5E-01

A Study of Malicious HTTP-based Auto-ware Identification Using Traffic Features

Manh Cong Tran[†] Yasuhiro Nakamura[‡]

[†‡]Department of Computer Science, National Defense Academy

[†]manhtc@gmail.com; [‡]yas@nda.ac.jp

I. Introduction

General users tend to select the Internet Protocol based on the advantage of convenience, such as SMTP and HTTP which can aggregate various information. Similar with that trend, for the convention in communication between malware and C&C server, the botnet is also being exchanged from IRC server to HTTP server. However, HTTP-based transmission is not currently just used in read or viewing webpage content, but also expended in utilizing for software update, file/movie transmission, or advertising. Therefore, for the tendency of using HTTP in communication of malware is expands, the distinction and identification of HTTP-based malware are becoming more difficult. Furthermore, the behavior in requests from HTTP-based adware, etc...is similarity in comparison with other malware using C&C server, so the measurement of malware is more difficulty. Traditional signature-based method get much achievement, however it has to face with the fast growing of new malware generation. Therefore, anti-virus companies have a hard time keeping their signature database up to date, and their anti-virus scanners often suffer from a high rate of false negatives [1].

In this study, software which automatically communicate to specific server through HTTP is defined as HTTP-based auto-ware, and based on the analysis of HTTP traffic, a method for malicious HTTP-based auto-ware is proposed and estimated.

II. Related Work

A noticeable number of studies about malware detection have adopted at host-level and network-level.

At host-level, based on the time interval values of HTTP GET requests, Ashley has suggested a method for detecting potential HTTP C&C activity based on repeated HTTP connections to a C&C website [2]. However, the approach is just using the evaluation for periodic access. For that reason, the results are noticed with a caution of accuracy.

At network-level, [3] has proposed a method in behavioral HTTP malware clustering based on HTTP traffic traces generated by different malware samples. With this approach, the positive results are archived, however with the fast growing of HTTP malware, it challenges with increase of false alarm.

By observing and analyzing common HTTP traffic features of auto-ware, in this study, a new technique is proposed in identification malicious HTTP-based auto-ware at network-level. Experimental result show positive results which might be still effective with new type of HTTP-based malware.

III. Features Observation and Extraction

For keeping the update from and posting client's data to servers, HTTP-based auto-ware(toolbar, adware, or botnet) periodically generate legal requests to their servers. However, there is still difference in some characteristics:

- Malicious HTTP-based auto-ware will query URLs structured in a similar way, and in a similar sequence [3].
- HTTP-based malicious software, such as botnet, which follow the PULL style where they periodically steadily connect to their server (i.e. command and control server) by requests with an interval in order to get the commands and updates [4].
- Normal auto-ware(e.g. updater and downloader) transmits a similar periodic pattern of traffic that has been generated within a short period of time. A suspicious software does not generate bulk data transfer [5].

According to those characteristics, for each pair client c_i and server S_j a set $\{c_i, S_j, ul_{ij}, pa_{ij}, ds_{ij}\}$ is established to represent communication from client c_i to server S_j. In that, ul_{ij} is average length of URLs (without parameters); pa_{ij} is average number of parameters in requests; ds_{ij} is average amount of data sent by requests.

Some malicious servers are observed as following steps:

- First, all HTTP requests are collected.
- Then, three above features from each client to each malicious server are extracted. For each observed server S_j has a set of $\{C, S_j, UL, PA, DS\}$. A centroid point $Cen_j(aul, apa, ads)$ is defined from that set, in that, each value (aul, apa, ads) are average values of UL, PA, DS respectively.
- At last, Euclidean distance is used to measure the

distance between Cen_j to each $\{c_i, S_j, ul_{ij}, pa_{ij}, ds_{ij}\}$. A graph of order in ascending of the sequence distance is generated which called *Access Variation Graph.*

An illustration of *Access Variation Graph* in one day data of a malicious server is shown in Fig. 1. By observing in many following days, *Access Variation Graphs* are marked as similar each day.

IV. Proposed Model

Proposed model in detection of malicious HTTP-based auto-ware is illustrated in fig.2. In this figure, the first five steps are executed in our experimentation.

The first three steps are implemented as in section III. For the fourth step, Modified Hausdorff Distance [6] is used to measure the similarity between *Access Variation Graphs* of each data set. With the collected data, for each server after step 4, a set of similarity values between graphs are established. The proposed *Suspicious Score(SS)* of a server is standard deviation of these similarity values.

In this study, a threshold is calculated in experimented for listing of suspicious servers. The sixth and the seventh steps are still not experimented this time, however periodical access and HTTP traffic features are considered in future research to detect malicious HTTP malware at host-level.

V. Experimental Results

For experimental purpose, all requests of an organization network are captured from the 10th to 18th of December 2013. It counts about 77,567,000 requests from about 2000 clients. In that, by manually checking in two blacklists of CRDF [7] and Virus Total [8], it records 26 malicious domains in 1459 domains which are used in experimentation. *Suspicious Scores* of domains are calculated, with the threshold equal to 80, the results are summarized in Table 1. The result in Table 1 shows that 92.31% malicious domains having SS < threshold.

VI. Conclusions and Future Work

A study of using HTTP traffic features in malicious HTTP-based auto-ware is presented with positive results in experimentation.

- Almost malicious domains (92.31%) have suspicious score are lower than threshold.

- Detection of malicious domains becomes difficult in the case that there are not enough requests from clients, or malware is not active enough.

- A list of suspicious (with normal and malicious) domains is filtered out by using *Suspicious Scores* which are lower than threshold, the short-list can be used as input for others IDS to reduce processing cost.

The continuous improvement of method to archive better results in malicious HTTP-based auto-ware and also malicious URLs detection is future work.

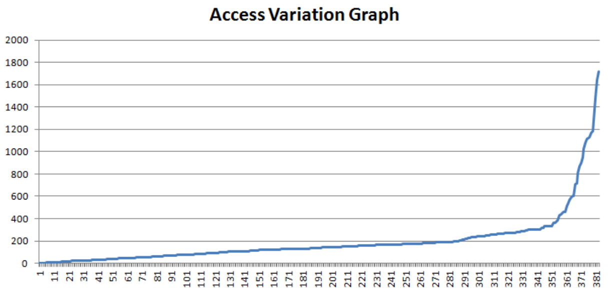

Fig. 1. An illustration *Access Variation Graph* of one day data of requests to a malicious server S_j. X asis is number of clients, and Y asis shows distance between centroid to $\{c_i, S_j, ul_{ij}, pa_{ij}, ds_{ij}\}$.

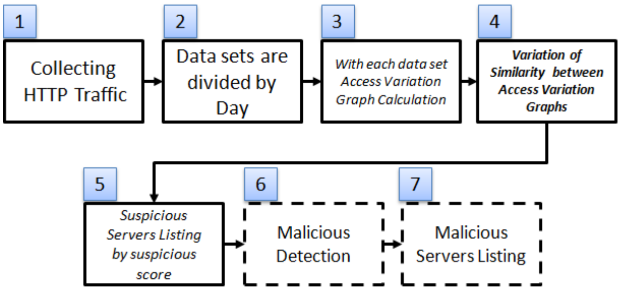
1
Collecting HTTP Traffic
2
Data sets are divided by Day
3
With each data set Access Variation Graph Calculation
4
Variation of Similarity between Access Variation Graphs
5
Suspicious Servers Listing by suspicious score
6
Malicious Detection
7
Malicious Servers Listing

Fig. 2. Description of proposed method.

Table 1: Experimental results

Items	Statistic	Rate (%)
1. Total Domains	1486	
Malicious domains (based on [7], [8])	26	1.75
Adware, Web analytics and others normal domains	1460	98.25
Domains which SS < 80 (threshold)	671	45.15
Domains which SS ≥ 80 (threshold)	815	54.85
2. Malicious domains SS < threshold	24	92.31
3. Malicious domains SS ≥ threshold ***There are not enough requests from clients to judge**	2	7.69

References

[1] D. Ashley, "An algorithm for http bot detection," *University of Texas at Austin - Information Security Office*, 2011.

[2] J. Oberheide et al., "CloudAV: N-Version antivirus in the network cloud," in *Proc. USENIX Security*, 2008.

[3] R. Perdisci et al., "Behavioral Clustering of HTTP-Based Malware and Signature Generation Using Malicious Network Traces," in *Proc. the 7th USENIX conference on Networked systems design and implementation*, pp. 26-39, 2010.

[4] M. Eslahi et al., "An Efficient False Alarm Reduction Approach in HTTP-based Botnet Detection," in *Proc. IEEE Symposium on Computers & Informatics*, pp. 201 - 205, 2013.

[5] W.T. Strayer et al., "Detecting Botnets with Tight Command and Control," in *Proc. the 31st IEEE Conference on Local Computer Networks*, pp. 195-202, 2006.

[6] M.P. Dubuisson et al., "A Modified Hausdorff Distance for Object Matching," in *Proc. International Conference on Pattern Recognition*, pp. 566-568, 1994.

[7] CRDF Threat France Center: List of new threats detected [online] http://threatcenter.crdf.fr/

[8] Virus Total [online] https://www.virustotal.com/