# Cyber Attack Defense: DID-based Email Security System

Han-Su Oh
*dept. Cyber Security*
*Ajou University*
Suwon, Republic of Korea
ogemini@ajou.ac.kr

*Ki-Hyung Kim
*def. Cyber Security*
*Ajou University*
Suwon, Republic of Korea
kkim86@ajou.ac.kr

***Abstract*—As interest in personal information protection and self-sovereign identity authentication increases, research into the integration potential of blockchain and DID technologies across various industries is growing. Although current email systems continue to be upgraded, their core technology is over 30 years old. These systems, focused more on communication rather than security, lack robust protocols to counter various modern cyberattacks. Addressing these vulnerabilities, this paper researches enhancing digital communication security and focuses on security using DID. The proposed DID-based system, compared to traditional PGP and S/MIME methods, manages identity information more securely and efficiently without relying on centralized authentication authorities. It's designed to ensure the integrity of emails, verify sender identity, and effectively resist spear-phishing and various cyberattacks, as explained through security and theoretical analyses in this DID-based system proposal.**

***Index Terms*—Email Security, Decentralized Identifier, Spear-phishing**

## I. Introduction

As interest in personal information protection and digitalization grows, various technologies are emerging as solutions. In particular, blockchain and Decentralized Identifier (DID) technology are researched for their high security and decentralization benefits in personal information protection and digitalization. Traditional email security systems still struggle with user key management, the centralization of the certification authority, and vulnerabilities to advanced cyberattacks, which blockchain technology research aims to resolve. Existing email security systems rely on digital certificates/digital signatures from the Certification Authority, leading to potential vulnerabilities. The centralized nature of these systems introduces risks of single points of failure and dependence on the Certification Authority's trustworthiness. A notable example is the 2016 WoSign event in China, where the authority issued numerous SHA-1 certificates despite known vulnerabilities. They backdated issuance dates and provided official website domain certificates to ordinary users without adequate verification, enabling impersonation.[1] Furthermore, a central authority error in Korea's administrative computer network in November 2023 caused service disruptions due to authentication issues for government officials. These incidents underline the existing systems' vulnerabilities.

Blockchain, by storing data on a decentralized network, addresses the vulnerabilities of centralized servers. This enhances resistance against various cybersecurity threats and bolsters security. DID, in addition to these technological advantages, allows users complete control over their data and identity. This characteristic led to the proposal of the DID-based email security system in this paper. The system, utilizing DID and VC, strengthens security by integrating email signature and verification processes. Unlike traditional PGP and S/MIME methods, the DID-based email security system benefits from decentralization, managing users' identity information more securely and efficiently. The paper is organized as follows: Chapters 2 and 3 introduce the background of DID-based research and related studies on digital signatures, Chapter 4 explains the system design proposed by the authors, Chapter 5 theoretically analyzes and evaluates the proposed system, and Chapter 6 concludes the paper.

## II. Background knowledge

### A. S/MIME and PGP

Secure Multipurpose Internet Mail Extension(S/MIME) is a standard designed for securing email communication. It ensures integrity, confidentiality, and authentication, providing end-to-end security through digital signatures and encryption. S/MIME uses digital certificates issued by a Certificate Authority(CA), containing user identity information and a public key for self-authentication and message integrity verification. If the message is encrypted using the public key contained in the digital certificate, the recipient can respond with an encrypted message.[2] S/MIME's security strength depends on the CA's security and is supported by email clients like Apple Mail, Microsoft Outlook, and Mozilla Thunderbird. However, the reliance on a single trust authority and challenges in certificate management by users affect its adoption.

Pretty Good Privacy(PGP) grants users full control over key management. Users manage their own public and private keys, exchange them directly, and use these keys to encrypt, decrypt, and sign emails. As it's not issued by a central authority, it relies on mutual trust, but this requires users to manually set up and search for key servers to obtain others' public keys. It also

*Corresponding author:Ki-Hyung Kim

979-8-3503-3094-6/24/$31.00 ©2024 IEEE

demands a basic understanding of asymmetric encryption and PGP technology, leading to usability issues. PGP's adoption has been challenging. a usability study in 1999 showed the necessity of encryption knowledge, and since 2006, there have been concerns regarding users' reliability.[3]

### *B. DID and VC*

Blockchain is a distributed ledger technology where all nodes in the network manage and verify data. It eliminates the drawbacks of centralized systems, maintaining integrity and offering resistance against tampering and denial, as all nodes possess the records.[4] DID, in addition to blockchain's features, offers the advantage of self-sovereign identity. As a unique identifier for individuals or organizations, DID is stored in a decentralized manner and linked to a DID document, which can reference a specific distributed ledger or network and include data and verification methods. Known as Verifiable Credential(VC), this structure contains identity information, exemplified in JSON format. VC, a digital equivalent of credentials like personal ID numbers and passports, enhances privacy and security through machine-verifiable means. It supports selective disclosure in the form of Verifiable Presentation(VP), allowing users to reveal only the desired amount of information.[5]

## III. Related Research

### *A. Blockchain-Based Secure Email Solution*

This research assigns a unique blockchain address to each email to effectively combat spam and phishing attacks, vulnerabilities in traditional email systems. Senders execute a blockchain transaction before sending an email to verify its authenticity. These transactions, recorded on the blockchain network, ensure email integrity due to their transparency and immutability. In the research, the proposed system automatically filters out emails that have not undergone a blockchain transaction, thereby providing resistance to various cyberattacks. This method ensures that only emails verified through blockchain transactions are processed, enhancing the security and reliability of email communications.[6] This approach guarantees the delivery of legitimate emails, ensuring safe and trustworthy communication.

### *B. Blockchain-Based Smart Contract Email Protocol*

Research is exploring the use of blockchain technology and smart contracts to fundamentally improve vulnerabilities in the traditional SMTP email protocol. Existing email systems, focused more on communication, lack robust authentication mechanisms, leaving them vulnerable to phishing, spoofing attacks, and spam. This research proposes using Ethereum-based smart contracts to replace the email protocol, leveraging the blockchain's distributed ledger advantages. It ensures the integrity and non-repudiation of all email messages by using blockchain technology.[7] However, this approach, a complete replacement rather than an integration, poses challenges for practical application and widespread adoption, and concerns about performance and scalability in an email system that handles large volumes of data frequently.

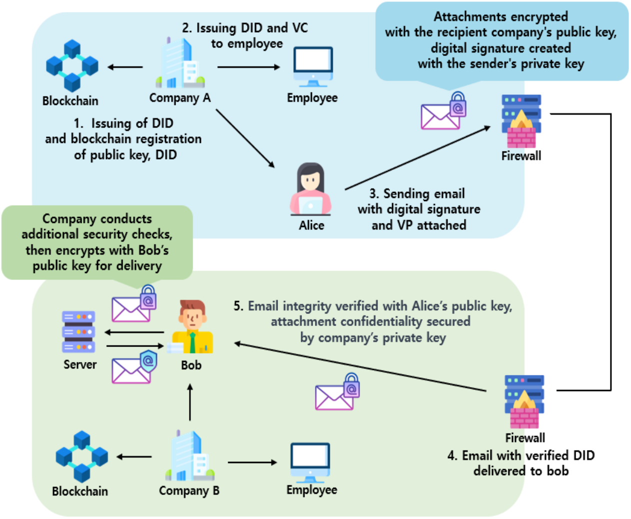


Fig. 1. Full system scenario

## IV. Proposed Content

### *A. Overall System Scenario*

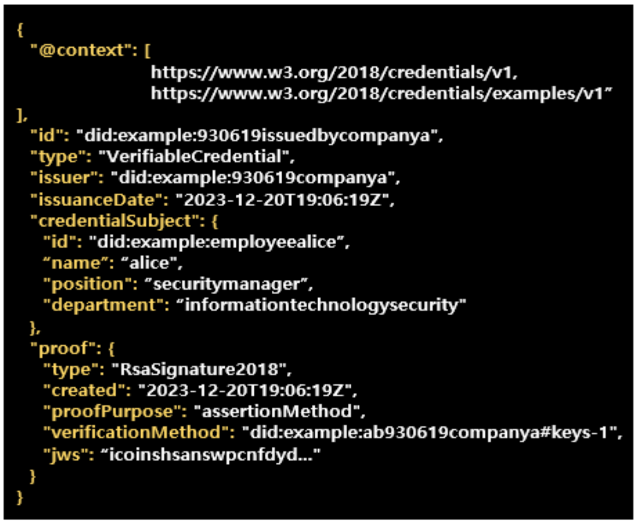
The overall system scenario is depicted in fig.1. The first step involves Company A creating a DID and a public-private key pair, registering the public key with the DID on the blockchain, allowing public verification of the company's identity.

The second step, Company A issues employees their own public-private key pairs and DID, which are registered on the blockchain, making them unique identifiers. Additional information, such as affiliation, identity, and email address, is added to the DID and recorded in the VC, signed with Company A's private key for verification by recipients.

The third step, Alice, an employee, composes an email encrypted with her private key for integrity, and any attachments are encrypted with the recipient, Bob's company public key, for confidentiality. Alice doesn't need Bob's company digital certificate or public key in advance; if Bob's company DID is known or previously interacted with, it can be used to encrypt attachments.

The fourth step, when the email is sent, the recipient's firewall checks the validity of the DID. Company B's firewall verifies the DID in the VP, referencing the verificationMethod field to validate the signature in the jws field and confirm Alice's affiliation. In this process, public keys are sourced from internal storage, nodes, or the blockchain, eliminating the need for prior exchange or possession as in traditional systems.

Finally, The digital signature of the received email is verified using Alice's public key to ensure its integrity. The attachment is transmitted to Company B's server, where it is decrypted using Company B's private key to ensure its confidentiality. Then, the attachment is encrypted using Bob's public key and delivered to him, concluding the process. Sending the attachment to the company's server allows for more granular access control according to company policies, and server-



```
{
  "@context": [
                    https://www.w3.org/2018/credentials/v1,
                    https://www.w3.org/2018/credentials/examples/v1"
],
  "id": "did:example:930619issuedbycompanya",
  "type": "VerifiableCredential",
  "issuer": "did:example:930619companya",
  "issuanceDate": "2023-12-20T19:06:19Z",
  "credentialSubject": {
    "id": "did:example:employeealice",
    "name": "alice",
    "position": "securitymanager",
    "department": "informationtechnologysecurity"
  },
  "proof": {
    "type": "RsaSignature2018",
    "created": "2023-12-20T19:06:19Z",
    "proofPurpose": "assertionMethod",
    "verificationMethod": "did:example:ab930619companya#keys-1",
    "jws": "icoinshsanswpcnfdyd..."
  }
}
```

Fig. 2. Example of VC

based processing offers an additional security layer, increasing sensitivity to attacks. Furthermore, in environments where compliance is key, this data flow is beneficial for regulatory adherence and auditability.

### B. Detailed System Design

Fig. 2. in the detailed system design shows an example of the VC issued by Company A to its employees in the second step of the overall system scenario. The issued VC follows the format shown in Fig. 2. The @context defines the VC's data model with a URL, and the id signifies its unique identifier. The issuer is Company A's DID, who issues the VC. The credentialSubject contains the employee's information. The proof field holds the verificationMethod and jws fields, confirming the issuer, Company A, of the VC.

Fig.3. details the process of Alice composing and sending an email. Alice encrypts the email attachment using Company B's public key, ensuring its confidentiality as it can only be decrypted with Company B's private key. The digital signature is created using the sender's and recipient's email addresses and the email body, encrypted with Alice's private key, ensuring integrity. Alice then attaches her VP to the email before sending it.

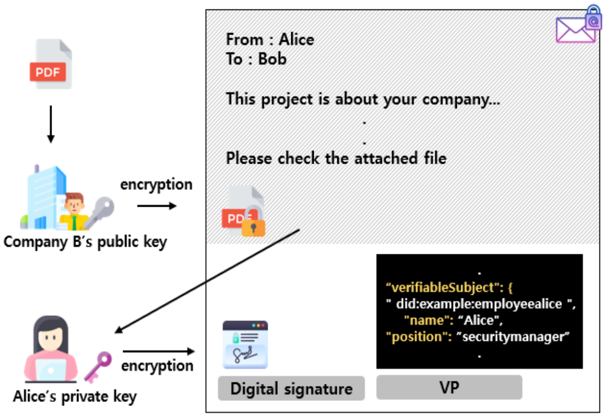


Fig. 3. email content

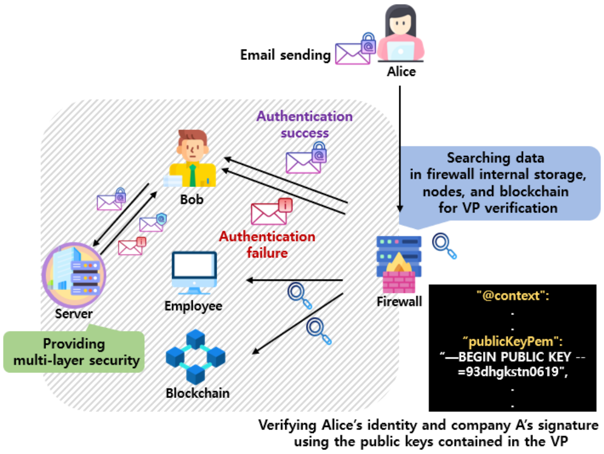
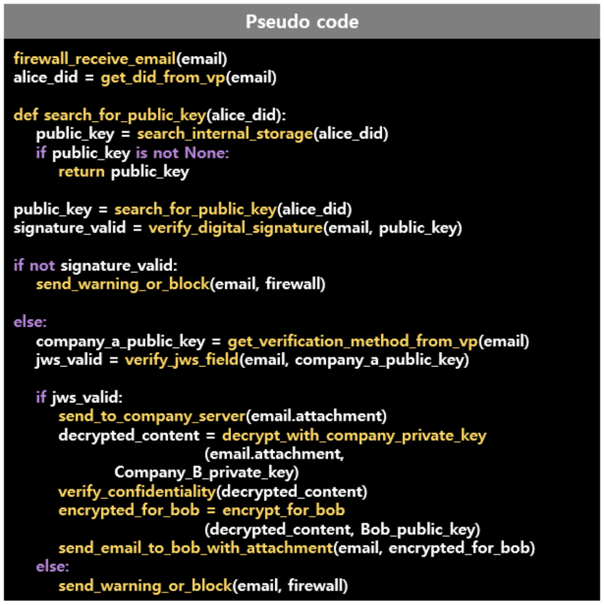

Email sending
Alice
Authentication success
Bob
Authentication failure
Searching data in firewall internal storage, nodes, and blockchain for VP verification
Server
Employee
Firewall
"@context":
"publicKeyPem":
"—BEGIN PUBLIC KEY --
=93dhgkstn0619",
Providing multi-layer security
Blockchain
Verifying Alice's identity and company A's signature using the public keys contained in the VP

Fig. 4. Receiving firewall inspection

Fig. 4. illustrates the process where an email reaches the recipient's domain as part of the overall system scenario. The email includes a VP with DID and publicKeyPem fields. The firewall searches its internal storage, nodes, and the blockchain using this information. If Alice's DID is valid, the publicKeyPem field's public key is used to verify the digital signature and confirm its integrity.If the digital signature is valid, the firewall verifies the jws field to confirm Alice's affiliation. Should the DID or signature be invalid, the firewall either blocks the email or forwards it to the recipient with a warning. In cases where an email is blocked, the recipient is notified of the blockage, enabling them to address the issue through a blocking review process.The attachments are sent to the company's server, which provides multiple security layers and increases sensitivity to attacks within the content. The company decrypts the attachments using its private key to check confidentiality. If the confidentiality is intact, the attachments are then encrypted with Bob's public key, ensuring that only authorized recipient Bob can decrypt them.

Fig. 5. represents the receiving process in pseudocode to facilitate understanding. The process begins with the firewall receiving the email through 'firewallreceiveemail'. 'getdidfromvp' is used to extract Alice's DID from the email's VP. Although partially omitted, 'searchforpublickey' searches for Alice's public key in internal storage, nodes, and the blockchain. If a public key is found, it is returned.The process continues with 'verifydigitalsignature', which checks the email's digital signature using the public key obtained earlier. If the signature is invalid, a warning is sent, or the email is blocked. If the digital signature is valid, 'getverificationmethodfromvp' extracts Company A's public key from the email's VP, verifying the jws field. If the jws field is verified, the email is delivered to Bob and the attachment is sent to the company server. If not, it is either blocked or sent with a warning.Through 'decryptwithcompanyprivatekey', the



```
Pseudo code

firewall_receive_email(email)
alice_did = get_did_from_vp(email)

def search_for_public_key(alice_did):
    public_key = search_internal_storage(alice_did)
    if public_key is not None:
        return public_key

public_key = search_for_public_key(alice_did)
signature_valid = verify_digital_signature(email, public_key)

if not signature_valid:
    send_warning_or_block(email, firewall)

else:
    company_a_public_key = get_verification_method_from_vp(email)
    jws_valid = verify_jws_field(email, company_a_public_key)

    if jws_valid:
        send_to_company_server(email.attachment)
        decrypted_content = decrypt_with_company_private_key
                            (email.attachment,
              Company_B_private_key)
        verify_confidentiality(decrypted_content)
        encrypted_for_bob = encrypt_for_bob
                            (decrypted_content, Bob_public_key)
        send_email_to_bob_with_attachment(email, encrypted_for_bob)
    else:
        send_warning_or_block(email, firewall)
```

Fig. 5. process pseudo code when receive

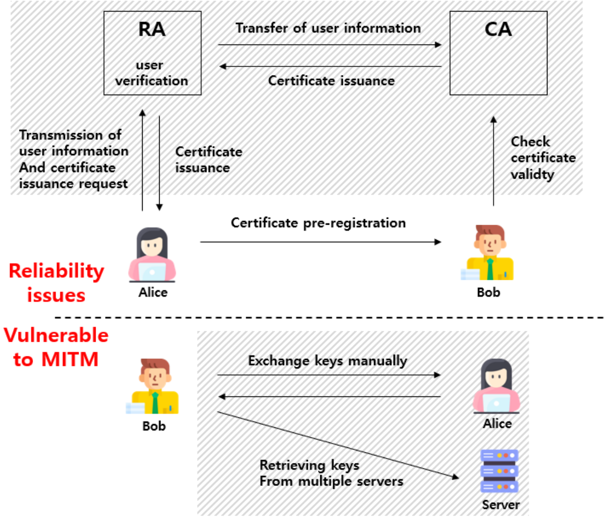

RA
user verification
Transfer of user information
Certificate issuance
CA
Transmission of user information And certificate issuance request
Certificate issuance
Check certificate validty
Certificate pre-registration
Reliability issues
Alice
Bob
Vulnerable to MITM
Exchange keys manually
Bob
Alice
Retrieving keys From multiple servers
Server

Fig. 6. structural changes

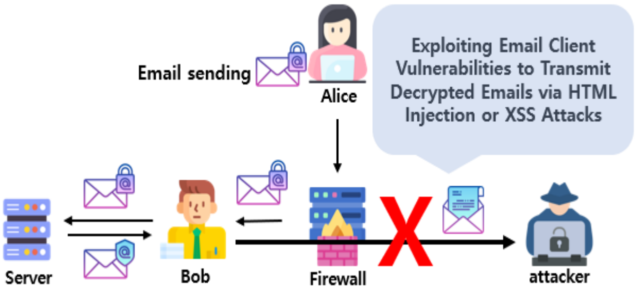

Email sending
Alice
Exploiting Email Client Vulnerabilities to Transmit Decrypted Emails via HTML Injection or XSS Attacks
Server
Bob
Firewall
attacker

Fig. 7. Increased resistance to HTML injection or XSS attack

attachment is decrypted using Company B's private key to confirm its confidentiality. Once confidentiality is verified, 'encryptforbob' encrypts the attachment with Bob's public key, after which it is delivered to Bob. Bob then uses his private key to decrypt the file, ensuring its safety, confidentiality, and integrity, as it has been scrutinized through multiple security layers.

## V. System Analysis

### A. Structural Changes in Existing Systems

S/MIME and PGP are standards designed for email security enhancement, but S/MIME has numerous issues due to its centralized structure. These vulnerabilities are addressed by applying blockchain, as shown in Fig. 6., where underlined parts indicate areas replaced by the DID-based system. This eliminates the need for a centralized CA and the cumbersome process of certificate pre-exchange, moving away from dependence on CA. Improved security is achieved as users manage their own certificates less and use DID to obtain public keys directly from the blockchain. This reduces the need for certificate registration and verification, enhancing usability and cyberattack resistance due to blockchain's proven integrity.

In the DID-based system, unlike PGP's key exchange method, users don't need to exchange keys, enhancing usability. Integrity is verified through blockchain during the exchange process, and identity is confirmed using DID, thus increasing resistance to MITM attacks. This shift to DID-based systems offers a more secure, efficient, and user-friendly approach to key management and identity verification.

### B. Providing multiple security layers

HTML Injection and XSS attack occur when attackers execute malicious scripts during the decryption process of encrypted emails[8], leading to the transmission of user information or decrypted content to the attacker. The proposed system decrypts attachments at the server and re-encrypts them with Bob's public key, making it difficult for attackers to intercept decrypted files. By decrypting at the server rather than at individual user systems, additional security measures can be applied before decrypted data reaches the user's device. This approach defends against side attacks exploiting local system vulnerabilities and enhances the overall security of the email system with multiple security layers at the server in the DID-based system.

### C. typo squating attacks and redirect

Spear-phishing attacks often gain user trust to obtain consent, typically causing confusion through tactics like typo squating. In the past, similar alphabets or numbers (like using '1' for 'l') were used, but recently attackers create visually identical addresses using Cyrillic characters[9]. When an attacker impersonates an identity or intercepts and resends a message with identity, it can be challenging for users to distinguish, even if the message carries a legitimate digital signature. However, the proposed DID-based system automatically blocks or warns against emails with visually

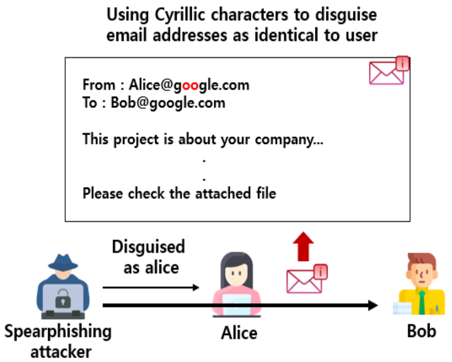

Using Cyrillic characters to disguise email addresses as identical to user
From : Alice@google.com
To : Bob@google.com
This project is about your company...
Please check the attached file
Disguised as alice
Spearphishing attacker
Alice
Bob

Fig. 8. typo squating attacks and redirect

identical but differently formatted addresses, using the verifiableSubject field's email address. Additionally, utilizing DID for site link verification in emails can enhance the security of web browsers, not just email systems.

## VI. Conclusion

This paper proposes enhancing existing email security systems using blockchain and DID technology, addressing the centralization and key management challenges that hinder widespread adoption of current email security protocols. The research focuses on decentralization and user convenience through a DID-based system, responding to sophisticated and evolving cyberattacks. This research aspires to increase resistance against attacks like injection and XSS, and to contribute to defending against spear-phishing, where human security awareness is crucial.With many infrastructures being developed and researched using blockchain and DID technologies, the integration of the proposed DID-based system could establish stronger security within the same infrastructure. Future developments and testing of this research, particularly integrating with existing protocols and infrastructures, will significantly advance this field of research.

## Acknowledgment

This research was supported by the Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0008703, HRD Program for Industrial Innovation), the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (2021R1F1A1045861), and the Institute of Information and communications Technology Planning and Evaluation (IITP) grant funded by the Korea government (MSIT) (2021-0-00590, Decentralized High Performance Consensus for Large-Scale Blockchains).

## References

[1] B. Jayaraman, H. Li and D. Evans, ”Decentralized certificate authorities”, arXiv:1706.03370, 2017.

[2] A. H. Al-Ghushami et al., ”Email Security: Concept, Formulation, and Applications,” 2022 14th International Conference on Computational Intelligence and Communication Networks (CICN), Al-Khobar, Saudi Arabia, 2022, pp. 825-829.

[3] Adrian Reuter, Ahmed Abdelmaksoud, Karima Boudaoud and Marco Winckler, ”Usability of End-to-End Encryption in E-Mail Communication”, Frontiers in Big Data, vol. 4, 2021.

[4] D. Yaga, P. Mell, N. Roby, and K. Scarfone, ”Blockchain technology overview”, arXiv:1906.11078, 2019.

[5] C. Brunner, U. Gallersdörfer, F. Knirsch, D. Engel, and F. Matthes, ”DID and VC: Untangling decentralized identifiers and verifiable credentials for the web of trust”, in Proc. 3rd Int. Conf. Blockchain Technol. Appl., Dec. 2020, pp. 61–66.

[6] Stewart MacGregor Dennis, ”Blockchain based email procedures”, unpublished.

[7] J. Chamadoira Gonzalez, V. Garcia-Diaz, ´ et al., “Replacing email protocols with blockchain-based smart contracts,” 2020.

[8] Agarwala, Devashish and devashishagarwala, “SQL injection and XSS”, unpublished

[9] Lee Joon Sern and Yam Gui Peng David, ”Typoswype: An imaging approach to detect typo-squatting” In 2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS), 2021, pp. 1–5.