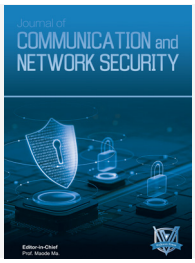
Journal of Communication and Network Security (JCNS)

DOI:http://doi.org/10.65098/jcns.01.2026.09.11



*RESEARCH ARTICLE*

# VULNERABILITY MINING AND REPAIR STRATEGIES FOR NETWORK SECURITY PROTOCOLS IN DIGITAL COMMUNICATION

**Weizhou Li**

*China Academy of Information and Communications Technology, Beijing 100073, China*
**Corresponding Author E-mail: Esther_Editor@volksonpress.com*



**ARTICLE DETAILS**

*Article History:*

Received 08 Nov 2025
Accepted 04 Jan 2026
Available online 9 Feb 2026

Online Article Code

**ABSTRAST**

**Background and Purpose**: With the rapid development of digital communication technologies, network security protocols are increasingly exposed to various types of vulnerabilities. Effectively identifying and addressing these vulnerabilities has become critical to ensuring communication security and protecting user data. This study aims to analyze common vulnerabilities in digital communication and network security protocols and to emphasize the importance of vulnerability mining in digital communication security.
**Methods**: This paper outlines digital communication and network security protocols and their common vulnerability types, and designs an efficient vulnerability mining process. Based on the identified vulnerabilities, targeted repair technologies and methods are proposed, and an efficient vulnerability repair and management system is constructed.
**Results**: The proposed vulnerability mining process and repair management system improve the efficiency and accuracy of vulnerability identification and remediation. The results demonstrate that the designed approach enhances system security and strengthens the protection of user data.
**Conclusion**: The study provides a comprehensive and practical strategy for improving the security of digital communication systems. By integrating vulnerability mining with targeted repair and systematic management, the proposed approach contributes to more robust network security and reliable digital communication environments.

**KEYWORDS**

Digital Communication, Network Security Protocol, Vulnerability Mining, Vulnerability Repair, Management System

## 1. INTRODUCTION

With the rapid development of digital communication technology, network security issues have become increasingly prominent. As an important guarantee for digital communication security, the integrity and reliability of network security protocols are directly related to the stable operation of communication systems and the security of user data. As attack methods continue to evolve, a series of vulnerabilities have also been exposed in network security protocols. If exploited by malicious actors for network attacks or data theft, these pose a serious threat to digital communication security.

### 1.1 Overview of Digital Communication Protocols and Network Security Protocols

In the vast field of digital communication, various communication protocols form the cornerstone of information transmission. Among them, key digital communication protocols such as TCP/IP and HTTP/HTTPS not only support the operation of the Internet but also directly affect the efficiency and security of communication. TCP/IP (Transmission Control Protocol/Internet Protocol), as the core protocol of the Internet, is responsible for the segmentation, transmission, and reassembly of data, ensuring reliable information delivery worldwide. HTTP/HTTPS (Hypertext Transfer Protocol/Secure Hypertext Transfer Protocol) dominates web communication, with HTTPS providing encryption protection for data transmission by incorporating the SSL/TLS (Secure Sockets Layer/Transport Layer Security) protocol, effectively preventing information from being eavesdropped on and tampered with during transmission.

Network security protocols are an important safeguard for digital communication security. Among them, the SSL/TLS protocol ensures secure communication between parties through multiple protection mechanisms such as authentication, data encryption, and integrity verification. It not only prevents data from being stolen or tampered with during transmission but also ensures the authenticity of the identities of the communicating parties, building a solid line of defense. IPSec (Internet Protocol Security) is primarily employed to secure IP packets. By utilizing encryption and authentication mechanisms, it ensures the confidentiality, integrity, and authenticity of data in transit.

### 1.2 Common Types of Vulnerabilities in Network Security Protocols

There are three main common types of vulnerabilities in network security protocols: First, insufficient encryption strength. Encryption is one of the core mechanisms in network security protocols, protecting the confidentiality of data during transmission. When the strength of the encryption algorithm is insufficient, it can lead to attackers stealing or tampering with data through brute force cracking or exploiting the weaknesses of the algorithm itself. Second, authentication mechanism defects. The authentication mechanism is a key link in ensuring the authenticity of the identities of the communicating parties. If the authentication mechanism has defects, such as a non-rigorous authentication process, or authentication information being easily forged or tampered with, it can lead to identity impersonation attacks. Attackers exploit these defects to impersonate legitimate communication parties to obtain sensitive information or perform malicious operations, which not only undermines the reliability of communication but also easily triggers a crisis of trust, seriously affecting the overall security of digital communication. Third, implementation errors. During the implementation process of network security protocols, due to programming errors, improper configuration, or misunderstanding, the actual behavior of the protocol deviates from expectations, resulting in security vulnerabilities. These manifest in various forms such as buffer overflows, format string vulnerabilities, and logic errors. For example, when processing user input data, if sufficient validation and filtering are not performed, leading to the injection and execution of malicious code, it provides an opportunity for attackers to bypass the security mechanisms of the protocol and attack the communication system.

## 2. VULNERABILITY MINING STRATEGIES FOR DIGITAL COMMUNICATION NETWORK SECURITY PROTOCOLS

### 2.1 The Key Role of Vulnerability Mining in Digital Communication Security

In the complex ecosystem of digital communication, vulnerability mining, as a forward defense line, is not only related to the immediate security of communication systems but also a key driver for the sustained and healthy development of the entire digital communication field. Vulnerability mining plays the core role of "early warning to reduce security risks." In the network space of digital communication, potential vulnerabilities are like hidden undercurrents that can trigger security crises at any time. Through systematic vulnerability mining work, these hidden risk points can be discovered in a timely manner, providing valuable remediation time windows for security teams. This can effectively reduce the probability of security incidents and protect user data from threats of unauthorized access and tampering (Chen et al., 2014). Vulnerability mining is an important promoter for "promoting the continuous optimization of protocols and systems." The development of digital communication technology is changing rapidly, with new protocols and systems constantly emerging, and the application of each new technology is accompanied by new security challenges. Through continuous vulnerability mining, defects and shortcomings in the design of protocols or systems can be revealed, providing valuable feedback and directions for improvement to developers. This iterative optimization process can not only enhance the overall security of communication systems but also drive the continuous progress and refinement of technology.

### 2.2 Vulnerability Mining Techniques for Digital Communication Protocols

Static analysis, as the cornerstone of vulnerability mining, can discover potential logic errors and security vulnerabilities without executing the program through in-depth analysis of code and configuration. By meticulously examining source code, binary files, code snippets, or configuration errors that may cause security problems can be identified. For example, in the implementation of digital communication protocols, static analysis can detect common vulnerabilities such as the use of uninitialized variables and buffer overflows. However, static analysis also has its limitations; it cannot capture vulnerabilities caused by the dynamic behavior of the program during runtime. In the field of dynamic testing, fuzz testing and symbolic execution are two particularly effective techniques. Fuzz testing attempts to trigger abnormal behavior or crashes in a program by inputting a large amount of random or semi-random data into it. It is especially suitable for discovering vulnerabilities when processing input data, such as format string vulnerabilities and SQL injection. In digital communication protocols, fuzz testing can be used to test the robustness of key components such as protocol parsers and encoders. Symbolic execution is a more refined testing technique. By representing program inputs and states in a symbolic way and tracking paths and conditions during program execution, it can precisely analyze program behavior under different inputs and discover vulnerabilities hidden in deep logic. In digital communication protocols, symbolic execution can be used to verify the correctness of protocol state machines and discover potential logic vulnerabilities and state inconsistency issues.

### 2.3 Design of an Efficient Vulnerability Mining Process

In the field of digital communication network security, efficient and precise vulnerability mining is a key link in ensuring system security. The starting point of the vulnerability mining process is to clarify the mining objectives. Based on factors such as risk assessment, business importance, and potential impact, specific digital communication protocols, software systems, or their key components are identified as mining targets. The accuracy of this step directly affects the efficiency and effectiveness of subsequent work. Entering the information collection stage. This stage requires comprehensively collecting relevant information about the target system, such as documentation, source code (if available), historical vulnerability records, security bulletins, and user feedback. Through in-depth analysis of these materials, a comprehensive understanding of the target system can be constructed, laying a solid foundation for subsequent vulnerability identification (Qiao, 2023). Entering the core link of vulnerability identification. At this stage, various technical means, such as static analysis, dynamic testing, fuzz testing, and symbolic execution, are used to conduct in-depth probing of the target system. Combining the information collected in the early stages, targeted test cases are designed and executed to reveal potential vulnerabilities. For example, by sending abnormal data packets to the parser of a communication protocol through fuzz testing and observing its abnormal behavior during the processing, potential vulnerability points can be identified. After vulnerabilities are identified, they need to be verified and reported. The verification process includes reproducing the vulnerability, analyzing its impact and exploitability to ensure its authenticity and severity. A detailed vulnerability report is written, including vulnerability description, scope of impact, reproduction steps, repair suggestions, and the CVE number of the vulnerability (if assigned), which serves as an important basis for communicating with developers and promoting vulnerability repair.

## 3. VULNERABILITY REPAIR STRATEGIES FOR DIGITAL COMMUNICATION NETWORK SECURITY PROTOCOLS

### 3.1 Basic Principles of Vulnerability Repair

In the field of digital communication network security, the existence of vulnerabilities is undoubtedly a serious threat to system security. Vulnerability repair work is not only related to the stable operation of the system but also directly related to the security and privacy protection of user data. When performing vulnerability repairs, a series of basic principles must be followed. Rapid response is the primary principle of vulnerability repair. Once a vulnerability is discovered, the repair process must be initiated immediately to patch the vulnerability as quickly as possible and minimize the duration of the security threat. Vulnerabilities are often exploited by malicious actors for network attacks or data theft. Rapid response is not only responsible for users but also a strong guarantee for system security (Liu, 2021). Ensuring the compatibility and stability of the repair process is also crucial. During the process of repairing vulnerabilities, the overall operating environment of the system and the compatibility of various applications must be considered.

### 3.2 Targeted Vulnerability Repair Technologies and Methods

There are four main types of vulnerability repair technologies and

methods for digital communication network security protocols: First, encryption algorithm upgrades. Encryption algorithms are the core technology for ensuring data security. With the continuous improvement of computing power, traditional encryption algorithms are gradually becoming less secure. By regularly upgrading encryption algorithms, including adopting longer key lengths, more complex encryption logic, and more advanced encryption standards—for example, upgrading from the DES algorithm to the AES algorithm, from SHA-1 to SHA-256, etc.—data security can be improved, preventing encryption algorithms from being maliciously cracked. Second, strengthening authentication mechanisms. The authentication mechanism is key to ensuring the legitimacy of user identities. In digital communication network security protocols, strengthening the authentication mechanism can prevent unauthorized users from accessing the system and effectively avoid potential security risks. This includes measures such as multi-factor authentication, biometric identification, and strengthening password policies. Multi-factor authentication effectively improves authentication security by combining multiple authentication methods, such as passwords, mobile verification codes, and fingerprint recognition. Biometric identification authenticates users based on their biological characteristics (such as fingerprints, facial features, etc.), which are unique and non-replicable. Strengthening password policies requires users to set complex and difficult-to-guess passwords and change them regularly (Wang et al., 2024). Third, code auditing and refactoring. Code auditing is the process of comprehensively inspecting the source code of a system to discover potential vulnerabilities and security risks. Through code auditing, logic errors, insecure function calls, unhandled exceptions, and other situations in the code can be discovered and repaired promptly. During the code auditing process, by using professional auditing tools and techniques to analyze the code line by line, combined with vulnerability scanning and penetration testing, a comprehensive assessment of the system's security is conducted. Once vulnerabilities or security risks are discovered, they are immediately repaired, and the effectiveness of the repairs is verified. Refactoring is the process of optimizing and restructuring existing code to improve code quality and maintainability. It can eliminate duplicate and redundant parts of the code, optimize the code structure, improve code execution efficiency, replace insecure code with more secure implementation methods, and enhance the overall security of the system. Fourth, patch development and testing. When vulnerabilities or security risks are discovered, developing patches is an effective means of quickly fixing problems. Patch development needs to be customized based on specific vulnerabilities or security risks to ensure that the problem can be accurately patched without affecting other functions of the system. During the patch development process, the vulnerabilities or security risks are analyzed in depth to clarify the cause of the problem and its scope of impact. Based on the analysis results, a patch solution is designed and coded for implementation. The patch undergoes comprehensive testing to ensure its functions are correct, its performance is stable, and it does not cause new security problems.

### 3.3 Building an Efficient Vulnerability Repair Management System

The construction of a vulnerability repair management system for digital communication network security protocols mainly starts from two aspects: First, standardization and automation of the repair process. Standardizing the repair process is the foundation for building an efficient vulnerability repair management system. By formulating detailed repair process specifications and clarifying tasks, responsible persons, and time nodes at each stage, vulnerability repair work can be carried out in an orderly manner, avoiding repair delays or omissions due to process confusion. This covers the entire process from vulnerability discovery, reporting, analysis, repair, to verification, ensuring that each link has clear operating guidelines and quality control standards. On the basis of standardized processes, achieving automation of the repair process can improve efficiency. By introducing automated tools and technologies, such as vulnerability scanners and automated repair scripts, tasks such as vulnerability identification, classification, repair, and verification can be completed automatically, reducing manual intervention and increasing repair speed. Second, tracking and evaluating repair effectiveness. During the repair process, repair progress and repair operations can be tracked in real time through methods such as log recording and status monitoring to ensure the repair work proceeds smoothly according to the established process. After the repair is completed, comprehensive system testing should be conducted, including functional testing, performance testing, and security testing, to verify whether the repair was successful and whether it introduced new problems or hidden dangers. Establish a vulnerability recurrence monitoring mechanism to regularly review repaired vulnerabilities to ensure they do not reappear or are exploited by new attack methods. Reviews can be achieved through regular vulnerability scanning, penetration testing, and user feedback collection (He & Ge, 2024).

## 4. CONCLUSION

Through in-depth research on vulnerability mining and repair strategies for network security protocols in digital communication, this paper not only reveals common types of vulnerabilities and their harms in network security protocols but also proposes a comprehensive and effective set of vulnerability mining and repair methods. By combining vulnerability mining techniques of static analysis and dynamic testing, along with targeted vulnerability repair technologies and methods, vulnerabilities in the system can be discovered and patched in a timely manner to enhance the overall security of the system. Constructing an efficient vulnerability repair management system ensures the orderly progress and continuous improvement of vulnerability repair work, providing a strong guarantee for digital communication network security.

## REFERENCES

Chen, J., Wu, Q., & Wang, Z. (2024). Analysis of the application of network security protocols in computer communication technology. *Information and Computer (Theory Edition)*, *36*(1), 171–173.

He, Q., & Chen, T. (2024). Research on university network security and prevention based on AI environment. *High-Technology and Industrialization*, *30*(11), 31–33.

Liu, B. (2021). Analysis of the role of network security protocols in computer communication technology. *Information Recording Materials*, *22*(11), 77–78.

Qiao, X. (2023). Application of network security protocols and computer communication systems in cloud computing environments. *Automation Application*, *64*(14), 228–230, 233.

Wang, W., Chen, K., He, X., et al. (2024). Countermeasure defense technology for network security attack tools based on Go language. *Network Security Technology & Application*, (12), 27–28.