LigeroAV: A Light-Weight, Signature-Based Antivirus for Mobile Environment

Jaehwan LEE†, Min Jae JO††, Nonmembers, and Ji Sun SHIN†(a), Member

SUMMARY Current signature-based antivirus solutions have three limitations such as the large volume of signature database, privacy preservation, and computation overheads of signature matching. In this paper, we propose LigeroAV, a light-weight, performance-enhanced antivirus, suitable for pervasive environments such as mobile phones. LigeroAV focuses on detecting MD5 signatures which are more than 90% of signatures. LigeroAV offloads matching computation in the cloud server with up-to-date signature database while preserving privacy level using the Bloom filter.

key words: signature-based antiviruses, Bloom filters, cloud

1. Introduction

The report from McAfee says that the number of malwares samples exceeds 300 million in the third quarter of 2014 and that is an increase of 76 percent compared to the year before [1]. Such exponential increment of malwares introduced the large size of signature database in signature-based antivirus solutions. As the database size increases exponentially, the computation overheads and memory requirements also increase dramatically, so the overall system performance can be degraded significantly. In addition, recent malwares mainly target smart phones, which has limited computation and energy resources compared to desktop environments. Considering these trends, we need more lightweight, performance improved antiviruses for pervasive environment.

2. Related Works

Signature-based antivirus solutions can be categorized in three ways: the cloud [5], the prematch [3], [4], and the hybrid [6]. CloudAV [5] is the typical solution in the cloud - it sends a target file that a client wants to ask an inspection to the server and the inspection result is sent back to the client. On the other hand, the prematch-based solution has a rough detection procedure to find suspicious files in advance and then proceeds an actual signature-based detection only for those suspicious files. Those solutions have privacy issues because the target file should be sent to the cloud server. SplitScreen [6] is a hybrid solution combining the prematch and the cloud ways. In SplitScreen, client receives the Bloom filter that parts of signatures are hashed, prematches to find suspicious patterns, and sends them to the server. The server matches the suspicious patterns against all the signatures and sends actual signatures of the suspicious patterns to the client so that client can identify the file whether is infected or not by actual pattern matching.

SplitScreen keeps client from saving all the signatures. This lightens client environment and enhances the performance because uninfected files were already filtered out by the prematch. In addition, SplitScreen sends only a part of the file instead of the whole file to the server and unlike CloudAV, this loosens user privacy violations. However, SplitScreen is optimized to inspect regular expression match rather than MD5 signature match, which differs from ours.

3. Our Contribution

In this paper, we propose LigeroAV to overcome such performance degradation introduced by large signature database. In particular, LigeroAV focuses on MD5 signature antivirus (which is more than 90% of entire signatures). We implement LigeroAV and provide its performance measurement and analysis. The advantage of LigeroAV is described in detail as follows.

• LigeroAV is light-weighted; it does not require lots of resources such as memory, computation, and network bandwidth. By leveraging cloud technology, it can offload computation (signature matching) to the cloud server so that the target machine to inspect can save CPU clocks and memory, and it does not need to get large size of signature database from the server. Thus, LigeroAV performs well in the pervasive, mobile environments.

• LigeroAV shows shorter inspection time by reducing the round-trip time in the network compared to SplitScreen. In addition, LigeroAV consumes less network bandwidth, which is better for mobile environments.

• LigeroAV preserves privacy. It does not send all information about inspection target files, so it does not reveal any privacy-related information to the server.

• LigeroAV provides the most up-to-date inspection results compared to other solutions. The initial screening step in LigeroAV runs on the cloud server, which always keeps up-to-date database about malware information. Therefore, LigeroAV can detect a new born, but quickly spreading virus.
4. LigeroAV

Inspired by SplitScreen, we propose LigeroAV to improve performance of signature-based antivirus solutions. By the analysis on ClamAV database [2], MD5 signatures account for 92 percent of signatures and 8.2 percent of the total searching time in ClamAV[2]. SplitScreen focuses on performance improvement of the regular expression search time while LigeroAV targets to minimize the signature search performance of MD5.

Figure 1 shows the system architecture of LigeroAV.

First, the client generates a Bloom filter ($\delta$) of files in the target disk (or directory). The bloom filter of multiple files is defined as a bitwise-OR of all Bloom filter outputs of each file. When constructing the Bloom filter, the client builds a linked-list in the memory to store flag bit position information and file IDs so that it can easily search the suspicious file signature and suspect signature at the later step. Second, after building the Bloom filter, the client sends $\delta$, the Bloom filter, to the server. The server carries out prematch: given the Bloom filter $\delta$ from the client, server generates bloom filters for each signature from database $\Sigma$ and finds a match to $\delta$. If the server finds a prematch, the server sends the signature ($\Sigma'$) corresponding to the Bloom filter ($\delta$). Then, the client proceeds the actual signature match using the signature given by the server and linked-list information in the local memory. Once a bloom filter of files has been delivered to the server, the server can do prematch whenever signature update occurs. It significantly reduces the client’s maintenance burden: once the client uploads the bloom filter of files, the client does not need manually inspects the file anymore unless the server notifies.

Furthermore, our solution can be combined with Splitscreen to provide the hybrid method for both regular expression signatures and MD5 signatures. This composition contributes to enhancing the performance of the signature-based antivirus.

5. Experiments

We implemented LigeroAV in each a server and a client using two virtual machines, each of which has Ubuntu 14.0.4.2 desktop 64bit OS, 1024 MB Memory, and 1 Core CPU (Intel i3-3220, 3.3GHz). Even though the experiment is done on virtual machines, current virtual machine technology is advanced enough to show real performances in a comparative way. We used ClamAV to compare performance and configured it with 0.98.7 version of the engine and 3,854,206 the number of signatures.

Setting of Bloom Filters. The settings of the Bloom filter used by LigeroAV are as the following. The size of the Bloom filter is 3,144 Byte, which is able to store 21,000 data in the Bloom filter. The number of used hash functions is two. If $k$ hash functions and $m$ bits array of the Bloom filter are used, and $n$ elements are inserted into the Bloom filter, the probability of a false positive of the Bloom filter is as follows:

$$P_{falsepositive} = \left(1 - e^{-kn/m}\right)^k$$

(1)

According to the Eq. (1), the probability of a false positive of the Bloom filter in our experiment is 0.03 when two thousand elements are inserted.

5.1 Experimental Results

In this section, we show our experimental results of temporal overhead and memory overhead.

Temporal Overhead. The temporal overhead of the client is the sum of the followings: (1) generation time of the Bloom filter, (2) transmission time of the Bloom filter, and (3) actual inspection time of files.

First, Table 1 shows the generation time of the Bloom filter according to the number of the file. The time to insert a file in the Bloom filter takes 0.0018ms and 3.7 seconds for a thousand of files.

Second, transmission time of the Bloom filter by a network bandwidth is shown (Table 2). The transmission time is 0.3144ms when the bandwidth is 100Mbps and 0.03144ms when the bandwidth is 1Gbps.

Third, the measurement value of the inspection time to match signatures from the server against files is under 0.001ms. Most of operations are search cost in the memory and bit-comparison among integer type values.

As a result, the overall time to complete total inspections by adding the generation time, the transmission time,
and the inspection time is a minimum of 0.0234ms and a maximum of four seconds (Fig. 2). On the other hand, about 11 seconds is required for ClamAV to inspect a file with 3,854,206 signatures (Table 3).

**Memory Overhead.** ClamAV consumes 6.6MB, whereas LigeroAV consumes only 486KB. ClamAV takes about 13.58 times memory consumption more than LigeroAV.

### 5.2 Discussion

We summarize the pros and cons of LigeroAV and other related solutions in various aspects as follows.

**Signature Update:** As discussed earlier, clients of LigeroAV do not involve signature updates: when a new signature pattern is found, it does not need to be downloaded to the client because only the signatures in need after the prematch are sent to the client. ClamAV, however, needs to download signatures that are newly found. In case that a new signature is created frequently, ClamAV clients have to download it every time and this causes an overhead to the client.

**Memory Overhead:** LigeroAV does not use lots of memory compared to ClamAV since it does not need to process all signature information in the database. The only major memory consumption in LigeroAV is linked-list to store bit-position information in the Bloom filter and file IDs, which is bound as a few times of the number of inspection target files in the disk (or directory). Suppose that the number of files is normally less than 100,000, the memory size of the linked-list is less than 1MB, which is practically reasonable amount of memory usage not to reduce total execution performance.

**Network Usage:** To leverage the cloud server capability, LigeroAV needs to use network resource. While SplitScreen needs at least three times to send/receive information to complete total inspection, LigeroAV needs only one send and one receive operation so that it can reduce round trip time of the total inspection. In addition, whenever the signature database is updated, SplitScreen requires to resend the Bloom filter to the all the clients, but LigeroAV does not need to update information proactively, which is suitable for dynamically changing recent environment to catch up quickly wide-spreading virus infections.

Privacy: LigeroAV does not reveal any privacy related information to the servers like SplitScreen, which is different from other cloud-based solution like CloudAV. LigeroAV preserves the privacy level as SplitScreen, while improving performance.

### 6. Conclusion

We propose LigeroAV that reduces the client workloads of MD5 signature-based antivirus by proceeding the prematch in the server. We implemented and experimented LigeroAV in various conditions from 10 Mbps network speed to 1000 Mbps network speed. According to our evaluation, compared to the ClamAV, both of the overall performance and memory consumption are enhanced by LigeroAV. Furthermore, LigeroAV relying on cloud server does not introduce violations of the privacy of users because the prematch by the sever handles only hash values of files. Even though LigeroAV focuses on pervasive environments, we can apply it to regular PC-based environments without any modifications. LigeroAV mostly focuses on the performance to detect MD5 based signatures. To handle various form of signatures, our future work can include the hybrid architecture combining LigeroAV to match MD5 signature and SplitScreen to match regular expression, and evaluations of the hybrid solution.

### Acknowledgments

J. Lee was supported by 2014 Korea Aerospace University faculty research grant. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (No. 2015R1C1A1A02036524 and NRF-2013R1A1A3009163).

### References

[1] Mcfree Report, http://www.mcafee.com/kr/resources/reports/quarterly-threat-q3-2014.pdf
[2] ClamAV, http://www.clamav.net/index.html
[3] O. Erdogan and P. Cao, "Hash-AV: fast virus signature scanning by cache-resident filters," International Journal of Security and Networks, vol.3, no.1-2, pp.50–59, 2007.
[4] R. Kandhan, N. Teletia, and J.M. Patel, "SigMatch: fast and scalable multi-pattern matching," Proceedings of the VLDB Endowment, vol.3, no.1-2, pp.1173–1184, 2010.
[5] Oberheide, J., Cooke, E. and Jahanian, F., "CloudAV: N-Version Antivirus in the Network Cloud," USENIX Security Symposium, pp.91–106, 2008.
[6] S.K. Cha, I. Moraru, J. Jang, J. Truelove, D. Brunley, and D.G. Andersen, ‘SplitScreen: Enabling efficient, distributed malware detection,” Commun. Netw., vol.13, no.2, pp.187–200, 2011.