP(S) traffic, we, in this work, address the challenging problem of extracting HTTPS-based webmail traffic from coarse-grained NetFlow data. Given its importance for personal and business use, and its exposure to imminent threats [3, 12], there exists a need for a comprehensive view of the Internet mail system [10, 26, 28], including webmail traffic. Measuring webmail traffic can enhance our understanding of shifting usage trends and mail traffic evolution. Surprisingly, we find that more than 66% of all HTTPS flows observed at the border routers of the SWITCH backbone network [31] are related to webmail. With GMail having recently enabled HTTPS by default (and other platforms expected to follow), this number will only increase in the future.

The most important contribution in this work is the introduction and evaluation of three novel features for identifying HTTPS

mail traffic. Key to our approach is that we leverage *correlations across protocols and time*: (i) The Internet mail system is an interplay of multiple protocols (SMTP, IMAP, POP, webmail). We find that webmail servers tend to reside in the vicinity of legacy SMTP, IMAP, and POP servers, which can be identified easily and reliably [17] using port numbers, thus providing significant hints for detecting webmail servers. (ii) Moreover, clients of a mail server share certain characteristics (e.g., usage patterns) irrespective of the used mail delivery protocol, i.e., IMAP, POP, or webmail. (iii) Finally, webmail traffic exhibits pronounced periodic patterns due to application timers (and more generally AJAX-based technologies). This paper shows that these features can be harvested solely from coarse-grained NetFlow data to classify webmail traffic.

To get a first impression of the ability of the above features to uncover webmail traffic, we train a simple classifier based on a (manually) pre-labeled set of hosts. Although finding the best possible classifier is beyond the scope of this paper, our simple classifier already exhibits 93.2% accuracy and 79.2% precision in detecting HTTPS servers within SWITCH, which is remarkable given that we rely solely on NetFlow data. Moreover, our work shows that we can also effectively detect webmail traffic (e.g., GMail) towards mail servers *outside* the SWITCH network, for which only a small sample of their total traffic is visible.

As a final note, we expect our methodology to stimulate advance in the field of traffic classification in general. For example, for our third feature we distinguish between "machine-generated" and "user-invoked" traffic based on network-level data, which is a very general method that can be used for several different types of detection problems.

The rest of this paper is structured as follows. Section 2 explains our data set we have used. Section 3 introduces and discusses features to discriminate mail-related traffic from other web traffic. Then, Section 4 describes how we construct a classifier based on the features we identify and presents our first results. Finally, we give an overview of related work in Section 5 and conclude in Section 6.

2. DATA SETS AND GROUND TRUTH

Throughout this paper we will rely on NetFlow traces coming from SWITCH [31], a medium-sized ISP. SWITCH is a backbone operator connecting approximately 30 Swiss universities, government institutions, and research labs. For our studies we have collected a trace in March 2010 that spans 11 days and contains unsampled flows summarizing all traffic crossing the borders of SWITCH. This results in 50 to 140 million NetFlow records per hour, with approximately 3% being HTTPS. Over the 11 days, we observe more than 1 billion HTTPS records.

Instead of analyzing individual flows, we mainly consider *sessions*. Our flow collectors break up long-lived flows into smaller fragments. Therefore, we merge flows sharing the same 5-tuple (source address and port, destination address and port, transport protocol identifier) if traffic exchange does not stay silent for more than 900s. Moreover, we group merged flows with the same IP addresses into a session if they are not more than $1,800$s apart in time. These values have been determined empirically, and found to work well in the past.

To train a classifier and to verify the accuracy of our classification in Section 4 we need a labelled data set. To this end, we first extract from a 1-hour trace recorded on Monday noon 2010-03-15 the 500 most popular internal HTTPS hosts/sockets (`top500`) in terms of unique IP addresses they communicate with. Then, we manually access each of the 500 obtained IP addresses with a web browser and determine the host type (e.g., Horde, OWA). In cases where this does not work (e.g., Skype), we use `nmap`. Table 1 summarizes the results of this time-consuming task.

class	type	# servers	# flows (mil.)
mail		77 (15.4%)	362.4 (66.0%)
	OWA	52	172.1
	Horde	10	84.9
	others	15	105.4
non mail		398 (79.6%)	159.0 (29.0%)
	WWW	137	88.9
	Skype	153	45.9
	VPN	15	8.3
	other	93	15.9
unknown		25 (5%)	27.7 (5.0%)
total		500	549.1

Table 1: `top500` data set

The fact that 66% of the observed HTTPS flows are related to Outlook Web Access, Horde, or other webmail interfaces (e.g., CommuniGate, SquirrelMail) is surprising and emphasizes the high popularity of webmail. Although the numbers may vary, we believe that our finding also holds for networks other than SWITCH. After all, e-mail is an omnipresent service in today's Internet. While we cannot label 5% of the `top500` hosts, we further differentiate 398 non-mail data sources into WWW content (e.g., e-shops, e-learning platforms), Skype, VPN services, etc.

3. FEATURES

We are now ready to discuss the features we propose for discriminating between mail related web traffic and other HTTPS traffic.

As mentioned earlier, classifying individual flows relying on flow-level statistics alone (e.g., packets and bytes per flow) does not work well. For example, cross-checking the distribution of number of bytes per flow does not reveal any significant differences between webmail and other HTTPS traffic flows.

An important challenge for feature extraction is to overcome the inherent heterogeneity of webmail traffic caused by the high number of different webmail implementations.

The features we present in the following are based on two main observations: first, the network-wide view provided by our flow-based data set allows to leverage correlations across hosts and protocols; second, periodic patterns due to regular timeouts are visible in flow data and provide useful application fingerprints.

To this end, this section sketches three approaches for distinguishing webmail from other HTTPS traffic. The key ideas are as follows: (i) classical mail services (POP, IMAP, SMTP) are frequently in the vicinity of webmail services (see Section 3.1), (ii) the client base of legacy mail services and the clients of its associated webmail service share common behavior in terms of user activity (Section 3.2), and (iii) traffic generated by AJAX-based webmail clients shows a pronounced periodicity that we can leverage to extract webmail traffic (Section 3.3). The following subsections explain these methods in detail, discuss their efficiency, and describe how to obtain features that can be used as input for a classifier.[1]

All three feature categories that we present in the following are broad in the sense that the key ideas are not limited to webmail classification. We believe that our features can be more generally applied towards demultiplexing HTTP(S) traffic into individual appli-

[1]We stress here that simplicity and interpretability, rather than optimality, have been our guiding principles in translating our observations into (scalar) features, used for our preliminary classification results.

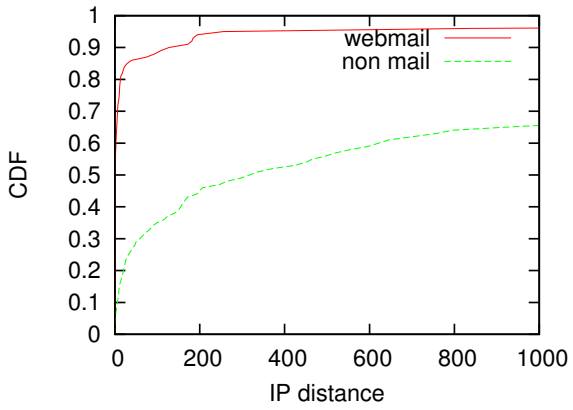

Figure 1: Distance to closest known legacy mail server.

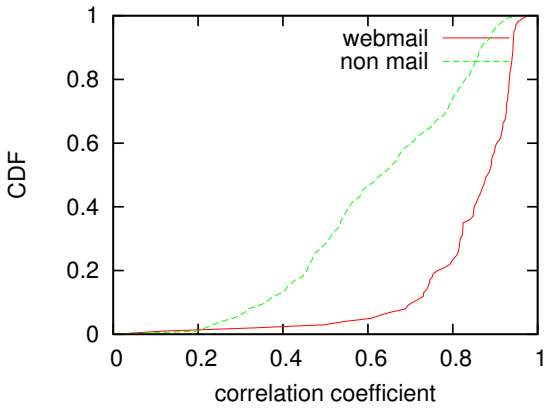

Figure 2: Correlation between activity profiles of web (mail or non-mail) servers and their closest legacy mail server.

cations. For example, inferring periodicity of sessions could translate into a powerful tool for distinguishing between "user-invoked" and "machine-generated" (e.g., AJAX) traffic and for extracting flow-level signatures associated with different applications. Also, studying the mix of well-known services within a subnet may provide hints for the existence of other unknown services in the same subnet. Finally, although we will focus on HTTPS traffic when introducing our features (motivated also by the earlier observations about webmail traffic), we stress that the techniques we describe are applicable to both HTTP and HTTPS flow traces.

3.1 Service proximity

The Internet mail system relies on the interplay of multiple services or protocols, including SMTP, IMAP, POP, and webmail. We have observed that if there exists a POP, IMAP, or SMTP server within a certain domain or subnet, there is a high chance to find a webmail server in the same subnet. This is often due to network planning reasons. Since legacy mail delivery traffic and respective servers can be easily and reliably identified using port-based classification [17], we can use their existence to infer webmail servers in the vicinity.

To verify our assumption of server proximity, we detect all mail servers in the SWITCH network relying on port numbers. Overall, we find 300 SMTP, 140 IMAP/S, and 176 POP/S servers. Moreover, we use our `top500` data set, see Section 2. For every HTTPS server we compute the IP address distance, which is simply the integer difference between two IP addresses towards the closest mail server. Our assumption is that legacy mail servers are closer to webmail than other HTTPS servers. Figure 1 shows the observed distance for webmail and non-mail HTTPS servers (e.g. WWW, VPN, Skype) found among the `top500` servers.

We observe that almost 60% of the webmail servers have a distance very close to the minimum, i.e., smaller than 10 IP addresses, while only 5% of the non-mail servers are in this range. Moreover, more than 90% of the webmail servers have a distance smaller than 200 IP addresses. These numbers indicate that webmail servers are substantially more likely than non-mail servers to be in the neighborhood of a legacy mail server.

3.2 Client base behavior

While Section 3.1 studies direct correlations between servers (i.e., how close they are), here we propose to analyze the entire client base of a server. We have found pronounced patterns for the following two properties: (i) the daily and weekly access patterns of HTTPS servers (Section 3.2.1), and (ii) the duration of client connections (Section 3.2.2).

3.2.1 Daily and weekly profile

It is well-known that Internet traffic as a whole has diurnal and weekly patterns. Different applications may also have their own specific access patterns. For example, people access their e-mail frequently and often in a scheduled way, i.e., first thing in the morning, whereas they access other web applications, such as online banking or Skype, in a substantially different way. More importantly, we expect client activity to be more balanced during the day for servers that have a worldwide user base (e.g., a web page) than for webmail servers with a local user base. For example, everyone can access the website of ETH Zurich, but only a limited group of people can use the webmail platform of ETH Zurich.

To quantify these differences, we use the activity profile of known IMAP and POP servers as a reference point and evaluate how it compares with the activity profile of unknown web servers in their vicinity. Our expectation is that IMAP and POP diurnal and weekly patterns will be more similar to webmail than to non-mail server patterns. In line with the observations of Section 3.1, we compare an unknown server to the closest legacy mail server in the same subnet. If there is no such server, we compare to the highest-volume legacy mail server in the same autonomous system.

Specifically, to compare two activity profiles, we partition a given time interval (in our case one week) into bins of one hour and count for every hour the number of client sessions that access the two servers. This way we derive two vectors of equal length that describe the profile of the servers. To measure statistical dependence between the profiles, we compute the Spearman's rank correlation coefficient [24] of the two vectors. The Spearman's coefficient has two useful properties. First, it is a rank correlation measure and therefore it effectively normalizes the two vectors. Second, it captures (non)-linear correlations between the two profiles, i.e., if the activity intensity of two profiles increases/decreases together. If two profiles are strongly correlated then the coefficient tends to 1. The resulting value is used as input to our classifier (Section 4).

To demonstrate the efficiency of this feature, Figure 2 plots the distribution of Spearman's correlation coefficient when comparing the activity profiles of all `top500` HTTPS services with the profile of the closest mail server. For more than 90% of the webmail servers the correlation coefficient is higher than 0.6 while this percentage is only about 50% for non-mail servers.

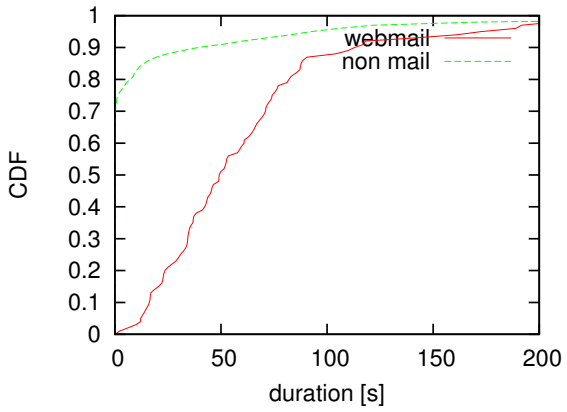

Figure 3: Median of session duration.

3.2.2 Session duration

A second characteristic of user behavior is how long clients normally use a service. We speculate that webmail users generally take some time to read and answer their e-mails and may even keep the browser window open to wait for incoming messages. Hence, access duration for a webmail service should be higher, on average, than the one for a normal webpage. To capture the access duration of a service we consider sessions as described in Section 2. By defining session duration as the time between the start of the first session flow and the start of the last session flow, we ignore potential TCP timeouts. Since we are interested in *typical* user behavior, we use the median across all session duration samples for a given server.

Figure 3 displays the distribution of the median duration for our `top500` servers. Again, we find significant differences between the client behavior of a webmail server and other non-mail servers. While the median session duration of a webmail service is shorter than 25s for only some 20% of the webmail servers, almost 90% of non-mail servers experience such short median access times. Although there exist some non-mail servers with long session durations (these are mainly VPN servers), the overall differences in access duration are sufficiently pronounced to use this as another feature for classifying webmail traffic.

3.3 Periodicity

The activity profiles of Section 3.2.1 capture fluctuations in client activity over long time periods such as one week. We now investigate the existence of higher frequency time patterns. The advent of AJAX-based technologies, which heavily rely on asynchronous interactions between clients and servers, has led to an increase in exchanged messages. Many of today's webmail solutions check periodically (e.g., every 5 minutes) for incoming messages and update the browser window if necessary. This is in line with the experience provided by mail programs such as Outlook or Thunderbird. Our idea is to leverage such periodicity that we expect to be visible in our flow-based data in order to classify e-mail related web traffic.

Using `Wireshark` and `Firebug` we analyze the communication triggered by different webmail implementations. We find evidence of distinct periodicity in the sense that even during idle times a synchronization message is sent at regular time intervals. This also results in a new observed flow between client and server at the same time intervals. Nevertheless, capturing this periodicity requires some signal processing.

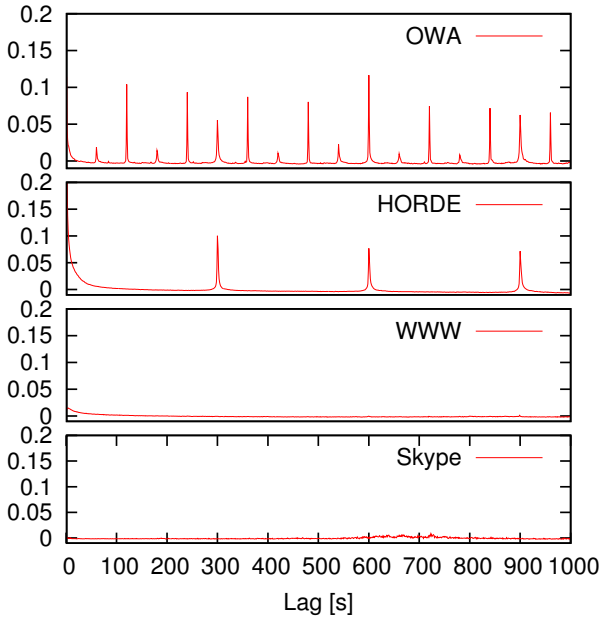

Figure 4: Autocorrelation of flow inter-arrival times.

We observe that webmail sessions are composed of noisy user-invoked traffic, resulting from activities such as sending e-mails or browsing folders, and evident machine-generated periodic traffic. To recover the periodicity buried in the signal, we start by filtering out all short sessions, since they are not useful for identifying periodic behaviors that repeat infrequently, e.g., every 5 minutes. For a server under study, we keep all sessions and respective flows with a duration higher than $1,800$s, which corresponds to six 5-minute intervals. We then split each session into time bins of 1s and count the number of flows that start within a bin. This is repeated for every client of the server under study. Then, we compute the autocorrelation function and average the autocorrelation signals over the entire client base of a particular server, in order to smooth the signal and obtain a server-level behavior.

The plots in Figure 4 display the results for four different types of servers from our `top500` class. Evidently, webmail applications show pronounced peaks at regular intervals, i.e., OWA at 60s and Horde at 300s, while web and Skype traffic does not show this pattern. Note that we observe similar periodic patterns for other webmail applications such as GMail or Yahoo!.

Although it is trivial to assess periodicity visually based on autocorrelation signals, we still need to introduce a simple metric that quantifies periodicity and can be used by a classifier. To this end, we consider the energies of different frequencies in the autocorrelation function. By cutting off small time lags (e.g., below 200s) and high time lags (e.g., above 400s), we mitigate the impact of noise. Then, we compare the maximum and median signal levels that we observe within this time interval. High ratios are a strong indicator for periodicity, i.e., for the existence of a webmail server.

We believe that the basic idea behind our approach could in general be used to distinguish between user- and machine-initiated traffic. Hence, a spectrum of other applications as well is likely to have a characteristic fingerprint related to flow timing (in fact, our preliminary results already hint to such a direction).

4. EVALUATION

The results of the preceding Section 3 are promising in the sense that it is apparently possible to discriminate between e-mail related web traffic and other HTTPS traffic leveraging correlations across protocols and time. In this section we explain how to build a classifier using our proposed features and present first results. Our main concern is not maximum accuracy. Rather we seek to demonstrate the general feasibility and the potential of our approach in a proof of concept study.

The general approach is as follows: We use the Support Vector Machine (SVM) library provided by [4]. For classification we rely on the four features presented in the preceding Section 3, namely service proximity, activity profiles, session duration, and periodicity. These features are used to discriminate between the two classes mail and non-mail. To estimate classification accuracy, we rely on 5-fold cross-validation: we partition the set of hosts into five complementary subsets of equal size. Four subsets are used as training data, the remaining subset serves as validation data. We repeat this process five times such that each of the five subsets is used exactly once as validation data.

Based on the `top500` data set, see Table 1, we classify webmail hosts that are inside the SWITCH network.

	Number of true HTTPS	
	mail servers	non mail servers
classified as mail	61	16
classified as non mail	16	382
accuracy (mean)	93.2% ($\pm$3)	

Table 2: Classification of internal HTTPS servers.

As shown in Table 2, our classification realizes a remarkably high mean accuracy of 93.2% (with standard deviation 3.0) across the five runs of our 5-fold cross-validation. In addition, the precision for the mail class is also reasonably good at 79.2% relying solely on flow data and can be further improved by optimizing the classifier. By manually scrutinizing the results of the classification, we find our classifier can effectively discriminate between webmail and Skype nodes mainly due to the service proximity feature of Section 3.1 and also between webmail and authentication services mainly due to the session duration feature of Section 3.2.2. However, it is challenging to distinguish between VPN and webmail servers, which is the main reason for false negatives when classifying webmail hosts.

Popular webmail platforms such as GMail, Yahoo!, or Hotmail do not maintain servers within the SWITCH network and, therefore, have been ignored in our discussion and analysis so far. Finally, we briefly illustrate that applying our approach to uncover arbitrary webmail applications is equally feasible, irrespective of whether hosts are internal or external.

To this end, we extract additional data from our trace collected in March 2010. Following the methodology outlined in Section 2, we select 500 popular *external* HTTPS hosts/sockets. Manually labeling this data set reveals 32 GMail servers[2]. For 452 additional HTTPS servers we established that they provide other non-mail services such as online banking, update, Facebook login servers, etc.

We now merge the original `top500` data set and the labeled external hosts creating a new data set with 1000 HTTPS servers. Based on our proposed features we build a classifier and again apply 5-fold cross-validation.

[2]Recall that GMail recently switched to HTTPS.

	Number of true HTTPS	
	mail servers	non mail servers
classified as mail	94	30
classified as non mail	19	820
accuracy (mean)	94.8% ($\pm$3)	

Table 3: Classification of internal and external HTTPS servers.

Table 3 shows that we are able to classify internal and external HTTPS servers with overall accuracy of $94.8\% \pm 3$ and precision of 75.8% for the mail class. Although these results are preliminary, it appears possible to detect mail servers outside the SWITCH network using the same features, even though only a small sample of the total traffic of these servers is visible in our network. Moreover, our case study on GMail suggests that the techniques proposed in this paper are not limited to specific applications (e.g., Outlook Web Access, Horde).

5. RELATED WORK

Though extensive research has focused on traffic classification[3], only a comparably tiny portion has studied methods for dissecting WWW traffic. Schneider *et al*. [29] extracted traffic to four popular Ajax-based sites from full packet traces and characterized differences between Ajax and HTTP traffic. Li *et al*. used manually constructed payload signatures to classify HTTP traffic into 14 applications and discussed trends from the early analysis [18] of two datasets collected three years apart. Compared to previous studies, our work focuses on extracting mail traffic from WWW traces and operates on *flows*, solving a substantially more challenging problem than previous HTTP classification studies, which used full packet traces (e.g., [8]).

Some research has recently focused on the traffic characteristics of mail. Notably, Ramachandran *et al*. [26] characterized a number of network properties of spammers and emphasized that spammers can alter the content of spam to evade filters, whereas they cannot easily change their network-level footprint. This work suggested the plausibility of network-based spam filtering and was followed by the SNARE system [10], which uses a small number of simple traffic features to block spammers with reasonable accuracy.

Finally, compared to host-based classification, the novelty of our method can be mostly traced to the addition of the following two "dimensions": (i) we introduce a timing dimension, as this contains valuable information about user- and machine- behaviors; (ii) while earlier host-based approaches [14] attempt to identify a *common* communication pattern (graph) shared by all hosts of this application, we try to correlate the unknown host behavior to *known* hosts running a different, but *related* application; as a result, hosts with different patterns could be validly identified under one class.

6. CONCLUSION

In this paper, we have presented a number of flow-level techniques that can be used to separate webmail traffic from other HTTPS traffic. The novelty of our approach goes into two main directions: (i) we leverage correlations across (related) protocols (e.g. IMAP, POP, SMTP, and webmail) and among hosts sharing a similar client base, and (ii) we identify and exploit timing characteristics of webmail applications. Based on these, we have introduced novel features, investigated their efficiency on a large flow data set, and used them to produce preliminary classification results on internal and

[3]For a survey of traffic classification literature refer to [16].

external HTTPS servers. This is the first work to show that it is possible to uncover HTTPS webmail applications solely based on flow-level data with approximately 93.2% accuracy and 79.2% precision.

While the main focus of this paper has been the presented novel features, in future work, we intend to optimize our classifier and if needed to further extend our set of features (along the two directions discussed) to improve the precision of our classification. In the same direction, we would also like to test the effect of sampling on our techniques, as we believe that most of the features presented are likely immune to sampling. Furthermore, we believe that our features are of broader interest for traffic classification, and could be used, for example, to facilitate demultiplexing of HTTP(S) traffic in general. Finally, having a reliable method to extract webmail traffic, we intend to perform an extensive characterization of the long-term evolution of mail-related traffic including usage trends and traffic mix between classical delivery and web-mail protocols.

Acknowledgments

We are grateful to SWITCH for providing their traffic traces and to the anonymous reviewers for their helpful comments. Also, we want to thank Bernhard Tellenbach, Martin Burkhart and Brian Trammell for their contributions. Special thanks to Simon Leinen for assisting with flow collection and processing. Finally, we are grateful to our shepherd Mark Allman. Part of this work has been funded by the EC Marie Curie IRG project 46528.

7. REFERENCES

[1] T. Auld, A. Moore, and S. Gull. Bayesian Neural Networks for Internet Traffic Classification. *IEEE Transactions on Neural Networks*, 2007.

[2] L. Bernaille, R. Teixeira, and K. Salamatian. Early application identification. In *Proc. of ACM CoNEXT*, 2006.

[3] Trojan Now Uses Hotmail, Gmail as Spam Hosts. `http://news.bitdefender.com/NW544-en-n-\-Trojan-Now-Uses-Hotmail-as-Spam-Hosts.html`, 2007.

[4] C. Chang and C. Lin. *LIBSVM: a Library for Support Vector Machines*, 2001. `www.csie.ntu.edu.tw/~cjlin/libsvm`.

[5] T. Choi, C. Kim, S. Yoon, J. Park, B. Lee, H. Kim, and H. Chung. Content-Aware Internet Application Traffic Measurement and Analysis. In *Proc. of IEEE/IFIP NOMS*, 2005.

[6] Email client popularity. `http://www.campaignmonitor.com/stats/email-clients`.

[7] M. Crotti, M. Dusi, F. Gringoli, and L. Salgarelli. Traffic Classification Through Simple Statistical Fingerprinting. *ACM SIGCOMM CCR*, 2007.

[8] H. Dreger, A. Feldmann, M. Mai, V. Paxson, and R. Sommer. Dynamic Application-Layer Protocol Analysis for Network Intrusion Detection. In *Proc. of USENIX Security Symposium*, 2006.

[9] P. Haffner, S. Sen, O. Spatscheck, and D. Wang. Acas: Automated Construction of Application Signatures. In *Proc. of SIGCOMM MineNet Workshop*, 2005.

[10] S. Hao, N. Feamster, A. Gray, N. Syed, and S. Krasser. Detecting Spammers with SNARE: Spatio-Temporal Network-level Automatic Reputation Engine. In *Proc. of USENIX Security Symposium*, 2009.

[11] The Horde Project. `www.horde.org`.

[12] Scammers Exploit Public Lists of Hijacked Hotmail Passwords. `http://www.computerworld.com/s/article/9139092/Scammers_exploit_public_lists_of_hijacked_Hotmail_passwords`, 2007.

[13] M. Iliofotou, P. Pappu, M. Faloutsos, M. Mitzenmacher, and G. Varghese. Network Monitoring Using Traffic Dispersion Graphs. In *Proc. of ACM IMC*, 2007.

[14] T. Karagiannis, K. Papagiannaki, and M. Faloutsos. BLINC: Multilevel Traffic Classification in the Dark. In *Proc. of ACM SIGCOMM*, 2005.

[15] T. Karagiannis, K. Papagiannaki, N. Taft, and M. Faloutsos. Profiling the End Host. In *Proc. of PAM*, 2007.

[16] H. Kim, K. Claffy, M. Fomenkov, D. Barman, M. Faloutsos, and K. Lee. Internet Traffic Classification Demystified: Myths, Caveats, and the Best Practices. In *Proc. of ACM CoNEXT*, 2008.

[17] H. Kim, M. Fomenkov, K. Claffy, N. Brownlee, D. Barman, and M. Faloutsos. Comparison of Internet Traffic Classification Tools. In *Proc. of IMRG workshop on application classification and identification report*, 2009.

[18] W. Li, A. W. Moore, and M. Canini. Classifying HTTP Traffic in the New Age. In *ACM SIGCOMM, Poster Session*, 2008.

[19] Z. Li, R. Yuan, and X. Guan. Accurate Classification of the Internet Traffic Based on the SVM Method. In *Proc. of IEEE International Conference on Communications (ICC)*, 2007.

[20] A. McGregor, M. Hall, P. Lorier, and J. Brunskill. Flow Clustering Using Maching Learning Techniques. In *Proc. of PAM*, 2004.

[21] The Middleware Threats. `http://biz.yahoo.com/msft/p7.html`.

[22] A. Moore and D. Zuev. Internet Traffic Classification Using Bayesian Analysis Techniques. In *Proc. of ACM SIGMETRICS*, 2005.

[23] A. W. Moore and P. Konstantina. Toward the Accurate Identification of Network Applications. In *Proc. of PAM*, 2005.

[24] J. Myers and A. Well. *Research Design and Statistical Analysis (2nd edition)*. Routledge, 2002.

[25] Microsoft Outlook Web Access. `www.microsoft.com/exchange/code/OWA/index.html`.

[26] A. Ramachandran and N. Feamster. Understanding the Network-Level Behavior of Spammers. In *Proc. of ACM SIGCOMM*, 2006.

[27] M. Roughan, S. Sen, O. Spatschek, and N. Duffield. Class-of-Service Mapping for QoS: a Statistical Signature-Based Approach to IP Traffic Classification. In *Proc. of ACM IMC*, 2004.

[28] D. Schatzmann, M. Burkhart, and T. Spyropoulos. Inferring Spammers in the Network Core. In *Proc. of PAM*, 2009.

[29] F. Schneider, S. Agarwal, T. Alpcan, and A. Feldmann. The New Web: Characterizing AJAX Traffic. In *Proc. of PAM*, 2008.

[30] S. Sen, O. Spatscheck, and D. Wang. Accurate, Scalable In-Network Identification of P2P Traffic Using Application Signatures. In *Proc. of WWW*, 2004.

[31] The Swiss Education and Research Network (SWITCH). `http://www.switch.ch`.