Empirical analysis of SSL/TLS weaknesses in real websites: Who cares?

Sanghak Oh[1], Eunsoo Kim[2], and Hyoungshick Kim[1]

[1] Department of Software, Sungkyunkwan University, Korea
[2] Department of Computer Science and Engineering, Sungkyunkwan University, Korea

Abstract. As SSL/TLS has become the de facto standard Internet protocol for secure communication in recent years, its security issues have also been intensively studied. Even though several tools have been introduced to help administrators know which SSL/TLS vulnerabilities exist in their network hosts, it is still unclear whether the best security practices are effectively adopted to fix those vulnerabilities in real-world applications. In this paper, we present the landscape of real websites about SSL/TLS weaknesses through an automatic analysis of the possibilities of six representative SSL/TLS attacks—Heartbleed, POODLE, CCS injection, FREAK, Logjam and DROWN—on popular websites. Surprisingly, our experiments show that 45% and 52.6% of top 500 most popular global and Korean websites are still vulnerable to at least one of those attacks, respectively. We also observed several interesting trends in how websites were vulnerable to those attacks. Our findings suggest that better tools and education programs for SSL/TLS security are needed to help administrators keep their systems up-to-date with security patches.

Keywords: SSL/TLS, vulnerability, attack, security patch;

1 Introduction

Secure Sockets Layer/Transport Layer Security (SSL/TLS) protocol has become the de facto standard for ensuring confidentiality and data integrity of messages that are exchanged between network parties in the Internet [1]. Due to the popularity of the SSL/TLS protocol, the security of SSL/TLS has also been intensively studied for a long time. Moreover, it is commonly believed that flaws could be fixed rapidly by the community if and when they are found. This belief motivated our study to assess the practical security levels of websites that are using the SSL/TLS protocol.

In this paper, we particularly investigate how many popular websites are vulnerable to the six representative SSL/TLS attacks—Heartbleed [2], POODLE [3], CCS injection [4], FREAK [5], Logjam [6] and DROWN [7] attacks—to understand the gap between academic research and real-world practice in using the SSL/TLS protocol.

In order to perform our experiments on a large scale, we developed an integrated tool to automatically check the possibilities of the six SSL/TLS attacks on websites, respectively. With this tool, we tested the 500 most popular global and Korean websites collected in Alexa (`http://www.alexa.com/topsites`) and then found that 45% and 52.6% of those global and Korean websites are still vulnerable to at least one of those attacks, respectively. We also observed several interesting trends in how websites were vulnerable to those attacks. For example, we found there exist some positive correlations between those six attacks. This implies that websites may be exposed to an SSL/TLS vulnerability (e.g., Logjam) with a high chance when they are exposed to another SSL/TLS vulnerability (e.g., FREAK). Also, when we observe the changes in the number of vulnerable websites over the four weeks, the overall adoption rates of patches seem slow although the number of vulnerable websites for each attack was rather decreased. We believe that our experiment results are quite impressive because the six tested SSL/TLS attacks and their defense techniques were already reported many times in media coverage as well as academic papers.

We summarize the key contributions of the paper as follows:

- We performed a large-scale quantitative analysis of the six representative SSL/TLS attacks on popular websites and found that many popular websites (more than about 45%) are still vulnerable to at least one of those attacks.
- We developed an integrated tool to automatically check the possibilities of the six tested SSL/TLS attacks on target websites at the same time. This kind of tools can be used for the purpose of identifying SSL/TLS security flaws early in websites.
- We discussed how to narrow down the gap between academic research and real-world practices for using the SSL/TLS protocol.

The rest of paper is organized as follows. Section 2 briefly explains the SSL/TLS protocol and the six SSL/TLS attacks used in our experiments. Section 3 presents our methodology for experiments. Section 4 presents the key experiment results. Section 5 suggests our recommendations to reduce SSL/TLS vulnerabilities in websites. Related work is covered in Section 6, and our conclusions are in Section 7.

2 Background

In this section, we describe the overall explanation of SSL/TLS protocol and six typical attacks related to SSL/TLS vulnerabilities.

2.1 SSL/TLS protocol

SSL/TLS, developed by Netscape, is the de facto Internet protocol for ensuring confidentiality and data integrity. The security of SSL/TLS has been studied for a long time, and formally verified [8]—its end-to-end security is equivalent to the cryptographic strength of the underlying algorithms if implemented properly. Surely, however, security proofs are no panacea. The implementations for SSL/TLS may often have bugs although flaws are generally fixed by the community when they are found.

2.2 SSL/TLS attacks

2.2.1 Heartbleed attack

Heartbleed vulnerability is a buffer overread due to the bug with the TLS Heartbeat extension which was released in OpenSSL version 1.0.1 on March 13, 2012 [2]. The Heartbeat extension enables each connected peer to determine whether the peer on the other end-point is still present or not by sending a HeartbeatRequest message to verify their connectivity. The request message consists of a type field, payload length field, a payload and random padding. If the payload length is greater than the amount of data in a HeartbeatRequest message, an attacker can read private data from its victim's memory. The latest OpenSSL patch checks whether the received HeartbeatRequest message's payload length field exceeds the length of the payload.

2.2.2 POODLE attack

POODLE attack is a man-in-the-middle attack which uses SSL requests to downgrade TLS version to older protocol versions (e.g., SSL 3.0). Fogel et al. [9] was able to gather data with TLS_FALLBACK_SCSV flag option using the vulnerability in the OpenSSL toolkit. To prevent POODLE attack, Google security team provided a temporary solution that inserts an extra flag (TLS_FALLBACK_SCSV) into SSL/TLS implementations on clients and servers sides. The flag forbids any attempt to downgrade TLS to SSL, but the vulnerability still exists with the clients with SSL v3.0. Disabling the entire SSL v3.0 can also be another alternative solution.

2.2.3 CCS injection attack

CCS injection attack was discovered to be using the vulnerability in OpenSSL library [4]. This attack can be used to exploit the vulnerability to decrypt, extract and modify traffic through a man-in-the-middle attack against an encrypted connection. During the handshake stage, the client or the server could often decide to modify the ciphering strategies of the connection by using a ChangeCipherSpec request. Unlike the standards (RFC 5246 [10]), OpenSSL accepted a ChangeCipherSpec request ChangeCipherSpec request which results in the state between both sides being desynchronized. CCS injection vulnerability was already patched in OpenSSL by fixing the way CCS packets and zero-length pre-master secret values are managed.

2.2.4 FREAK attack

FREAK attack is a technique which uses a man-in-the-middle attack to obatain RSA key by downgrading the key length to 512-bit export-grade length in a TLS connection. Beurdhouch et al. [5] showed that the attack involves factoring RSA_EXPORT keys with only a modest amount of computation, and because many TLS libraries still provide compatibility to handle legacy cipher-suites, it may cause clients to fallback to RSA_EXPORT. In their study, they tested on a number clients and found that most mobile web browsers, such as Android Browser, Safari, Chrome, Blackberry and Opera, were vulnerable to FREAK attack. It was reported that as of March 2015, the vulnerable web browsers were patched by their vendors.

2.2.5 Logjam attack

Logjam attack is a flaw of TLS protocol with Diffie-Hellman cryptography in a TLS connection. Adrian et al. [6] performed a man-in-the-middle attack to recover the session key by attacking connections between the web browsers and any servers that accept export-grade Diffie-Hellman which allows downgrading connection to export-grade. During this attack, a server would select DHE_EXPORT instead of the regular DHE connection for negotiation when there is an export-grade fallback, and issue a signed ServerKeyExchange message. It is important to note that the structure of this message and the message sent during the standard DHE ciphersuite is the same, meaning that there is no indication of which ciphersuite is chosen by the server. Many popular vendors, such as Apple, Google, Microsoft and Mozilla already released patches to fix the problem.

2.2.6 DROWN attack

The DROWN attack uses the vulnerability of SSLv2. The latest clients and servers use TLS protocol to perform cryptographic communication, but a lot of servers still provide compatibility with SSLv2. It is even possible that the servers, which do not allow SSLv2 connection by default, may have their options modified unintentionally by the administrators during the server optimization process. Aviram et al. [7] presented a novel cross-protocol attack called special DROWN, which can decrypt passively collected TLS sessions by using a server which supports SSLv2 as a Bleichenbacher padding oracle. The exploitation process includes a chosen-ciphertest attack which can be used to steal a session key for a TLS handshake. Aviram et al. advised to consider two properties when designing a protocol to prevent DROWN attack; to use longer RSA plaintexts, such as 48 bytes, and let server authenticate the client first to check if it has the knowledge of the RSA plaintext.

3 Methodology

To discover SSL/TLS vulnerabilities on a set of websites in a scalable manner, we implemented a tool based on Nmap 7.1.2 [11] to integrate existing Nmap scripts for the five SSL/TLS vulnerabilities (Heartbleed, POODLE, CCS injection, FREAK and Logjam) on each of target websites by iteratively sending predefined packets to the target website and analyzing the response from that website. For DROWN attack, we used the website [12] that offers the interface to check whether a given host appears to be vulnerable to DROWN attack. We also developed another tool based on Selenium WebDriver [13] to perform automated tests with a large scale of websites. Given a set of websites, our tool checks the existence of the DROWN attack vulnerability on those websites and collects all test results.

The developed tools are available on our GitHub (see `https://github.com/sanghak/Checking_SSL_attacks/`).

4 Experiments

This section presents the experiment results for most popular websites. We checked the possibilities of the six attacks introduced in Section 2.2 on Alexa Top 500 websites, respectively, in both the world and South Korea to analyze not only the trends of SSL/TLS security in global websites but also compare the results with those of regional websites in South Korea to understand inter-country differences in keeping SSL/TLS servers up-to-date to deal with known SSL/TLS vulnerabilities.

We note that the main motivation of our experiments was not to damage real websites. We just intended to conduct a threat and risk analysis on websites to understand the SSL/TLS attack trends and evaluate their severity and likelihood. For secure communication over the Internet, we will publish statistics of the SSL/TLS vulnerabilities found on websites tested and suggest how to fix the discovered vulnerabilities.

For experiments, we used a PC (with a 3.2GHz Intel Core i5 CPU and 8GB RAM) running the Ubuntu 14.04 version, and equipped with a non-congested 100Mbit/s LAN that was connected to the Internet. The experimental results are shown in the following sections. We measured the running time of our implementation to show the relative efficiency of the SSL/TLS scanning methods. Table 1 shows performance measurements of our implementation for each of SSL/TLS attacks tested.

Table 1: Running time of our implementation for scanning SSL/TLS vulnerabilities on websites (μ: mean, σ: standard deviation).

	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
Time (μ)	6.58s	8.18s	32.03s	3.2s	13.17s	18.64s
Time (σ)	8.07s	5.31s	208.45s	5.24s	13.34s	12.92s
Tool	Nmap	Nmap	Nmap	Nmap	Nmap	Web

We measured the running time of our implementation to evaluate the performance of the SSL/TLS vulnerability scanners with a real dataset. Overall, except for Logjam, all scanning methods were efficiently implemented. For Logjam attack, the standard deviation is very large in comparison with other attacks, which means that the testing time for Logjam attack varied greatly depending on websites.

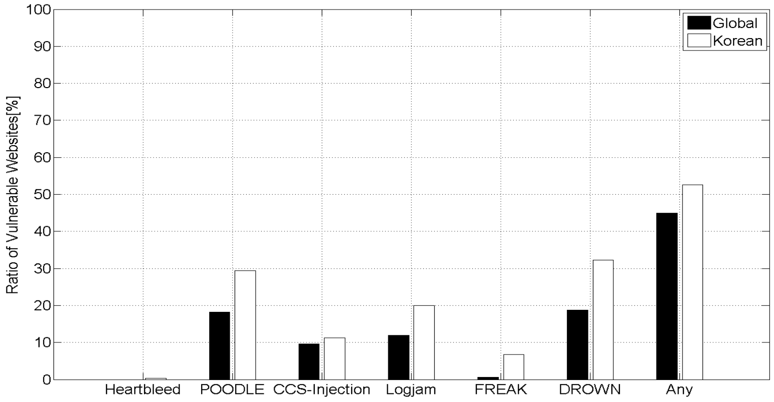

Fig. 1: Ratio of vulnerable websites for each of SSL/TLS attacks.

4.1 Ratio of vulnerable websites for SSL/TLS attacks

We first analyzed how many websites are still vulnerable to each of the six SSL/TLS attacks tested. In Figure 1, we show the ratio of websites vulnerable to each SSL/TLS attack. A significant portion of popular websites is vulnerable to at least one of the SSL/TLS attacks tested in both global websites (about 45%) and Korean websites (52.6%) (see "any" in Figure 1). These results are quite surprising because the tested attacks were already intensively studied. We found that nearly half of all websites tested have supported legacy protocols (e.g., SSL 3.0), weak encryption algorithms (e.g., `EXPORT_DHE` and `EXPORT_RSA`) and/or used an unpatched OpenSSL version without taking proper countermeasures.

Among the vulnerabilities tested, DROWN and POODLE vulnerabilities remain the most popular ones in both global and Korean websites. This is not surprising since DROWN vulnerability is the most recently introduced while POODLE vulnerability cannot be easily fixed—there is the only way to mitigate POODLE vulnerability by disabling SSL 3.0.

For Heartbleed attack, there was no vulnerable website in the global websites, which is similar to the results in a technical report [2]. However, although very few, there were still 2 of 500 vulnerable websites in South Korea. Unfortunately, it appears to be a very critical result since one of those websites provides a real name verification service (with "i-PIN") for Korean users where users' sensitive personal data (including user name, password, RRN [14], phone number) could be improperly managed.

Overall, Korean websites (52.6%) are significantly more vulnerable to SS-L/TLS attacks than global websites (45%) (p = 0.019, Fisher's exact test). This is probably because the latest SSL/TLS attacks might be more popular for global websites compared with Korean ones. Therefore, in South Korea, there is a need to develop a more effective patch management process so as to meet global standards for SSL/TLS protocol implementations.

4.2 Characteristics of SSL/TLS vulnerabilities

We study the distribution of the number of vulnerabilities per website in order to examine how many websites are vulnerable to multiple vulnerabilities. Figure 2 shows the results for global and Korean websites, respectively. We observed a similar trend in both global and Korean websites—most websites are vulnerable to a small number of SSL/TLS vulnerabilities and considerably skewed to the left. Interestingly, in global websites, the websites vulnerable to a single SSL/TLS attack were dominated (100%) by POODLE attack (see Table 2) while a clear majority is DROWN attack in the case of the websites vulnerable to two or three SSL/TLS vulnerabilities. For the Korean websites with a small number of vulnerabilities, DROWN and POODLE attacks more popular than other attacks (see Table 3).

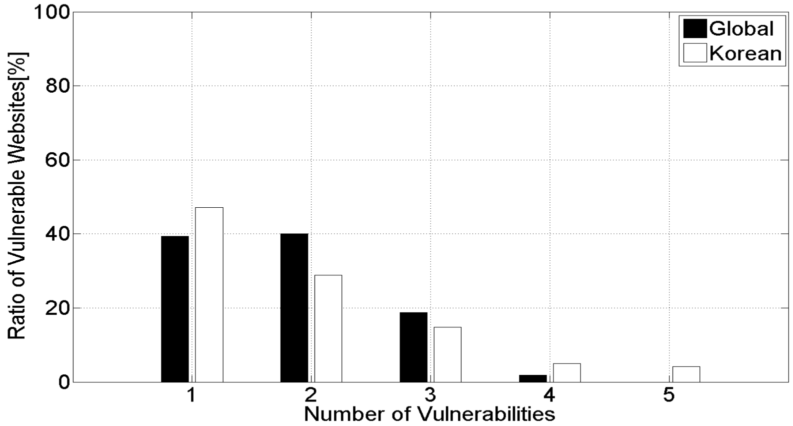

Fig. 2: Number of vulnerabilities per website.

We also measured correlations between the six different types of vulnerabilities tested to examine how those attacks are related to each other. To analyze correlations between those vulnerabilities, Pearson's correlation coefficients were calculated. Table 4 and 5 show the results for global and Korean websites, respectively. For global websites, we found positive correlations between POODLE, Logjam and FREAK. We also found a positive correlation between Logjam and DROWN (see Table 4). For Korean websites, we can see that all SSL/TLS attacks except for Heartbleed are positively correlated. Our findings suggest that the detection of an SSL/TLS vulnerability could be used as an early warning

Table 2: Proportions and numbers of SSL/TLS attacks with the number of vulnerable attacks on global websites. Numbers in parentheses indicate the numbers of vulnerable websites for each attack.

# vulnerabilities	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
1	0% (0)	100% (65)	0% (0)	0% (0)	0% (0)	0% (0)
2	0% (0)	0% (0)	46.96% (31)	53.03% (35)	0% (0)	100% (66)
3	0% (0)	74.19% (23)	83.87% (26)	41.93% (13)	0% (0)	100% (31)
4	0% (0)	100% (3)	100% (3)	0% (0)	100% (3)	100% (3)
5	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)	0% (0)

Table 3: Proportions and numbers of SSL/TLS attacks with the number of vulnerable attacks on Korean websites. Numbers in parentheses indicate the number of vulnerable websites for each attack.

# vulnerabilities	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
1	0% (0)	23.39% (29)	12.90% (16)	9.68% (12)	0% (0)	54.03% (67)
2	1.32% (1)	75% (57)	46.05% (35)	17.11% (13)	2.63% (2)	57.89% (44)
3	0% (0)	94.87% (37)	64.1% (25)	33.33% (13)	30.77% (12)	76.92% (30)
4	0% (0)	100% (13)	100% (13)	53.85% (7)	76.92% (10)	69.23% (9)
5	9.09% (1)	100% (11)	100% (11)	100% (11)	90.9% (10)	100% (11)

indicator of other SSL/TLS security vulnerabilities since they have positive correlations. Therefore, we first scan only a representative SSL/TLS vulnerability (e.g., DROWN attack) before testing all known vulnerabilities. We note that the vulnerability of DROWN attack was always found in the global websites with multiple vulnerabilities (see Table 2). This approach might be helpful to improve the performance of network scanning systems.

We also analyzed the relationship between the ranks of websites and the number of vulnerabilities to examine whether more popular websites are more likely to be secure against SSL/TLS attacks. However, we failed to find any significant correlations with the websites tested.

Table 4: Correlations between SSL/TLS attacks on global websites. Numbers represent correlation coefficients and p-value (numbers in parentheses). Significant values (p-values < 0.05) are marked with bold font.

Attacks	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
Heartbleed						
POODLE	NaN					
Logjam	NaN	**0.1608** **(p < 0.001)**				
CCS injection	NaN	-0.0657 (p = 0.142)	0.0468 (p = 0.296)			
FREAK	NaN	**0.1647** **(p < 0.001)**	**0.2104** **(p < 0.001)**	-0.0253 (p = 0.572)		
DROWN	NaN	-0.0014 (p = 0.974)	**0.1059** **(p = 0.017)**	0.0517 (p = 0.248)	-0.0374 (p = 0.404)	

Table 5: Correlations between SSL/TLS attacks on Korean websites. Numbers represent correlation coefficients and p-value (numbers in parentheses). Significant values (p-values < 0.05) are marked with bold font.

Attacks	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
Heartbleed						
POODLE	0.0287 (p = 0.522)					
Logjam	**0.1267** **(p = 0.004)**	**0.4346** **(p < 0.001)**				
CCS injection	0.078 (p = 0.081)	**0.2441** **(p < 0.001)**	**0.2346** **(p < 0.001)**			
FREAK	-0.0171 (p = 0.702)	**0.4011** **(p < 0.001)**	**0.3813** **(p < 0.001)**	**0.2568** **(p < 0.001)**		
DROWN	0.0241 (p = 0.590)	**0.2787** **(p < 0.001)**	**0.137** **(p = 0.002)**	**0.1489** **(p < 0.001)**	**0.2049** **(p < 0.001)**	

4.3 Changes in the number of vulnerable websites over time

Lastly, we analyzed how the number of vulnerable websites were changed over time. We repeated the scanning procedure with the same websites tested once a week from March 22nd 2016 to April 19th 2016. The results for global and Korean websites are shown, respectively, in Table 6 and 7. We note that the number of vulnerable websites can be rather increased compared to previous results since some websites were often unreachable in our test attempts.

We observed a similar trend in both global and Korean websites—the number of vulnerable websites was slightly decreased in all SSL/TLS vulnerabilities tested except for FREAK in global websites. In particular, in global websites, the websites vulnerable to DROWN (18%) attack were more likely to be patched than others while the numbers of POODLE (12.9%) and Logjam (12%) attacks were the most highly reduced vulnerabilities in Korean websites. Although slight improvements among SSL/TLS vulnerabilities may be noted, the adoption rates of patches seem still slow.

Table 6: Changes in number of vulnerable global websites tested over four weeks.

	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
Week 1	0	91	60	48	3	100
Week 2	0	90	61	41	3	80
Week 3	0	91	61	44	4	80
Week 4	0	85	53	42	4	82

Table 7: Changes in number of vulnerable Korean websites tested over four weeks.

	Heartbleed	POODLE	Logjam	CCS injection	FREAK	DROWN
Week 1	2	147	100	56	34	161
Week 2	2	143	98	57	34	160
Week 3	2	142	97	53	33	148
Week 4	2	128	88	53	32	151

5 Recommendations

In Section 4, we can see that a significant number of websites (i.e., 45% of global websites and 52.6% of Korean websites) are still vulnerable to SSL/TLS attacks even though there already exist several vulnerability scanners such as Zmap [15] and Masscan [16]. This implies that the automation of security vulnerability detection is necessary but not sufficient for less-skilled administrators. Most existing tools simply detect vulnerabilities and do not offer actionable advices to fix the problems (see the results in Figure 3). When vulnerabilities are discovered, less-skilled administrators would lack the security knowledge to understand the dangers of those vulnerabilities and to patch them. Hence, we should develop tools to provide more actionable advices to administrators; if an automated security analysis tool detects a security vulnerability on a target host, it seems better to show its potential risks and step-by-step instructions that enable an administrator to implement the best security practices to fix the problem.

Here, government could have a role in periodically performing such security checks and publishing the checking results for system administrators to encourage them to keep their systems up-to-date with latest security patches.

```
                                    ...
|     State: LIKELY VULNERABLE
|     IDs:  CVE:CVE-2014-3566  OSVDB:113251
|           The SSL protocol 3.0, as used in OpenSSL through
|           1.0.1i and other products, uses nondeterministic
|           CBC padding, which makes it easier for man-in-the
|           -middle attackers to obtain cleartext data via a
|           padding-oracle attack, aka the "POODLE" issue.
|     Disclosure date: 2014-10-14
                                    ...
```

Fig. 3: Example of detection results in Nmap.

6 Related Work

As the SSL/TLS protocol has become the de facto standard for Internet, the security about SSL/TLS has also been intensively studied. CCS injection was first reported on December 2013 [4]. Heartbleed, a notorious vulnerability of SSL/TLS, emerged during April 2014 [2]. POODLE attack, which exploits the vulnerability of supporting SSL 3.0, was reported on October 2014 [3]. During March 2015, FREAK attack caused quite a stir amongst the security administrators as it makes use of the vulnerability of export RSA cipher [5]. Logjam, reminiscent of FREAK attack, was reported to be related to the vulnerability of export DH cipher on October 2015 [6]. Lastly, DROWN attack [7] was reported recently on March 2016.

Benjamin et al. [9] analyzed how many websites were vulnerable to POODLE and FREAK attacks on Alexa's top websites and found that there was a slow patch adoption rate in mitigating those attacks. We extend their work by testing the six representative SSL/TLS attacks—Heartbleed, POODLE, CCS injection, FREAK, Logjam and DROWN—on popular websites. We also compared the analysis results on global websites with those on Korean websites to examine the differences between regional websites and global websites.

7 Conclusion

We analyzed how most popular websites are vulnerable to well-known SSL/TLS attacks to understand the gap between academic research and real-world situations in deploying security practices. We conducted experiments on the top 500 most popular global and Korean websites, respectively. Our results showed that 45% of global websites and 52.6% of Korean websites are still vulnerable to at least one of the six SSL/TLS attacks tested. This is probably because the state-of-the-art of defense methods against SSL/TLS attacks has not been popularly introduced to system administrators (particularly in Korean websites). To bridge the gap between academia and real-world systems, we need better vulnerability scanners and education programs that deal with the SSL/TLS security issues.

Acknowledgements

This work was partly supported by the MSIP/IITP (R0166-15-1041), the ITRC (IITP-2016-R0992-16-1006), and the IITP (No.R-20160222-002755). Authors would like to thank all the anonymous reviewers for their valuable feedback.

References

1. Mittal S Bhiogade. Secure socket layer. In *Proceedings of the Computer Science and Information Technology Education Conference*, 2002.
2. Zakir Durumeric, James Kasten, David Adrian, J Alex Halderman, Michael Bailey, Frank Li, Nicolas Weaver, Johanna Amann, Jethro Beekman, Mathias Payer, et al. The matter of heartbleed. In *Proceedings of the Conference on Internet Measurement Conference*, 2014.
3. Bodo Möller, Thai Duong, and Krzysztof Kotowicz. This POODLE bites: exploiting the SSL 3.0 fallback. *Google, Sep*, 2014.
4. MITRE. CCS-injection CVE Report(CVE-2014-0224). `http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-0224`, 2013.
5. Benjamin Beurdouche, Karthikeyan Bhargavan, Antoine Delignat-Lavaud, Cédric Fournet, Markulf Kohlweiss, Alfredo Pironti, Pierre-Yves Strub, and Jean Karim Zinzindohoue. A messy state of the union: Taming the composite state machines of TLS. In *Proceedings of the IEEE Symposium on Security and Privacy*, 2015.
6. David Adrian, Karthikeyan Bhargavan, Zakir Durumeric, Pierrick Gaudry, Matthew Green, J Alex Halderman, Nadia Heninger, Drew Springall, Emmanuel Thomé, Luke Valenta, et al. Imperfect forward secrecy: How Diffie-Hellman fails in practice. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, 2015.
7. Nimrod Aviram, Sebastian Schinzel, Juraj Somorovsky, Nadia Heninger, Maik Dankel, Jens Steube, Luke Valenta, David Adrian, J Alex Halderman, Viktor Dukhovni, et al. DROWN: Breaking TLS using SSLv2. 2008.
8. Lawrence C. Paulson. Inductive Analysis of the Internet Protocol TLS. *ACM Transactions on Information and System Security*, 2(3):332–351, 1999.
9. Benjamin Fogel, Shane Farmer, Hamza Alkofahi, Anthony Skjellum, and Munawar Hafiz. POODLEs, More POODLEs, FREAK Attacks Too: How Server Administrators Responded to Three Serious Web Vulnerabilities. In *Engineering Secure Software and Systems*, pages 122–137. Springer, 2016.
10. T Dierks and E Rescorla. RFC 5246: The Transport Layer Security (TLS) Protocol Version 1.2, Aug. 2008. *Updated by RFCs*, 5746(5878):6176.
11. Gordon Fyodor Lyon. *Nmap network scanning: The official Nmap project guide to network discovery and security scanning.* Insecure, 2009.
12. Aviram. The DROWN Attack. `https://drownattack.com/`, 2016.
13. Jason Huggins. Selenium WebDriver. `http://docs.seleniumhq.org/projects/webdriver/`, 2016.
14. Youngbae Song, Hyoungshick Kim, and Jun Ho Huh. On the guessability of resident registration numbers in south korea. In *Proceedings of Australasian Conference on Information Security and Privacy*, 2016.
15. Zakir Durumeric, Eric Wustrow, and J Alex Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *Proceedings of the Usenix Security*, 2013.
16. Robert David Graham. MASSCAN: Mass IP port scanner. *URL: https://github.com/robertdavidgraham/masscan*, 2014.