Check for updates

Journal of KIIT. Vol. 18, No. 3, pp. 149-157, Mar. 31, 2020. pISSN 1598-8619, eISSN 2093-7571
http://dx.doi.org/10.14801/jkiit.2020.18.3.149

Potential Vulnerabilities of Automated Points Accumulating System and Countermeasures

Yong-Hyeon Park*, Jae-Young Jeong**, Hong-Jin Kim***, Nam-Hyeong Kim****, Min-Ju Jo*****[1], Kyung-Min Lee*****[2], and Jun-Beom Lee******

Abstract

In the early 2000s, as the Internet got spread widely and a lot of electronic commerce companies appeared, people in the world became used to transactions that happened on the Internet. The scale of electronic commerce had gotten bigger, eventually, the scale of it reached 400 billion dollars in 2016. Those electronic commerce companies issue online property called points which can be used as cash with a certain ratio to reduce commission for transactions on credit and encourage people to use electronic commerce by simplifying steps needed to buy products on the Internet. Automated points accumulating system transplants this convenient feature of electronic commerce into small offline shops and owners of them get interested in this system. Now a lot of stores introduce and use this system. This paper studies the potential vulnerabilities of this system and suggests several instructions preventing damages from these.

요 약

2000년대 초, 인터넷이 확산되고 많은 전자상거래 업체들이 등장함에 따라 사람들의 온라인 거래 의존도가 높아졌다. 전자상거래의 규모는 계속 확대되어 2016년에는 4000억 달러에 이르렀다. 이러한 전자상거래 업체는 신용카드 거래 수수료를 절감하고 구매 절차를 간소화하여 소비자들을 끌어들이기 위한 목적으로 일정 비율로 현금처럼 사용될 수 있는 포인트를 발급하였다. 이렇게 포인트를 자동으로 적립하고 사용할 수 있는 편리한 기능을 오프라인 매장에 적용할 수 있는 포인트 적립 시스템이 개발되었고, 오프라인 매장 점주들은 해당 시스템에 관심을 가졌으며, 현재 많은 매장이 해당 시스템을 사용하고 있다. 본 논문에서는 오프라인 매장에 널리 사용되고 있는 포인트 자동 적립 시스템에서 발견될 수 있는 취약점을 연구하고 이로 인한 피해를 예방하기 위한 몇 가지 방법을 제안한다.

Keywords

electronic commerce, automated points accumulating system, cryptographic protocol, session, brute force attack, password

* KAIST Graduate School of Information Security
- ORCID: http://orcid.org/0000-0002-1093-7193
** Hanyang University Department of Computer Science
- ORCID: http://orcid.org/0000-0002-4855-6716
*** Soongsil University Ph.D of Business Administration
- ORCID: http://orcid.org/0000-0003-4341-9728
**** Soonchunhyang University School of Information Security
- ORCID: http://orcid.org/0000-0003-0532-4656
***** Dongguk University in Gyeongju Department of Computer Science
- ORCID[1]: http://orcid.org/0000-0002-8990-5468
- ORCID[2]: http://orcid.org/0000-0002-4441-6177
****** Catholic University of Daegu Department of Computer Science
- ORCID: http://orcid.org/0000-0003-2474-4525

· Received: Nov. 28, 2019, Revised: Jan. 23, 2020, Accepted: Jan. 27, 2020
· Corresponding Author: Yong-Hyeon Park
Graduate School of Information Security, KAIST
Tel.: +82-31-555-8237, Email: yh8237@kaist.ac.kr

Ⅰ. Introduction

In the early 2000s, electronic commerce became popular and the number of related companies increased significantly[1]-[3]. As the scale of electronic commerce grows, they have delegated their electronic payment tasks to specialized electronic payment service providers. On the other hand, electronic commerce companies issue and provide online properties called points, which can be used like cash. Points help to reduce commission for transactions on credit, advertise their company, and encourage people to use electronic commerce by stimulating to use points and simplifying steps needed to buy products on the Internet.

Furthermore, the remarkable features of electronic commerce are that companies easily construct a database of users' information and the products they bought. Most companies analyze this database to find out suitable marketing strategies and provide some points to customers to attract them to revisit their mall, which leads to better sales.

Many offline shops also issue points or coupons to take advantage of electronic commerce. However, side effects are clearly shown: it is easy to lose coupons and takes a long time to add points manually in front of the cashier. Also, it is difficult to gather customers' information manually. For these reasons, small offline shop owners had been looking for ways to overcome these shortcomings.

At that time, automated points accumulating system which satisfy all needs of offline shop owners is developed, and it is widely used these days. This system helps to attract consumers to be absorbed in accumulating points by removing inconvenient steps, gathering as much information about consumers and products they buy as possible, and finding out proper marketing strategies from the information it gathers.

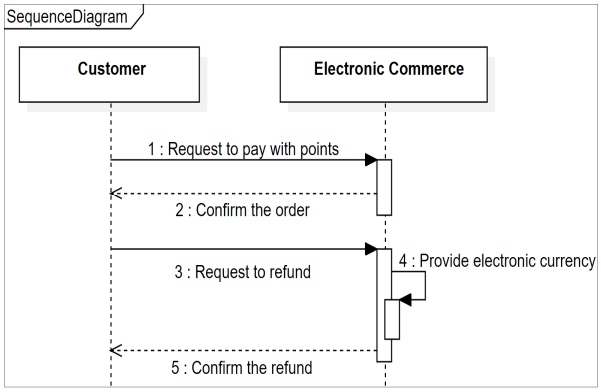


Fig. 2. Process of a purchase and a refund with points

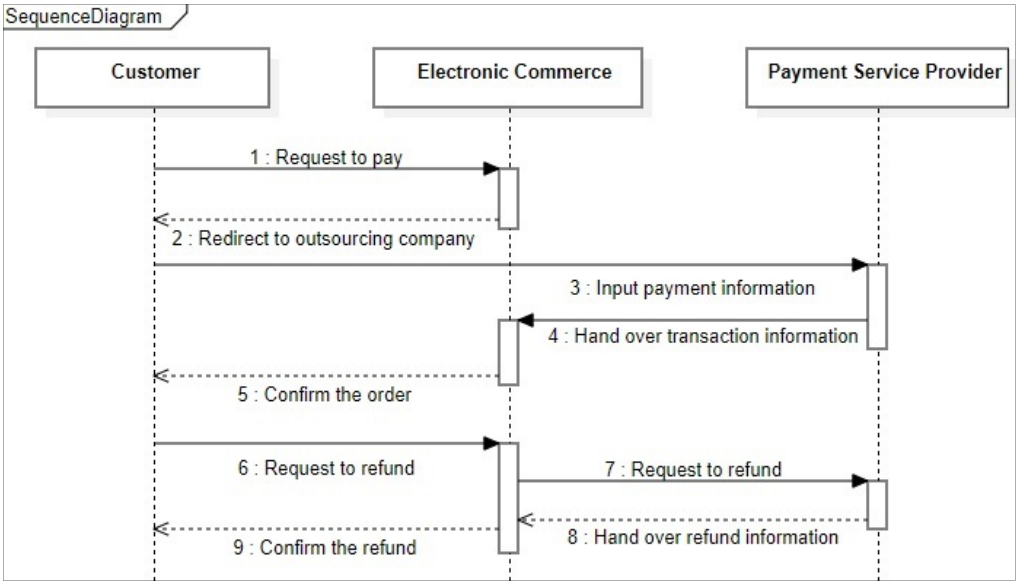


Fig. 1. Process of a purchase and a refund with a cash

We can consider the advantages that electronic commerce takes are transplanted into a small offline shop. In short, this automated points accumulating system which takes almost no time to accumulate points, gathers and manages customers' information automatically, and helps for an owner to get proper promotion strategies to attract customers to visit their shop again satisfies them successfully.

Theoretically, this system sounds good, however, its vulnerabilities might cause severe financial loss. However, the security issues of this system are not regulated properly by current laws even though this system is concerned with online property, which is because current laws only deal with cash itself. To solve the problem mentioned above, we would like to explain this system in detail in chapter 2 and introduce all kinds of expected vulnerabilities and mitigation techniques of this system in chapter 3. We hope these mitigations are applied to further updates.

II. Explanation of Points Accumulating System

As mentioned above, some small offline shop owners are getting interested in automated points accumulating systems operated in additional devices. Devices needed to operate this system are a tablet and a POS. The structure of this system is depicted in Fig. 3 above.

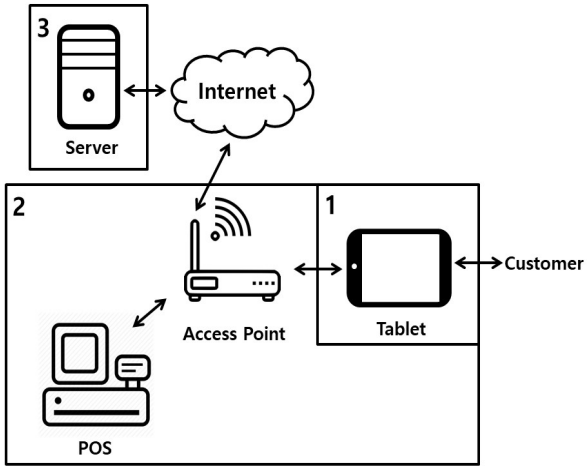


Fig. 3. Structure of points accumulating system

The tablet is positioned in area 1, which can be accessed by all customers and is managed by the shop owner. The manager application is operated on the tablet which takes phone numbers from customers and then adds a certain amount of point automatically when POS allows. The shop owner has to sign-in on the application and synchronize it with a program running on a POS. Whether the connection between the POS and the tablet is direct or via the server is up to the company.

A POS device and the router are in area 2 which can be only accessed and managed by the shop owner. No customers are allowed to access and control these. In other words, area 1 is outside of checkout, area 2 is inside of checkout, and both areas represent inside of the store. The program operated on the POS monitors customers' information and allows the tablet to add points. The shop owner also has to sign-in on the program and synchronize it with the application running on the tablet.

Information that the tablet takes is saved at a server, physically allocated in the company only for one shop, and the shop owner can use part of this information for certain purposes. The server is in area 3, which is in a remote place from the shop and only accessed by the company which provides and monitors all points system running. The shop owner cannot access the information on this server directly.

III. Expected Vulnerabilities and Mitigations

This chapter explains the types of vulnerabilities that might exist or already exist in this system based on pre-inspections. First, this system depends on the telecommunication among nodes, the tablet, POS, and the server. So we deal with whether this system guarantee integrity, confidentiality when they exchange packets. Second, according to the static analysis of tablet applications, many companies load a light web browser in the application. It means each activity of an application is just a web page. So we thought

inspections of the safety of the web are needed. Finally, non-technical security issues such as people looking down security riskiness are also addressed.

Each vulnerability has attack scenarios and expected damages. We would like to introduce them first, and then propose proper mitigation techniques that can protect against each vulnerability. The concept that classifies vulnerabilities is helpful to find out and analyze vulnerabilities, however, not all vulnerabilities can be classified into several types clearly.

3.1 Network

All communications over the Internet protocol are always threatened by the possibility of sniffing and man in the middle(MITM) attack. The sniffing attack can be conducted simply by gathering packets in the air or spoofing attack over a LAN might precede before sniffing packets. If the sniffing were successful, an attacker might be able to be a middleman and monitor all packets between them. It makes replay attack possible.

For instance, if an attacker analyzes packets and parameters of them such as a phone number and the amount of points to add, the attacker can send malformed packets to add points. On the contrary, if packets were perfectly hidden and manipulated packets were ignored by the server, the system would be secure almost absolutely. The aim of security techniques at the network level is hiding data to the third party and distinguishing fake packets perfectly.

3.1.1 Cryptographic protocol

When an user put personal information into the input fields of an webpage of the tablet to sign in, it is transmitted to the web server via a router. If this information were transmitted in plain text, it would be easily sniffed through capturing packets. This means a malicious user can easily take a random user's information and authority of a shop owner. It also leads to illegal spending and adding points by analyzing packets' structure.

So, using cryptographic protocols such as HTTPS and SSL pinning are highly recommended because these are efficient[4] to make eavesdropping and analyzing packets difficult[5]. As a result, creating malicious malformed packets and sending it to the server get impossible. Also, the cutting-of-edge technique used to prevent bypassing SSL pinning pretty intensifies the security level. Applying three mitigations: HTTPS, SSL pinning, anti-bypassing SSL pinning simultaneously helps to construct the security level of commercial banks in the world such as Bank of America and National Australia Bank[6].

3.1.2 Dynamic Cookie Value

Updating cookie value for every communication, called set cookies dynamic, is recommended to defend replay attack. If the system is built for a client to accept new cookie values for each request and send back a request with it, a simple replay attack just sending a captured packet without manipulating parameters is impossible. Dynamic cookie forces an attacker to analyze the structure of packets and strong cryptographic protocols make packet analysis impossible. As a result, the combination of both mitigations above considerably reduces the risk of network related attack.

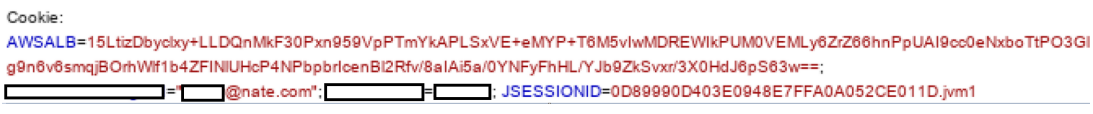


Fig. 4. A request packet with cookie and parameters

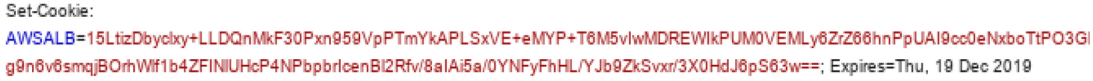


Fig. 5. A response packet renewing next cookie value

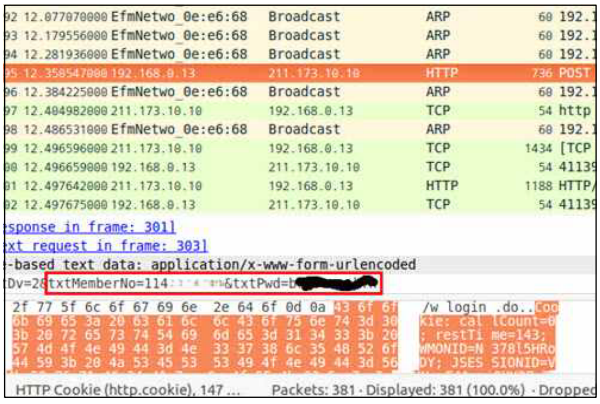


Fig. 6. Packets captured at Korail reservation website reveal personal information without encryption[7]

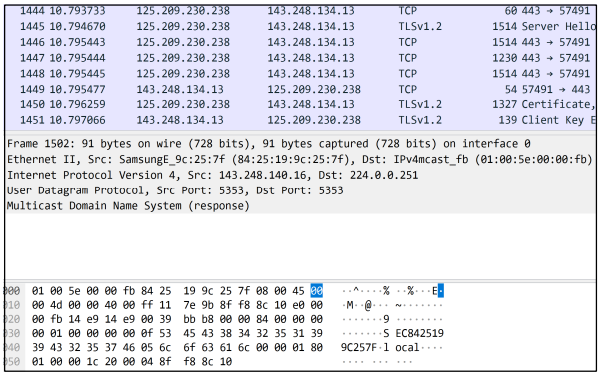


Fig. 7. Packets concealing data with TLS cryptographic method

These mitigation techniques must be applied because repeating the same attacks is not difficult and the expected damage is massive. This kind of vulnerability is discovered at Korail website for a ticket reservation and reported to the Korea Internet and Security Agency(KISA).

3.2 Web

The web is the easiest way to construct a system or a service because there are lots of available developers and documents. For this reason, there are lots of platforms based on the web and its importance is getting high[8]. However, the web has several inherent vulnerabilities.

First, a session value is used to keep the sign-in state. so leaking of the session has the similar impact as leaking of ID and password of an user. Next, a client must reveal itself and take several inputs, so it might lead to Cross-Site Scripting(XSS) attacks. The lines of script codes from the client side are sometimes sent to and executed at the server. It not only causes unexpected minor errors also makes server authority vulnerable.

3.2.1 Session Management[9]

Sessions must be created and managed properly. Developer guidelines for session management will be addressed. A session is a method of storing user information. When a user signs in successfully, the session is randomly generated by the web server. It is used to make up the shortcoming of the HTTP protocol, statelessness. The browser maintains a sign-in state using the session value, which means an unauthorized sign-in is possible until the session expires once an attacker takes a session. We would like to propose several techniques that prevent session hijacking attacks used to seize sessions.

First, a session must be random, unpredictable, meaningless, and enough long. Also, a session must be different whenever it is generated and expired after an appropriate time. Finally, the cryptographic protocol mentioned at 3.2.1 makes session management more secure. Besides, the HTTP only attribute can be applied to prevent script code from accessing cookie values.

3.2.2 Cross Site Script

According to the static analysis of the tablet application, we found out that the tablet receives lines of script code from the server and executes without any filtering. It is potential vulnerabilities because the authority of the tablet can be seized by lines of malicious script code. Web page defacement and changing hyperlink of some elements into fake web pages might happen. Key logging that illegally takes customers' input on the tablet is also possible. As an analysis, executing command window is probably possible.

To prevent this, first, the system should be revised to identify who sends the lines of script codes. It fundamentally blocks unauthorized nodes from sending codes. Next, basic XSS mitigations such as filtering and using CSP are recommended first. Also, there are lots of recent researches focusing on detecting potential malicious inputs. A detector with machine learning devised in the researches shows outstanding effectiveness[10].

3.3 Social Engineering

There must be non-technical attacks that take ungiven information. There are several techniques to prevent this sort of attack, which might cause discomfort of users because it is ambiguous to distinguish normal users' abnormal behaviors and abnormal users' normal behaviors. For instance, a user who tries repeated wrong signing in is blocked by the system, however, the service provider doesn't know whether this user is an attacker or the right user. If the user is the right user, the service provider just has caused inconvenience to the user. In the case of APT attacks using email, lowering the threshold of detecting suspicious email and blocking them also increase the discomfort of email service users. Finally, it leads to less frequency of blocking. Eventually, finding an appropriate point between convenience and safety comes the most significant issue.

Also, some vulnerabilities come from the ignorance for the security of the public[11]. For example, many people even don't know they should change the admin password of a router when they introduce it to provide public AP. Similar problems might happen in this system. If a shop owner who introduces this system doesn't change the admin password of the tablet application or the POS program, the admin authority might be easily seized by attacks such as a dictionary attack. In the worst case, an attacker might perform brute force attack for ID with fixing the password as default password. Even though changing passwords into a strong one is efficient[12], they just don't do this. The developers should think the identification process with id and password might not be safe and devise a more secure extra identification processes. The aim of techniques at this layer is preventing possible social engineering attacks with minimizing inconvenience of users.

3.3.1 Additional Certification Process

A service provider can require an extra password when users try to use some sensitive works, which helps to mitigate damage even if account information is already leaked. For example, one of the telecommunication companies in Korea requires to set an extra password for online payment to prevent damage from SMS sniffing when using an online payment service. Also, the extra password can be replaced with one time password call OTP using SMS or Internet protocol.

3.3.2 Password Management

A service provider can force users to change password when they use the service first. As an survey, 61% of applications allow users to use default password and pretty weak password[13]. Forcing users to change default password into strong password helps to reduce damages from dictionary attacks[12].

3.3.3 Defending Brute Force Attack

Brute Force Attack is a primitive attack technique, but it is still a valid and frequent attack method on the network environment[14]. This attack might be tried if the server of the systems takes inputs and identifies using these inputs. It might not only harm confidentiality but give too much burden to the server, which is the same effect as DoS(Denial of Service) attack. There is a lot of research going on to defend this technique, however, the easiest way which a developer can apply is to use reCAPTCHA.

It is a function that presents the random text in the form of an image that cannot be read by a computer easily and allows us to keep doing works only when an input is the same text as an image. This image probabilistically distinguishes a human and a robot, which means it can delay or even cancel robots' tasks.

We can somehow defeat damages of brute force attack if we set reCAPTCHA activated when repetitive tasks conducted by a web scrapper and a script code are detected. As an experiment, an webpage with no reCAPTCHA is easily hacked by the python macro, however, web pages with reCAPTCHA successfully defend.

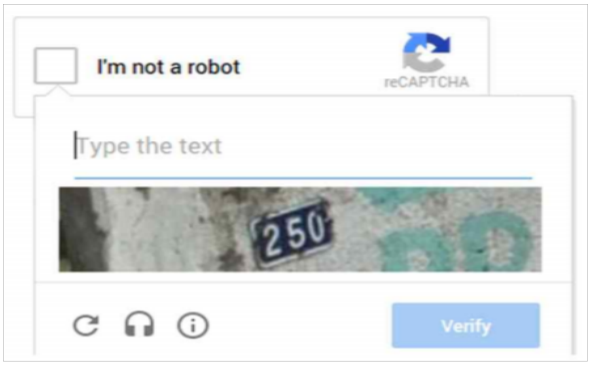


Fig. 8. An example of reCAPTCHA[15] from Google

```
from selenium import webdriver

driver = webdriver.Chrome("C:WWUsersWWKITRIWWDesktopWWchromedriver.exe")

for c in range(65, 105):
    driver.get("https://nid.naver.com/nidlogin.login?mode=form&url=https%3A%2F%2Fwww.naver.com")
    button = driver.find_element_by_class_name("btn_global")
    driver.find_element_by_name("id").send_keys("bob_8th_stu")
    driver.find_element_by_name("pw").send_keys("" + chr(c))
```

Fig. 9. Python code trying brute force attack

Fig. 10. Activated reCAPTCHA when figure 9 is executed

3.3.4 Monitoring Sign-in State

We can prevent unauthorized sign-in using multiple sign-in prevention and additional authentication requests based on the location where a trial of sign-in happens. Multiple sign-in protection is a feature that prevents signing in with one account to more than one device at the same time. It considers sign-in on an extra device while already signed in as an abnormal situation and blocks, which can prevent an unauthorized extra sign-in.

If account information is leaked in a long-distance network, an attacker tries to sign in far away from the original user. Recognizing this happens, requiring additional verification and declining sign-in help to reduce the risk of account information leaks.

IV. Conclusion

This paper first explains the expansion of electronic commerce and its merits which offline shop owners covet: being easy to create a dataset of customers and products they bought, add points quickly, analyze them automatically and find out proper marketing strategies. After that, the details of the automated points accumulating system and the security issues are discussed. The potential security vulnerabilities mentioned above are mainly related to financial issues. For example, if the admin password of this system in the store were leaked, a malicious attacker could issue an unlimited amount of points. Also, a malicious attacker can spend points illegally if he or she takes user accounts information. Finally, the leak of users' personal information causes lots of unforeseen additional damages. These kinds of security accidents have happened many times even in the system that conglomerate developed and managed[16].

There have been a lot of regulations and suggestions that emphasize secure online financial transactions. But the target of them has been just money itself, and online properties such as points

were not in the spotlight. For this reason, the current point systems are pretty more vulnerable than other financial transactions on the Internet. So we emphasize the importance of managing points securely, offer potential vulnerabilities, and suggest a guideline for future development and further patches.

Acknowledgement

We thanks to anonymous reviewers and their constructive comments that help to improve this paper. This paper is improved and partially translated from the preliminary version[17], which was presented at CISC-W'19.

References

[1] Thompson S. H. Teo and Yuanyou Yu, "Online buying behavior: a transaction cost economics perspective", Omega, Vol. 33, No. 5, pp. 451-465, Oct. 2005.

[2] How big is the global e-commerce market?, https://hackernoon.com/how-big-is-the-global-e-commerce-market-93920e61f687 [accessed : Oct. 28, 2019]

[3] Korea Online Shopping Association, 2015 Online Shopping Trends and Prospects, http://www.korcham.net/FileWebKorcham/OnlineSeminar/File/20151202_05.pdf. [accessed: Oct. 28, 2019]

[4] Veelasha Moonsamy and Lynn Batten, "Mitigating man-in-the-middle attacks on smartphones-a discussion of SSL pinning and dnssec", Proceedings of the 12th Australian Information Security Management Conference, Edith Cowan University, Perth, Western Australia., pp. 5-13, Dec. 2014.

[5] Lucky Onwuzurike and Emiliano De Cristofaro, "Danger is my middle name: experimenting with SSL vulnerabilities in Android apps", Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks, ACM, Article No. 15, pp. 1-6, Jun. 2015.

[6] Francisco José Ramírez-López and Ángel Jesús Varela-Vaca, et al., "A Framework to Secure the Development and Auditing of SSL Pinning in Mobile Applications: The Case of Android Devices", Entropy, Vol. 21, No. 12, pp. 1136-1136, Nov. 2019.

[7] A Vulnerability of a Website for Train Ticket Reservation on Lunar New Year's Day, Using Plaintext Transfer Protocol, https://www.boannews.com/media/view.asp?idx=45037&kind=1 [accessed : Oct. 28, 2019]

[8] Marius Steffens and Christian Rossow, et al., "Don't Trust The Locals: Investigating the Prevalence of Persistent Client-Side Cross-Site Scripting in the Wild", Network and Distributed Systems Security (NDSS) Symposium 2019, San Diego, CA, USA, Feb. 2019. https://dx.doi/10.14722 /ndss.2019.23009.

[9] Shellie Wedman, Annette Tetmeyer, and Hossein Saiedian, "An analytical study of web application session management mechanisms and HTTP session hijacking attacks", Information Security Journal: A Global Perspective, Vol. 22, No. 2, pp. 55-67, Mar. 2013.

[10] Zhang, Jingchi, Jou, Yu-Tsern, and Li, Xiangyang, "Cross-Site Scripting (XSS) Detection Integrating Evidences in Multiple Stages", Proceedings of the 52nd Hawaii International Conference on System Sciences, pp. 1-10, Jan. 2019. https://dx.doi/10.24251/HICSS.2019.860.

[11] Jinsook Cho, "Likelihood to abort an online transaction: influences from cognitive evaluations, attitudes, and behavioral variables", Information & Management, Vol. 41, No. 7, pp. 827-838, Sep. 2004.

[12] Chun-Li LIN, Hung-Min SUN, and Tzonelih HWANG, "Attacks and solutions on strong-password authentication", IEICE transactions on communications, Vol. 84, No. 9, pp. 2622-2627, Sep. 2001.

[13] Brandon Knieriem and Philip Levine, et al., "An overview of the usage of default passwords", in International Conference on Digital Forensics and Cyber Crime", International Conference on Digital

Forensics and Cyber Crime, Prague, Czech Republic, pp. 195-203, Oct. 2017.

[14] NAJAFABADI, Maryam M., et al., "Machine learning for detecting brute force attacks at the network level", 2014 IEEE International Conference on Bioinformatics and Bioengineering, IEEE, pp. 379-385, 2014.

[15] Google reCaptcha Bypass Technique Uses Google's Own Tools, https://threatpost.com/google-recaptcha-bypass-technique-uses-googles-own-tools/12 4006/ [accessed : Oct. 29, 2019]

[16] Sameer Saxena, Sonali Vyas, B. Suresh Kumar, and Shaurya Gupta, "Survey on Online Electronic Paymentss Security", 2019 Amity International Conference on Artificial Intelligence (AICAI), IEEE, Dubai, pp. 756-751, Feb. 2019.

[17] Yonghyeon-Park, Minju-Jo, Namhyeong-Kim, Jaeyoung-Jeong, Kyeongmin-Lee, Junbeom-Lee, and Hongjin-Kim, "Security Methods for Secure E-Commerce: An Aspect of Online Property", CISC-W, Seoul, Korea, Nov. 2019.

Authors

Yong-Hyeon Park

2015 ~ 2019 : BS degree in Sungkyunkwan University, Department of Computer Science

2019 ~ present : Master degree in Department of Information Security, KAIST.

Research interests : Network security, GPU optimization

Jae-Young Jeong

2017 ~ 2019 : BS degree in Hanyang University, Department of Nuclear Engineering

2020 ~ present : BS degree in Hanyang University, Department of Computer Science

Research interests : Reverse engineering

Hong-Jin Kim

2001 ~ present : LG CNS Co., Ltd

2013 ~ 2015 : Master's degree in Department of Management of Technology, Korea University

2015 ~ 2018 : Ph.D in Department of Business Administration, Soong-sil University

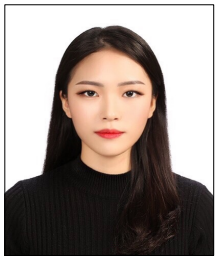
Research interests : Social Engineering Attack, Security Consulting, Project Management, Communication Skill

Nam-Hyeong Kim

2014 : BS degree in Soonchunhyang Univeresity, Department of Information Security

Research interests : Network security

Min-Ju Jo



2016 ~ present : BS degree in Dongguk University in Gyeongju, Department of Computer Science

Research interests : Blockchain

Kyung-Min Lee

2014 ~ present : BS degree in Dongguk University in Gyeongju, Department of Computer Science

Research interests : Software Engineering

Jun-Beom Lee

2013 ~ present : BS degree in the Catholic University of Daegu, Department of Computer Science

Research interests : Big Data