A Practical Solution Against Business Email Compromise (BEC) Attack using Invoice Checksum

Songpon TEERAKANOK
Research Organization of Science and Technology
Ritsumeikan University
Shiga, Japan
songpon.te@cysec.cs.ritsumei.ac.jp

Hiroaki YASUKI
College of Information Science and Engineering
Ritsumeikan University
Shiga, Japan

Tetsutaro UEHARA
College of Information Science and Engineering
Ritsumeikan University
Shiga, Japan
t-uehara@fc.ritsumei.ac.jp

***Abstract*—This research presents a practical countermeasure against the problem of the bogus invoice scheme, one of the most threatening BEC attacks in modern business. We introduce a straightforward yet highly practical method of creating a checksum from the invoice and shared secret information. Using the generated checksum allows the recipient to confirm the authenticity and integrity of the invoice before proceeding with the actual payment. In this work, generating and verifying of checksum information are done via a smartphone application. Also, the predetermined secret information is stored inside the smartphone to provide better protection against data theft. Lastly, an Android-based application for checksum generation and verification, supporting both manual input and QR code scan, is implemented to demonstrate the use case scenario and practicability of the proposed method.**

***Index Terms*—BEC attack, phishing, email compromise, bogus invoice scheme**

I. Introduction

Business Email Compromise (BEC) attack is one of today's most common security issues in modern businesses. Even though this attack is not something new, it, however, poses threats to many businesses, which can potentially result in considerable damage and loss (e.g., multimillion-dollar) for the company.

Regarding BEC, the attack usually involves exploiting human error and tricking victims before commencing the attack. In this attack, the criminal first impersonates him/herself to be someone associated with the target (e.g., a business partner, an executive, or an attorney). The impersonated criminal then sends an email to the target demanding a payment into his/her fraudulent bank accounts, which are usually possessed by money mules. With well-crafted emails, it is very challenging and not likely for the victim to become aware of the fact that he/she is being tricked, and before the company realizes, the money is long gone.

One of the excellent examples of recent BEC attacks in real life is the case of Japan Airlines (JAL). In September 2017, JAL fell prey to a BEC scam via an email purporting to be from U.S. financial services company leasing aircraft to the airline [1]. In this incident, the attacker sent emails called for payment of lease fees into his/her fraudulent bank accounts in Hong Kong. This caused the airline to pay a total of ¥384 million (or approximately $3.4 million), which is considered a huge loss.

Another example of the recent incident in September 2019 with even a greater loss was the case of Nikkei America, the U.S. subsidiary of Nikkei. In this incident, the company lost $29 million to BEC scammers pretending to be an executive of the company [2], [3]. According to the news, it was found out that attackers somehow gained access to one of the company's email account and then used it to observe internal communication and finally commenced their attacks.

Generally, there are many types of known BEC schemes: account compromise, impersonation (attorney or CEO fraud, for example), and bogus invoice scheme, for example. In the bogus invoice scheme, an attacker creates a fraudulent invoice requesting payment for some products or services, which looked very legit in the victim's perspective, and send it to his/her target. Since the target seems to trust the attacker, it is difficult for the victim to notice that the invoice he/she received is a fake. To overcome this problem, changing or adding more security features to the invoice issuing system appears to be the most appropriate choice. However, since the Enterprise Resource Planning (ERP) software and invoice system of each company may be different, adding additional security features to the system may not be feasible.

In this paper, a practical solution for solving a bogus invoice problem is introduced. Utilizing a shared-key and short-length checksum of invoice data, we propose a straightforward yet usable security solution allowing a user to check an invoice's contents and confirm it with the checksum information. The checksum can come in forms of both series of numbers and QR codes, which are recommended to be printed/added on the invoice for easy confirmation. As mentioned, the invoice system of a company may differ from the others, adding additional information like checksum to the invoice may not be possible in some cases. Therefore, the proposed method is also designed to tackle this problem by allowing a user to send checksum (i.e., text or QR code) along with the corresponding invoice through phone or email. In addition, generating and confirming invoice checksum with the smartphone (e.g., Android) application is also an option.

The rest of this paper is organized as follows. Section

978-1-7281-8915-4/20/$31.00 ©2020 IEEE
DOI 10.1109/QRS-C51114.2020.00036

2 discusses the scenario and attack model of BEC. Next, the proposed method is introduced in section 3. Section 4 presents a prototype and implementation details of the proposed method. In the following section 5, we discuss the security and practicability of our approach. Finally, we conclude this paper in the last section 6.

II. BEC Attack Scenarios

Business Email Compromise (BEC) is a typical form of cybercrime, usually targeting individuals or companies who conduct business with their partner abroad. According to the FBI's Internet Crime Complaint Center (IC3), there are five main scenarios for BEC attack [4]: 1) business working with a foreign partner/supplier, 2) an executive requesting for a wire transfer (a.k.a. "CEO fraud"), 3) business receiving fraudulent email due to compromised email accounts, 4) impersonation as an executive or attorney, and 5) data theft.

The process of BEC attack involves exploiting human error via social engineering, utilize the gathered information, and create fraud information to trick the target into believing what is told. Generally, BEC schemes come in many forms: vendor email compromise (VEC), impersonation (as CEO or an attorney), bogus invoice scheme. In this section, we discuss the details and the attack scenario behind this fraudulent invoice scheme.

A. Concepts

The concept of a bogus invoice scam is rather simple. In this scheme, the criminal commences his/her attack by first gathering information regarding the upcoming transaction between two business parties (i.e., the victim company and its business partner). The information gathering can be done in many ways, including social engineering, sending a spoof email (using spear-phishing [5], [6]), and compromising email accounts of target employees to eavesdrop their conversation.

Once the attacker knows that the next invoice is coming out, he/she creates a fraudulent invoice that looks seemingly legitimate and subsequently sends it to the victim, asking him/her to pay money to the fraudulent bank account. Without proper verification protocol, the victim falls prey to the attacker scheme because he/she thinks the invoice is authentic since it was sent by someone he/she trusted. Figure 1 shows the overall process of the bogus invoice scheme.

The reason that attackers usually target companies doing business with oversea partners is that these companies are likely to be easier targets for this attack since verification of oversea transactions typically take more time and is considerably more complicated. Also, criminals usually use money mules to move the money around, from one bank account to the other, to anonymize and prevent themselves from being tracked.

B. Incidents

There are several reported incidents involving BEC in modern business. A member of Black Axe, a Nigerian criminal organization, was arrested in Canada in 2015, which later

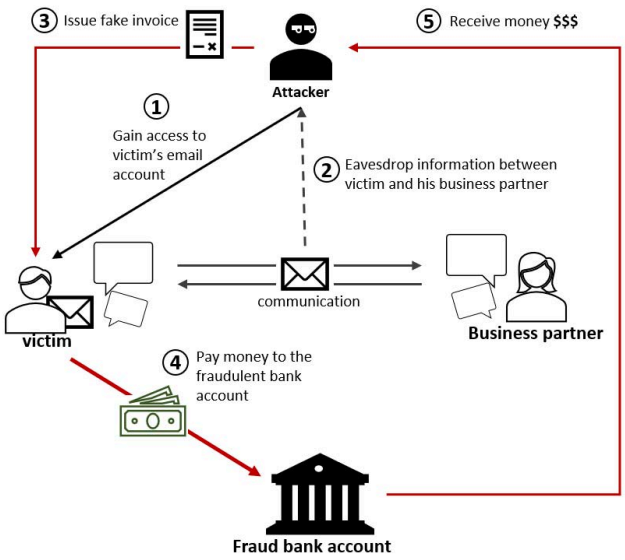

Fig. 1. Business Email Compromise (BEC) attack model.

revealed that its branch in Canada was heavily involved with BEC, money laundering, and fraudulent wire transfer [4].

London Blue, a criminal gang that operates out of Nigeria, was reported to use BEC scams against top executives. The strength and fearfulness of this criminal gang rely on their well-organized modern corporation-like operations, combining with the use of commercial data brokers to create lists of potential targets around the world. It is also mentioned that, in [8], these scams usually prey on the high-tension environments of large companies where employees are naturally forced to complete their tasks quickly rather than securely.

The Bank of Japan (BoJ) has issued a warning, since 2013, regarding scam mails using the BoJ's name [9]. Also, in early 2016, FACC, a global company leading in aerospace and aircraft components, and Crelan bank in Belgium fell prey to BEC scams costing them $54 million and $75.8 million, respectively [10]. The study in [10] shows that employees with the position of chief financial officer (CFO) are most likely to be a target of BEC scams, specifically via spear-phishing attacks.

Recent research in business-related cybercrime [11], in late 2019, points out potential threats about how contact center employees can unintentionally leak confidential information to the scammers due to their lack of security-awareness and discipline. It is claimed that 42% of agents chose not to report their situations when they experience a breach attempt, either by insiders or outsiders.

Also, according to the recent report from Mimecast [12] in June 2019, it found that nearly all of the organizations studied by Mimecast experienced phishing attacks, with 88% of them reported received fraudulent emails faked as vendors or their business partners. Furthermore, it is shown in Mimecast's report that both methods and targets of BEC keep evolving.

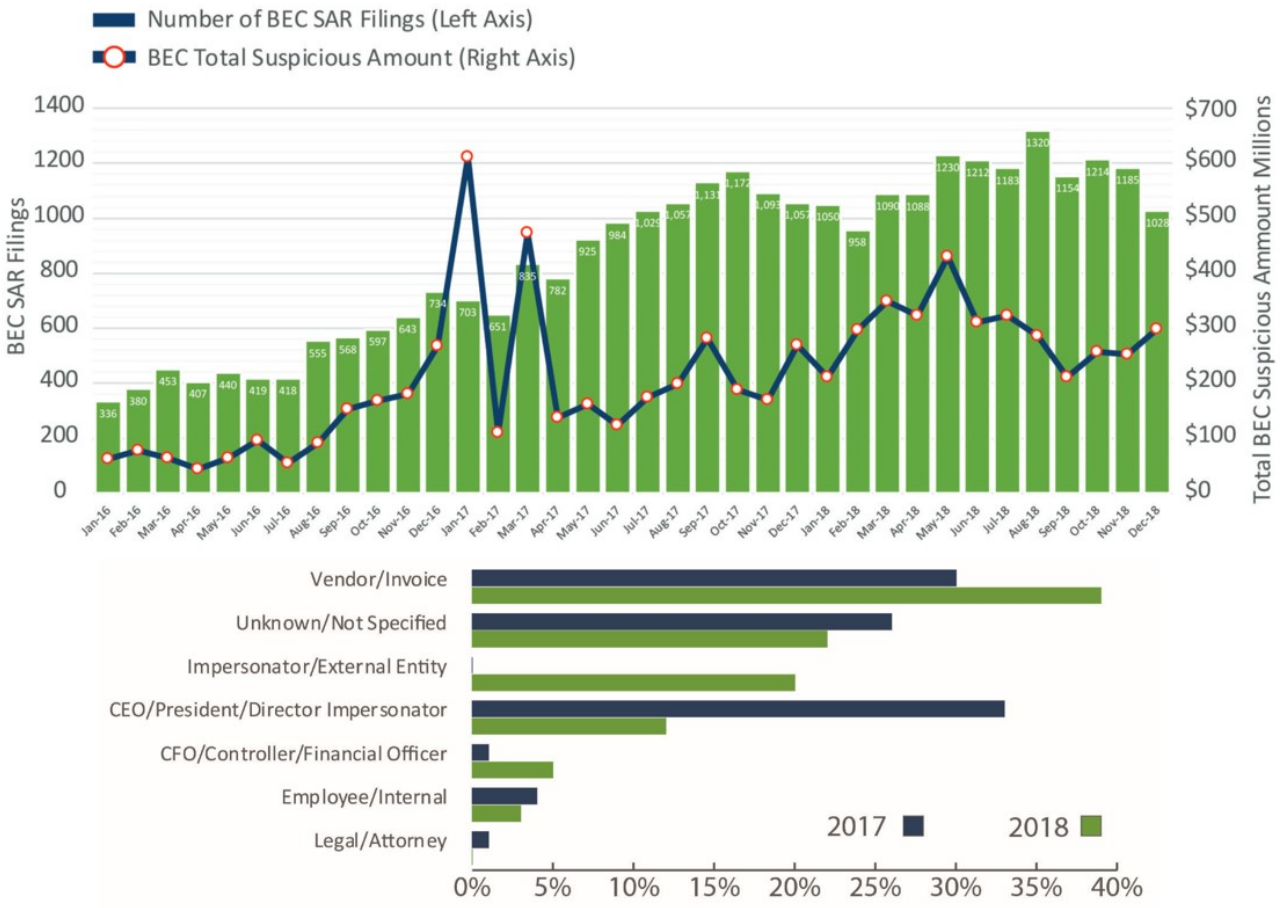

Fig. 2. 2016 to 2018 BEC scams information (original chart image from 2019 FinCEN's Financial Trend Analysis report on Manufacturing and Construction Top Targets for Business Email Compromise [7]).

BEC scams are no longer a threat only for businesses but also an individual [12]. With phishing-as-a-service kits accessible from the dark web, it even provides a more comfortable way to achieve it.

Lastly, according to the Bank Secrecy Act (BSA) reports (July 2019) from the Financial Crimes Enforcement Network (FinCEN) [7], the number of suspicious activity reports (SARs) has increased at a terrifying rate with an average of nearly 500 cases/month in 2016, and above 1100 per month in 2018. Also, the total value of attempted BEC thefts rose to $301 million to $110 million per month (270% increase, approximately) from 2016 to 2018.

In FinCEN's report, BEC scam methods evolve over time since BEC actors are likely to change their methods as awareness of their scheme increases. For example, the number of reports of fraud CEO scams, which accounted for 33% of incidents in 2017, had declined to 12% in 2018 while the cases of fraudulent vendor invoices had risen significantly from 30% in 2017 to 39% in 2018.

Among all known BEC schemes, fraudulent vendor invoice (a.k.a. "bogus invoice scheme") is a top methodology. Despite accounting for 30% of total BEC attempts reported in 2017, the fraudulent vendor invoice method accounted for approximately 41% of the overall transaction values. Figure 2 shows the increment of each type of BEC scams in 2017 and 2018.

C. Recent research on BEC prevention

BEC prevention methods can be classified into two primary categories: technical and non-technical approaches. Regarding technical methods, there are several ways to strengthen a corporate email system, which can help to reduce the risk of receiving fraudulent emails [13], [14].

In 2017, AlSabah et al. [15] introduce a secure e2e email communication method. Using their proposed certificate-less (CL) key agreement protocol, the technique allows users to update their public keys without the need to contact the certificate authority (CA). Cohen et al. (2018) [16] introduce a method to detect malicious emails using machine learning-based approaches. Extracting features from the entire emails (i.e., header, body, and attachments) combining with the Random Forest classifier, the method claims to have a very high

level of accuracy at 92.9% with the true-positive and false-positive rate at 94.7% and 3%, respectively.

Furthermore, regarding today's standard, DMARC [17], an open authentication standard built upon two email authentication standards (SPF and DKIM), is by far considered the most effective email authentication technology in the market. DMARC provides prevention against domain-spoofing emails from reaching users' inbox. Using DMARC, it can block, quarantine, and also monitor any malicious email that was sent from the domain under control. Many email providers, such as Google's Gmail-hosted mailboxes and Microsoft's Office365, provide supports for DMARC policy.

On the other hand, non-technical approaches involve training to raise employees' security awareness. The practice may vary from company to company, which may include training basic incident handlings, a countermeasure against some suspected malicious actions (e.g., sending a report to the higher-up, and basic methods on how to confirm and detect malicious email). As mentioned before, BEC schemes are evolving with time. Therefore, there is also a need for a corporate to retrain its employees periodically to maintain their security awareness.

In reality, many companies put in efforts to raise the security of their email system (deploy email filtering, for example) and also security awareness of their employees to prevent themselves from falling prey to BEC scams. An employee is usually told to double-check the sender's email address, not to open any link or attachments inside the email from unknown parties, and many more. However, as we already know, BEC scammers also put even more effort (hacking, spoofing email, social engineering [5], for example) into earning trust and deceiving victims to believe everything they are told. The current treatments for this particular type of BEC scam still leave much to be desired. Therefore, a practical solution is needed. In the next sections 3, we present and discuss the proposed feasible solution to counter a fraudulent vendor invoice scheme. Furthermore, a sample prototype (i.e., mobile application) of our proposed mechanism is presented in the following section 4.

III. Proposed Method

Our proposed fraudulent invoice scheme prevention method consists of three primary phases: initialization, checksum generation, and verification. Sections III-A to III-C explain each step in detail.

A. Initialization

First, let us give a scenario such that a company C_A is considering buying supplies from company C_B located oversea. The contract between them will be renewed periodically. Therefore, C_B is not considered as a one-time business partner.

In this phase, the process of the initialization phase is very straightforward. When the initial business contract is determined, company C_B creates a random secret-key Sb and sends it to C_A. The generated secret key may come in many forms: short text, number sequences, or a single random number. This secret key will be used in every transaction between two parties during the length of the contract and will be reissued once the contract is renewed. Note that the generated secret key information should be sent through a secure and reliable method such as international courier services offering proof of delivery information (such as DHL, FedEx, and UPS) to ensure the information reaches the intended recipient safely.

B. Checksum Generation

Next, C_A proceeds with the payment process by asking C_B to issue an invoice for it. In addition to the traditional invoice generation method, in this step, checksum information is added to the invoice. In this work, the checksum comes in the form of an 8-digit number sequence generated from the previously shared secret key (S_B) and the basic yet most critical contents of the invoice, i.e., invoice number (IN), price (P), issuing company's name (C), bank's swift code (SW), date (D), and recipient name (R).

Regarding the checksum generating algorithm, first, we create a JSON string of the raw checksum data, as shown in the following example.

```
{
    "secret_key":"A4538FD32CBE2678"
    "issuer":"Cysec",
    "invoice_no":"A1135-FE4329",
    "price":"150000JPY",
    "swift":"SIGAJPJTXXX",
    "date":"2020-2-5",
    "recipient":"Songpon",
}
```

Next, we digest this JSON string with *SHA256* message-digest algorithm resulting in 256-bit data, denoted as M. Lastly, we generate an 8-digit number sequence from the 256-bit digested data as follows.

$$M = SHA256(S_B, C, IN, P, S, D, R) \tag{1}$$

$$checksum(CHK) = M \mod 10^8 \tag{2}$$

The reasons for reducing the size of 256-bit hashed data down to an 8-digit sequence are simply to make it more friendly for both an issuer and the recipient and to avoid any misunderstanding. The issuer can not only insert this information into the invoice via the computerized system but can also write it on the paper, send it by email, or even tell it over the phone if necessary. For the intended recipient, C_A can avoid any unnecessary misunderstanding that might cause by the issuer's handwriting. Since all information is written in integer, there is no need to guess whether a checksum C_A reading is a number, alphabet, or something else.

To add the generated checksum information to an invoice, printing this 8-digit number sequence (or QR code containing the same information) directly on the invoice is highly recommended. By printing checksum directly on the invoice, it will

help avoid unnecessary complication and misunderstanding, making it more applicable in the real scenario.

However, in some cases where adding additional information is not possible, the 8-digit checksum (or QR code) can be sent together with an invoice through email or any messaging application (e.g., Slack). This provides users with more options making it more applicable for many modern businesses. Note that the contents of JSON data (except the secret key) and the invoice should be identical. Otherwise, checksum verification will fail.

C. Verification

When the invoice from C_B arrives, the first thing the recipient (i.e., C_A) should do is to verify the integrity of this invoice. By using the same information written on the invoice and the predetermined secret key, C_A creates the same JSON string and hash it with *SHA256* message-digest algorithm. Following the same procedure, CA performs modulus operation with the *SHA256* output to obtain an 8-digit checksum sequence. Finally, CA confirms the integrity and authenticity of the invoice by comparing the received checksum information and the newly generated sequence.

If these two pieces of the checksum are perfectly matched, C_A, therefore, can trust this invoice and proceed with the actual payment. On the other hand, in case the checksum verification fails, the received invoice should not be trusted. Also, C_A should keep in mind that the email system might already be compromised. The recipient is, therefore, highly recommended to contact their partner C_B by different means (e.g., phone) for further inquiries. At this point, it marks the completion of the entire invoice verification process.

IV. Implementation

In this work, we create a proof of concept application on the Android platform to demonstrate the practicability of the proposed BEC prevention method. The mobile application was implemented using Kotlin programming language and tested on a HUAWEI JDN2-W09 tablet, running Android 9.0 (API level 28). There are two primary features regarding the developed application: checksum generator and invoice verifier.

A. Checksum Generator

During the invoice issuing process, the checksum is generated from the contents of the invoice being issued. The checksum may come in the form of a QR code, barcode, or text (i.e., number sequence). The issuer, then, attaches the checksum information to the invoice or sends them together through an email. For demonstration purposes, we created a manual checksum generator in which the user is required to manually input invoice contents before using them to generate the QR code result. Figure 3 shows the demonstration of the invoice issuing process.

B. Invoice Verifier

The verification process is rather easy to understand, involving only two steps: scan and confirm. Using our developed application, a recipient can perform a simple invoice verification by scanning the QR code on the invoice. The application will extract all necessary information from the QR code and show all the details on the screen. Finally, the recipient confirms the information on the screen with the target invoice to complete the process. Also, we implemented the manual invoice verifier in which the user needs to input all data (including checksum number sequence) by him/herself to complete the process. Figure 4 shows a demonstration of using our developed application to verify the integrity of an invoice.

V. Discussion

In this section, the security and practicability of the proposed method are discussed. First, we discuss the security behind our proposed invoice checksum mechanism, including some limitations and drawbacks of our approach. Next, the practicability and user-friendliness aspect of the proposed method, as well as some use case scenarios, are discussed.

A. Security Analysis

In this work, the security of our approach relies heavily on the sharing of secret information during the initialization phase. Unlike the computerized methods (e.g., sharing secret over the network), secret-sharing is done physically by sending a sealed letter via postal/courier service, which may cause some difficulties. Despite these difficulties, using the secure and reliable courier service, however, allows a company to track and ensure that the information will reach its business partner safely. Table 1 shows a comparison between each type of countermeasure against BEC attacks.

Generally, the mail filtering techniques (e.g., DMARC) are designed to work with the header information of the email. Email filtering policy will observe the incoming and outgoing email and try to block emails from any suspicious or look-alike (fake) domains. However, this method is weak against impersonation, in which emails may come from a domain outside the filter scope. Also, since the mail filtering technique involves only the header of the email, therefore, it cannot prevent the email system against some contents tampering-based attacks such as a bogus invoice scheme.

Regarding non-technical means to prevent BEC attack, training is one of the most common, necessary, yet unreliable methods. At the end of a training session, one cannot guarantee or claim with confidence that the trained employees will not fall prey to BEC scams. It can only be said that the risk of being prey is lower comparing to untrained personnel.

The proposed method, while it cannot prevent malicious email from arriving at the users' inbox, it provides a verification method allowing users to ensure the integrity, authenticity, and non-repudiation of the invoice coming with an email. Our proposed method is not a silver bullet solution to the problem of BEC, and it does not have to. We can combine the proposed

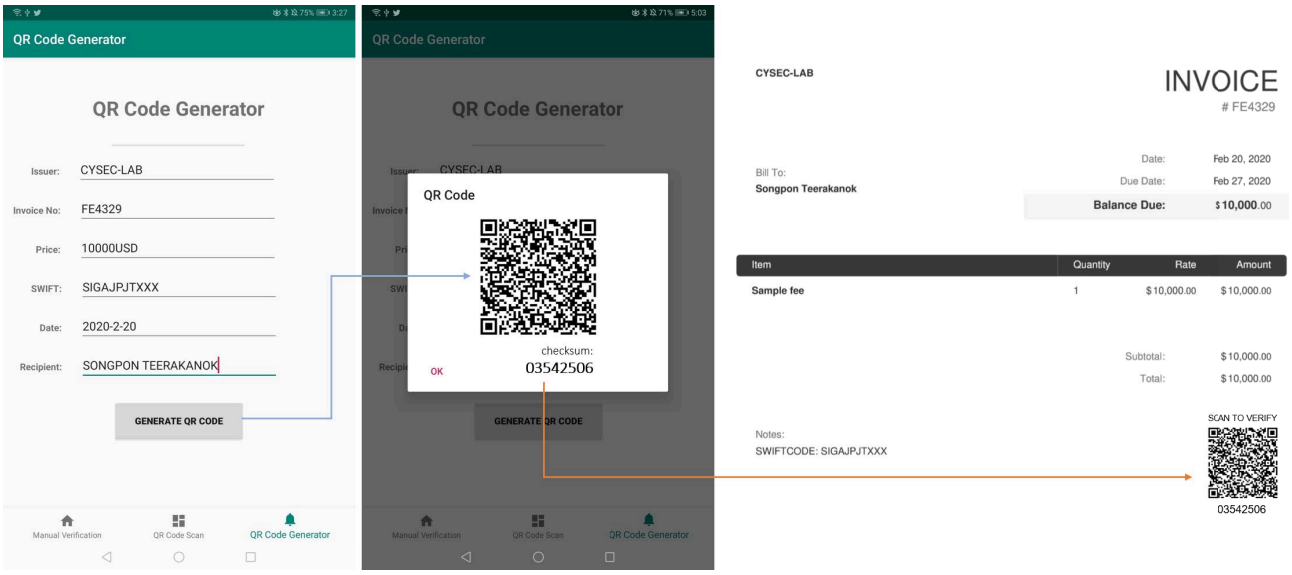

Fig. 3. Invoice issuing procedure.

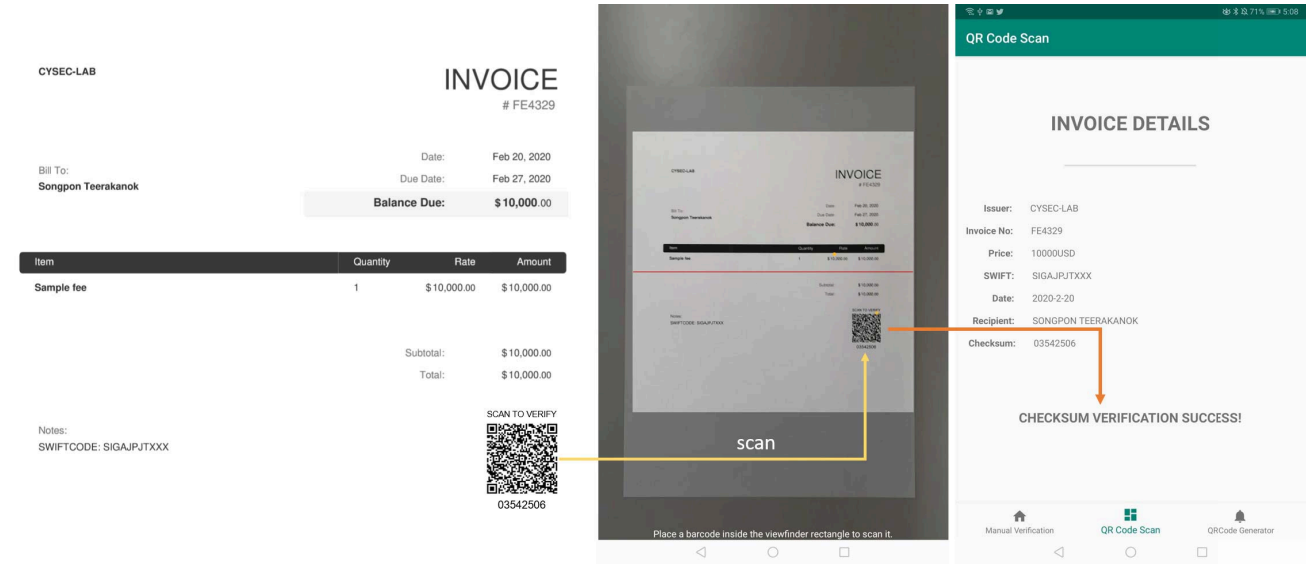

Fig. 4. Invoice verification process.

method with existing techniques, such as DMARC, which will eventually result in a better solution to the BEC problem.

Concerning the bogus invoice scheme, table 2 shows a comparison between our proposed method and email encryption techniques (e.g., TLS and S/MIME). Transport Layer Security (TLS) is a well-known method in providing email security, offering encrypted data communication between servers. While TLS is an excellent method for securing email transmission, not all email providers support TLS. Traveling through a server without opportunistic TLS support, the emails/messages will be automatically delivered in an insecure fashion.

Secure/Multipurpose Internet Mail Extensions (S/MIME) [18] is an encryption standard, similar to PGP, providing security to the email's content. Using public-key cryptography as a foundation, S/MIME requires the use of Certificate Authorities (CAs) to issue certificates for each party (i.e., sender and receiver). This approach requires both parties to trust the CA. Although some company prefers to use their self-issued certificates, those certificates are usually considered untrustworthy, which may create a security gap in the system. Furthermore, S/MIME does not protect the user against vendor email compromise attacks (VEC), especially when the user

TABLE I
COMPARISON BETWEEN THE PROPOSED METHOD AND TRADITIONAL APPROACHES.

Methods	Prevention against		
	Fake Domain	Impersonation	Bogus Invoice Scheme
Mail filter (e.g., DMARC)	✓	✗	✗
Proposed method	✗	✗	✓
Training	n/a	n/a	n/a

TABLE II
COMPARISON BETWEEN THE PROPOSED METHOD AND EXISITING EMAIL ENCRYPTION TECHNIQUES.

Methods	Bogus Invoice Scheme			
	Secure Transmission	Content Encryption	Content Integrity	VEC Prevention
TLS	✓	✗	✗	✗
S/MIME	✗	✓	✓	✗
Proposed method	✗	✗	✓	✓

utilizes WebMail's servers that keep users' private-key on the servers.

As we can see, in table 2, no method can provide a full range of protection against the fraudulent invoice scheme. TLS offers only encrypted transmission, while S/MIME provides confidentiality and integrity of the contents but weak against the VEC attack. Also, the proposed invoice verification scheme focuses on providing the integrity of the invoice and prevention against VEC. With our proposed method, the invoice cannot be reissued, even if an attacker successfully compromises one of the corporate email accounts. Hence, it is recommended to utilize all options, i.e., TLS, S/MIME, and the proposed method, to protect a company from the attack.

B. Use cases and Scenarios

In this section, we discuss the use cases and scenarios, as well as the benefits of the proposed method over the automated approaches using public-key encryption.

Automated approaches utilizing digital signature and public-key cryptography provide users with an excellent way to conveniently verify the integrity of an invoice, which requires little to no human interference. These approaches can significantly reduce the error caused by users. On the contrary, we proposed a method to verify the invoice's contents using checksum information. In our approach, the user can verify the integrity of the received invoice by computing the checksum using the mobile application or simply scanning the QR code, which is considered not automated and still requires human interaction.

However, there is a reason behind our design, which is due to the problem of practicability. Generally, many companies currently use only documents (e.g., paper, PDF file) and emails as their main channel for conducting business. Some companies may also utilize Enterprise Resource Planning (ERP) software, making it difficult to add additional security features to the system. With these restrictions, implementing an automated invoice verification system is much more complicated and may not be a feasible choice for these companies. On the other hand, with our proposed method, it is much easier to install a single application on a mobile device than to implement the entire automated system making our approach more applicable.

Lastly, the proposed approach is not developed to compete with automated cryptographic approaches. Our method is designed to assist users (i.e., companies) under the circumstance that the automated method is not a viable option.

C. Practicability

In this section, we further discuss the feasibility of our developed approach. First, the proposed method relies on the secure exchange of secret information during the initialization phase. This can be achieved by utilizing a sealed letter and a reliable international courier service. In case the seal on the envelope is removed, the recipient should not trust any received information and ask for a reissuing and resending of the secret information.

Although the secret information exchange may cause inconvenience to both parties, the process is performed only once during the entire period of the contract. Also, the proposed method can be deployed in addition to any existing email securing methods, such as TLS and S/MIME, to ensure the integrity of invoices. Finally, during invoice verification, we can avoid human errors by creating a well-design user interface (e.g., QR code scanner instead of manual input to reduce the risk of mistyping) for the verifying application.

VI. CONCLUSION

In this work, a practical solution to the problem of the bogus invoice scheme using invoice checksum is presented. The proposed method offers additional protection against BEC scams by ensuring the integrity of an invoice using its checksum. Our proposed method is performed by an invoice recipient first confirms the authenticity and integrity of the invoice by comparing checksum information. The recipient then proceeds with payments only if the received checksum is perfectly matched. Finally, we implemented an Android application to demonstrate the potential and practicability of the proposed mechanism.

References

[1] Kyodo. (2017) Japan Airlines falls victim to email fraud, paying out ¥384 million to Hong Kong accounts. [Online]. Available: https://www.japantimes.co.jp/news/2017/12/21/business/japan-airlines-bilked-¥384-million-getting-bogus-emails-seeking-lease-fees/#.XlSFsyNS8xs

[2] L. O'Donnell. (2019) BEC Scam Costs Media Giant Nikkei $29 Million. [Online]. Available: https://threatpost.com/bec-scam-nikkei-29-million/149834/

[3] F. Y. Rashid. (2019) Nikkei Hit By BEC Scam As Payments Get Larger. [Online]. Available: https://duo.com/decipher/nikkei-hit-by-bec-scam-as-payments-get-larger

[4] A. Meyers, "Not your fairy-tale prince: the Nigerian business email compromise threat," *Computer Fraud & Security*, vol. 2018, no. 8, pp. 14–16, Aug 2018. [Online]. Available: http://dx.doi.org/10.1016/S1361-3723(18)30076-9 https://linkinghub.elsevier.com/retrieve/pii/S1361372318300769

[5] A. Ferreira and S. Teles, "Persuasion: How phishing emails can influence users and bypass security measures," *International Journal of Human-Computer Studies*, vol. 125, no. December 2018, pp. 19–31, May 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1071581918306827

[6] J. Steer, "Defending against spear-phishing," *Computer Fraud & Security*, vol. 2017, no. 8, pp. 18–20, Aug 2017. [Online]. Available: http://dx.doi.org/10.1016/S1361-3723(17)30074-X https://linkinghub.elsevier.com/retrieve/pii/S136137231730074X

[7] Financial Crimes Enforcement Network (FinCEN), "Financial Trend Analysis," Tech. Rep., Jul 2019.

[8] "Major BEC gang targets top executives," *Computer Fraud & Security*, vol. 2018, no. 12, p. 19, Dec 2018. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1361372318301210

[9] Bank of Japan. (2013) Warning against Scam Mails Using the Bank of Japan's Name. [Online]. Available: https://www.boj.or.jp/en/about/organization/notice/index.htm/

[10] S. Mansfield-Devine, "The imitation game: how business email compromise scams are robbing organisations," *Computer Fraud & Security*, vol. 2016, no. 11, pp. 5–10, Nov 2016. [Online]. Available: http://dx.doi.org/10.1016/S1361-3723(16)30089-6 https://linkinghub.elsevier.com/retrieve/pii/S1361372316300896

[11] C. Knauer, "How contact centres can leave businesses exposed to cybercrime," *Network Security*, vol. 2019, no. 11, pp. 6–9, Nov 2019. [Online]. Available: http://dx.doi.org/10.1016/S1353-4858(19)30130-8 https://linkinghub.elsevier.com/retrieve/pii/S1353485819301308

[12] "Mimecast: The State of Email Security Report 2019," Tech. Rep. 6, Jun 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1353485819300698

[13] M.-H. Shao, G. Wang, and J. Zhou, "Some common attacks against certified email protocols and the countermeasures," *Computer Communications*, vol. 29, no. 15, pp. 2759–2769, Sep 2006. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0140366405004172

[14] M. Z. Sabir and M. Yousaf, "Design and implementation of an end-to-end web based trusted email system," *Procedia Computer Science*, vol. 141, pp. 231–238, 2018. [Online]. Available: https://doi.org/10.1016/j.procs.2018.10.176

[15] M. AlSabah, A. Tomescu, I. Lebedev, D. Serpanos, and S. Devadas, "PriviPK: Certificate-less and secure email communication," *Computers & Security*, vol. 70, pp. 1–15, Sep 2017. [Online]. Available: https://doi.org/10.1016/j.cose.2017.04.008 https://linkinghub.elsevier.com/retrieve/pii/S0167404817300834

[16] A. Cohen, N. Nissim, and Y. Elovici, "Novel set of general descriptive features for enhanced detection of malicious emails using machine learning methods," *Expert Systems with Applications*, vol. 110, pp. 143–169, Nov 2018. [Online]. Available: https://doi.org/10.1016/j.eswa.2018.05.031 https://linkinghub.elsevier.com/retrieve/pii/S0957417418303312

[17] E. Derouet, "Fighting phishing and securing data with email authentication," *Computer Fraud & Security*, vol. 2016, no. 10, pp. 5–8, Oct 2016. [Online]. Available: http://dx.doi.org/10.1016/S1361-3723(16)30079-3 https://linkinghub.elsevier.com/retrieve/pii/S1361372316300793

[18] A. Levi and C. B. Güder, "Understanding the limitations of S/MIME digital signatures for e-mails: A GUI based approach," *Computers and Security*, vol. 28, no. 3-4, pp. 105–120, 2009.