Street Art & Urban Creativity | Vol. 12, No. 5s, 2026 | ISSN 2183-9956
Street Art & Urban Creativity. Scientific Journal

A Blockchain-Inspired Digital Evidence Management System Using SHA-256 Hashing

VAISHNAVI KANDARAM[1], (kandaramvaishnavi@gmail.com)[1]
DR. GNV VIBHAV REDDY[2] , (VICECHAIRMAN@SREEDATTHA.AC.IN)[2]
[1]M TECH STUDENT, DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, SREE DATTHA INSTITUTE OF ENGINEERING AND SCIENCES, SHERIGUDA, HYDERABAD, INDIA
[2]ASSOCIATE PROFESSOR, DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING, SREE DATTHA INSTITUTE OF ENGINEERING AND SCIENCES, SHERIGUDA, HYDERABAD, INDIA

Abstract

Digital evidence plays a crucial role in modern forensic investigations and legal proceedings, making its secure storage, integrity verification, and traceability essential. Traditional evidence management systems often rely on centralized storage, which is vulnerable to unauthorized modifications, accidental data loss, and difficulties in maintaining a reliable chain of custody. This paper presents a Digital Evidence Management System (DEMS) developed using the Django framework to provide secure evidence handling through cryptographic integrity verification. The proposed system enables users to register, upload digital evidence, and verify file authenticity using the SHA-256 cryptographic hashing algorithm. Each uploaded file is assigned a unique hash value that serves as its digital fingerprint and is stored along with metadata such as the uploader's information, filename, upload timestamp, and hash key. An administrator manages user accounts through an approval mechanism, ensuring system. The application also incorporates secure email-based OTP authentication for password recovery and provides an administrative dashboard for monitoring evidence records. During verification, the uploaded file's hash is regenerated and compared with the stored hash to detect any modifications or tampering. Experimental evaluation demonstrates that the proposed system provides reliable evidence integrity verification, secure access control, efficient evidence tracking, and improved accountability while maintaining a lightweight architecture that is practical for forensic laboratories, law enforcement agencies, and judicial organizations. The proposed approach offers a cost-effective and scalable solution for secure digital evidence management without the complexity of a full blockchain infrastructure.

Keywords: Digital Evidence Management, SHA-256, Evidence Verification, Tamper Detection, Digital Forensics, Django, Chain of Custody, Cryptographic Hashing.

1. Introduction

The rapid advancement of information technology and the widespread adoption of digital devices have transformed the manner in which information is created, transmitted, and stored. As a result, digital data has become an essential source of evidence in criminal investigations, cybercrime analysis, corporate security audits, and judicial proceedings. Digital evidence may include documents, images, videos, emails, system logs, mobile data, and various other electronic records collected from computers, smartphones, cloud platforms, and storage devices. The reliability of such evidence plays a crucial role in determining the outcome of investigations and legal cases. Therefore, maintaining the integrity, authenticity, and confidentiality of digital evidence has become one of the primary concerns for forensic investigators and law enforcement agencies.

Traditional evidence management systems generally rely on centralized storage and manual documentation for maintaining evidence records. Although these methods are widely used, they suffer from several limitations, including vulnerability to unauthorized access, accidental deletion, evidence tampering, incomplete audit trails, and human errors during evidence handling. Maintaining a proper chain of custody is another significant challenge, as every interaction with the evidence must be documented accurately to preserve its legal admissibility. Any modification, intentional or accidental, may compromise the credibility of the evidence and affect the outcome of judicial proceedings. Consequently, there is an increasing need for secure, automated, and reliable digital evidence management systems

capable of preserving evidence integrity throughout its lifecycle.

Cryptographic techniques have emerged as one of the most effective approaches for protecting digital information against unauthorized modification. Among these techniques, the Secure Hash Algorithm-256 (SHA-256) has gained widespread acceptance due to its ability to generate a unique and irreversible hash value for every digital .produces an entirely different hash value, making SHA-256 highly effective for detecting evidence tampering. Because of these characteristics, SHA-256 has become a standard integrity verification mechanism in cybersecurity, digital forensics, and secure information systems.

In recent years, researchers have proposed blockchain-based evidence management frameworks to strengthen the integrity and traceability of digital evidence. Blockchain provides an immutable ledger that records every transaction, making unauthorized alterations extremely difficult. However, implementing a complete blockchain infrastructure often requires distributed nodes, consensus mechanisms, smart contracts, and significant computational resources, increasing both complexity and deployment costs. For many educational institutions, forensic laboratories, and small- to medium-scale organizations, such infrastructure may not be practical. Therefore, lightweight cryptographic approaches that provide similar integrity verification capabilities without the overhead of blockchain have become increasingly attractive.

To address these problems, this paper proposes a Digital Evidence Management System (DEMS) built with the Django web framework and Python programming language. The proposed approach combines secure user identification, administrator permission, cryptographic hashing, evidence storage, metadata management, and evidence verification into a single online application. Each uploaded evidence file is processed using the SHA-256 algorithm, producing a unique digital fingerprint that is securely stored together with important metadata including the filename, uploader information, upload timestamp, and hash value. During verification, the uploaded file is hashed again, and the resulting hash is compared to the stored value to determine whether the evidence is still legitimate or has been tampered with.

The system also incorporates evidence. New users must register and obtain administrator approval before using the application. In addition, an email-based One-Time Password (OTP) mechanism is implemented to provide secure password recovery, while session management maintains secure user authentication throughout each login session. An administrative dashboard enables centralized monitoring of registered users and uploaded evidence, thereby improving accountability and simplifying evidence management.

The proposed Digital Evidence Management System offers several advantages over conventional evidence handling methods. By integrating cryptographic hashing with secure authentication and metadata tracking, the system provides reliable evidence verification, transparent audit trails, and efficient tamper detection while maintaining a lightweight architecture. Unlike blockchain-based solutions that require complex deployment environments, the proposed approach can be easily implemented using standard web technologies, making it organizations, and other sectors that require secure digital evidence management.

Detects modified files, preserves evidence integrity, and provides a practical solution for maintaining a trustworthy chain of custody throughout the evidence management process.

2. LITERATURE SURVEY

The rapid increase in cybercrime and the growing dependence on digital devices have Consequently, numerous researchers have focused on developing secure evidence management systems capable of preserving evidence integrity, ensuring authenticity, and maintaining an uninterrupted chain of custody. Various approaches based on digital forensics, cryptographic hashing, access control mechanisms, blockchain technology, and secure web applications have been proposed to improve the management of digital evidence.

Pollitt [1] introduced one of the earliest concepts of computer forensics by emphasizing the importance of preserving digital evidence throughout the investigative process. His work established the fundamental principles of evidence highlighting that the credibility of digital evidence depends on maintaining its original state during investigations. Casey [2] further expanded these concepts by describing comprehensive forensic methodologies for collecting, analyzing, and presenting author demonstrated that secure documentation and standardized forensic procedures significantly improve.

These guidelines emphasize secure evidence acquisition, preservation, documentation, and verification while recommending systematic procedures for maintaining evidence integrity. The Secure Hash Standard published by NIST [4] introduced SHA-256 as a reliable cryptographic hash function capable of generating unique digital fingerprints for electronic data. Because SHA-256 produces fixed-length hash values that change completely even for minor modifications to the input data, it has become one of the most widely adopted algorithms for integrity verification.

Stallings [5] explained the theoretical foundations of cryptographic hash functions and highlighted their applications in data integrity, digital signatures, authentication, and secure communication systems. Similarly, Schneier [6] discussed the security properties of modern cryptographic algorithms and demonstrated that SHA-256 provides strong resistance against collision attacks and unauthorized modifications. Carrier [7] further emphasized the importance of cryptographic hashing in digital forensic

investigations by demonstrating how hash values can be used to verify that evidence remains unchanged throughout the investigation process.

Several researchers have focused on improving evidence collection and preservation techniques. Sammons [8] presented practical methodologies for conducting digital forensic investigations while maintaining evidence authenticity. Brown [9] discussed systematic procedures for evidence acquisition, storage, and preservation, emphasizing that secure evidence management significantly improves investigative reliability. Jøsang [10] highlighted the importance of identity management and role-based access control in protecting sensitive digital assets from unauthorized access, making secure authentication an essential component of modern forensic information systems.

Recent advancements in web technologies have enabled the development of secure, scalable, and user-friendly forensic applications. The Django framework [11] provides robust features such as authentication, session management, database integration, and security mechanisms that simplify the implementation of evidence management systems. Python [12] further supports rapid application development through its extensive libraries for cryptography, file processing, database operations, and web development, making it a preferred platform for forensic software development.

In addition to cryptographic approaches, blockchain technology has recently gained considerable attention in forensic evidence management. Researchers have proposed blockchain-based systems that record evidence transactions within immutable distributed ledgers, ensuring transparency, traceability, and resistance to tampering [13], [14]. These systems often integrate SHA-256 hashing, digital signatures, and consensus mechanisms to strengthen evidence integrity while maintaining decentralized audit trails. Although blockchain significantly enhances security, its implementation typically requires distributed infrastructure, smart contracts, specialized platforms, and higher computational resources, making deployment more complex and expensive.

Considering these limitations, lightweight cryptographic approaches have become attractive alternatives for practical forensic environments. The proposed Digital Evidence Management System adopts this approach by integrating SHA-256 hashing, role-based authentication, evidence metadata management, secure file storage, and hash-based verification within a Django-based web application. Unlike conventional evidence management systems, the proposed framework provides automated integrity verification, secure user management, centralized evidence tracking, and reliable tamper detection without requiring the complexity of a full blockchain infrastructure. Consequently, the developed system offers an efficient, scalable, and cost-effective solution for digital evidence management while preserving the integrity and authenticity required for forensic investigations and legal proceedings.

3. PROPOSED SYSTEM

The goal of the proposed Digital Evidence Management System (DEMS) is to offer a dependable, secure, and effective platform for the management, preservation, and verification of digital evidence used in court cases and forensic investigations. The SHA-256 cryptographic hashing method is used by the system, which was created using the Django web framework, to guaranty the legitimacy and integrity of uploaded evidence files.. Unlike traditional evidence management methods that rely solely on centralized storage and manual record-keeping, the proposed system integrates automated evidence verification, secure user authentication, and metadata management to maintain a trustworthy chain of custody.

The system follows a role-based access mechanism consisting of Administrator and Registered User modules. New users are required to register by providing their personal details and uploading a profile image. For enhanced security, newly registered accounts remain inactive until approved by the administrator. The administrator is responsible for activating or deactivating user accounts, monitoring uploaded evidence, and managing system users through a dedicated administrative dashboard.

Once authenticated, authorized users can upload digital evidence files through the web interface. During the upload process, the system reads the file contents and generates a unique SHA-256 evidence file is securely stored along with its associated metadata, including the uploader's information, filename, upload timestamp, and generated hash value. This metadata provides a reliable audit trail that supports evidence tracking and accountability throughout the investigation process.

To verify the authenticity of digital evidence, users can upload the evidence file again through the Verify Evidence module. The system regenerates the value. If both hash values are identical, the evidence is verified as authentic and unaltered. Otherwise, the system reports that the evidence has been modified or tampered with, ensuring rapid detection of unauthorized changes.

The proposed system also incorporates an email-based OTP password recovery mechanism, enabling users to securely reset forgotten passwords. Session management is implemented to maintain secure user authentication throughout each login session, while the administrator can monitor all uploaded evidence through a centralized dashboard.

Overall, the proposed Digital Evidence Management System provides a lightweight, secure, and scalable solution for digital evidence handling by combining SHA-256-based integrity verification, role-based access control, secure evidence storage, metadata tracking, and tamper detection. The system enhances the reliability, transparency, and accountability of forensic evidence management while

avoiding the complexity and deployment overhead associated with full blockchain-based infrastructures.

4. SYSTEM ARCHITECTURE

The architecture of the proposed Digital Evidence Management System (DEMS) is designed to provide secure user authentication, efficient evidence management, and reliable integrity verification through SHA-256 cryptographic hashing. The system follows a client–server architecture developed using the Django framework, where users interact with the web interface while the application server processes requests, manages the database, and performs evidence verification.

Initially, a user registers profile image. The registered account remains inactive until it is approved by the administrator. The administrator authenticates through the admin module and manages user accounts by activating or deactivating users, thereby ensuring that only authorized individuals can access the system.

The uploaded file is processed using representing the file's digital fingerprint. The system securely stores the evidence file in the server's storage while maintaining metadata such as the uploader's identity, filename, upload timestamp, and generated hash value in the database.

During the verification process, users upload the evidence file again through the verification module. The system computes a new SHA-256 hash for the uploaded file and compares it with the previously stored hash value. If both hash values match, the evidence is confirmed to be authentic and unchanged. Otherwise, the system identifies the evidence as modified or tampered with and notifies the user accordingly.

The architecture also includes an email-based OTP module that enables secure password recovery and session management to maintain authenticated user sessions. An administrative dashboard provides centralized monitoring of registered users and uploaded evidence records, ensuring complete traceability and accountability throughout the evidence lifecycle.

The overall architecture integrates user authentication, role-based access control, cryptographic hashing, secure evidence storage, metadata management, and evidence verification digital evidence management in forensic applications.

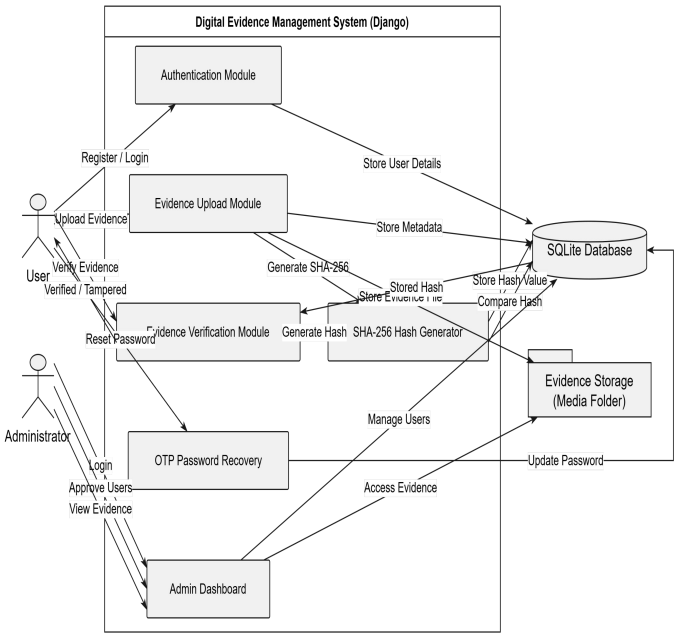

Figure 1. System Architecture of the Proposed Digital Evidence Management System (DEMS)

5. RESULTS AND DISCUSSION

The proposed Digital Evidence Management System (DEMS) was successfully implemented using the Django framework with SHA-256 cryptographic hashing for secure digital evidence verification. The system was evaluated by uploading original and modified evidence files to validate its integrity verification mechanism. identifies authentic evidence and effectively detects tampered files.

5.1 Successful Evidence Verification

Figure 5 illustrates the successful verification of an uploaded evidence file. During the verification process, the system computes the SHA-256 hash of the uploaded file and compares it with the corresponding hash stored in the database. Since both hash values are identical,

modified. The interface displays the verification status along with important metadata, including the file name, uploader, upload timestamp, and stored hash value, thereby maintaining a reliable chain of custody.

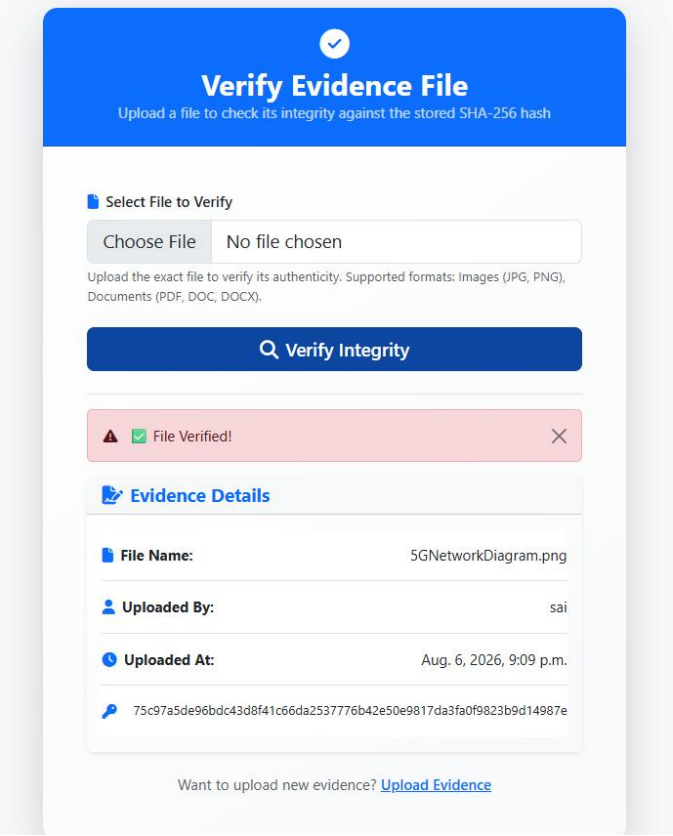

Figure 2. Successful Verification of Digital Evidence Using SHA-256 Hash Matching

5.2 Detection of Tampered Evidence

Figure 6 presents the verification result when a modified or unregistered evidence file is uploaded. The system generates a new SHA-256 hash and compares it with the stored hash values in the database. Since no matching hash is found, the evidence repository. This demonstrates the effectiveness of the proposed system in detecting unauthorized modifications and preventing the acceptance of compromised digital evidence.

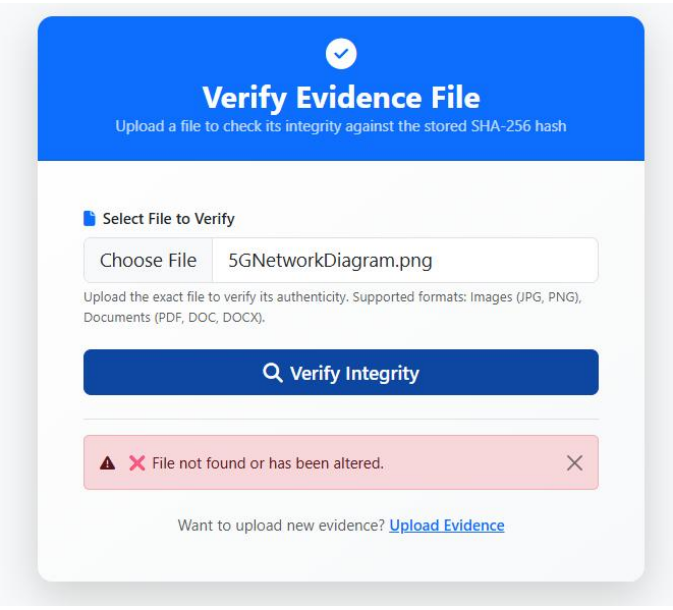

Figure 3. Detection of Modified or Tampered Digital Evidence

6. WEB APPLICATION IMPLEMENTATION

The proposed Digital Evidence Management System (DEMS) was and Python to provide secure,

efficient, and user-friendly digital evidence management. The application follows a role-based architecture consisting of Administrator and Registered User modules. The administrator is responsible for approving user accounts, managing registered users, and monitoring uploaded evidence, while authorized users can upload and verify digital evidence through an intuitive web interface.

The implementation begins with the user registration module, where users provide their profile image. Newly registered accounts remain inactive until approved by the administrator, ensuring that only authorized individuals can access the system. Once approved, users can securely log in and access the evidence management dashboard.

The evidence upload module enables authenticated users to upload digital evidence files. During the upload process, the system computes a SHA-256 cryptographic hash for the uploaded file and stores the generated hash value together with metadata such as the filename, uploader information, upload timestamp, and file location in the database. This process establishes a secure digital fingerprint for every evidence file and supports reliable evidence tracking.

The evidence verification module allows users to upload an evidence file for integrity verification. The system generates a new associated metadata. If the generated hash differs from the stored value, the system immediately reports that the file has been altered or is not present in the repository, thereby ensuring effective tamper detection.

The application also includes an email-based OTP password recovery mechanism, allowing users to securely reset forgotten passwords. Session management is implemented to maintain secure user authentication throughout the application. An administrative dashboard provides centralized management of users and evidence records, enabling efficient monitoring and maintaining a transparent audit trail.

Figures 2 and 3 illustrate the implementation results of the evidence verification module. Figure 5 demonstrates the successful verification of an authentic evidence file, while Figure 6 shows the detection of a modified or unregistered file through SHA-256 hash comparison. These implementation results validate the effectiveness of the proposed web application in preserving evidence integrity, preventing unauthorized modifications, and supporting secure digital forensic investigations.

7. CONCLUSION

The proposed Digital Evidence Management System (DEMS) provides a secure, reliable, and efficient platform for managing digital evidence in forensic investigations and legal proceedings. By integrating the SHA-256 cryptographic hashing algorithm with a Django-based web application, the system ensures the integrity and authenticity of uploaded evidence while enabling rapid detection of any unauthorized modifications. The implementation of role-based access control, administrator approval, metadata management, secure session handling, and email-based OTP password recovery enhances the overall security and usability of the application.

The experimental results demonstrate that the system successfully verifies authentic evidence by comparing cryptographic hash values and accurately identifies altered or tampered files. Furthermore, the centralized management of evidence metadata, including uploader details, timestamps, and hash values, provides a transparent audit trail that strengthens the chain of custody. Compared with traditional manual evidence management approaches, the proposed system offers improved security, traceability, accountability, and operational efficiency while maintaining a lightweight architecture suitable for practical deployment.

Overall, the developed system serves as a cost-effective and scalable solution for digital evidence management and can be effectively utilized by handling and verification of digital evidence. Future enhancements may include integration with blockchain technology, cloud-based storage, multi-factor authentication, and artificial intelligence-based forensic analysis to further improve system security, scalability, and automation.

8. FUTURE SCOPE

Although the secure evidence storage and reliable integrity verification using the SHA-256 cryptographic hashing algorithm, functionality, scalability, and security. One significant improvement is the integration of blockchain technology to maintain a decentralized and immutable chain of custody for digital evidence, ensuring greater transparency and resistance to unauthorized modifications.

The system can also be enhanced by integrating cloud-based storage, enabling secure access to digital evidence from multiple locations while ensuring high availability, backup, and disaster recovery. Implementing multi-factor authentication (MFA) and biometric authentication can further strengthen user authentication and

techniques to automate evidence classification, detect suspicious patterns, and assist forensic investigators in analyzing large volumes of can further improve the authenticity and non-repudiation of uploaded evidence.

Additionally, developing a mobile application would enable investigators to securely upload and verify evidence directly from smartphones or tablets during field investigations. The system can also be extended to support multiple forensic file formats, real-time evidence monitoring, automated report

generation, and interoperability with law enforcement databases. These enhancements would transform the proposed Digital Evidence Management System into a more comprehensive, intelligent, and scalable platform suitable for large-scale forensic investigations and judicial environments.

References

1. M. Pollitt, "Computer Forensics: An Approach to Evidence in Cyberspace," National Information Systems Security Conference Proceedings, Baltimore, MD, USA, pp. 487–491, 1995.
2. E. Casey, Computer Crime and Digital Evidence: Forensic Science, Computers, and the Internet, 3rd ed., Academic Press, 2011.
3. Guide to Integrating Forensic Techniques into Incident Response, K. Kent, S. Chevalier, T. Grance, and H. Dang, National Institute of Standards and Technology, NIST Special Publication 800-86, 2006.
4. Secure Hash Standard (SHS), FIPS PUB 180-4, National Institute of Standards and Technology, August 2015.
5. W. Stallings, Network Security and Cryptography: Principles and Practice, 8th ed. Pearson Education (2020).
6. B. Schneier, Applied Cryptography: Protocols, Algorithms, and Source Code in C, John Wiley & Sons, 2nd ed. (2015).
7. B. File System Forensic Analysis, Carrier. Professional Addison-Wesley, 2005.
8. J. Sammons, An Introduction to Digital Forensics, 2nd ed. Syngress (2015).
9. Computer Evidence: Collection and Preservation, C. Brown, Charles River Media, 2015, 2nd ed.
10. Digital Identity Management, A. Jøsang. Springer, 2017.
11. Django Software Foundation, "Django Documentation," accessible at https://docs.djangoproject.com/.
12. Python Software Foundation, "Python 3 Documentation,"
13. D. Eastlake and P. Jones, "US Secure Hash Algorithm 1 (SHA-1)," RFC 3174, Internet Engineering Task Force, September 2001; accessible at https://docs.python.org/3/.
14. M. Bishop, Computer Security: Art and Science, Addison-Wesley, 2018, 2nd ed.
15. Understanding Cryptography: A Textbook for Students and Practitioners, C. Paar and J. Pelzl, Springer, 2010.
16. "Hide and Seek: An Introduction to Steganography," N. Provos and P. Honeyman, IEEE Security & Privacy, vol. 1, no. 3, pp. 32–44, May–Jun. 2003.
17. The Guide to Secure Web Services by A. Singhal and T. Winograd, National Institute of Standards and Technology, NIST Special Publication 800-95, 2007.
18. Wiley, 2021; M. Stamp, Information Security: Principles and Practice, Third Edition.
19. Information Security: Concepts and Practices, R. C. Joshi and A. K. Misra. India: Wiley, 2011.
20. Digital Investigation, vol. 7, August 2010, pp. S64–S73; S. Garfinkel, "Digital Forensics Research: The Next 10 Years."
21. P. Ramesh Babu, "Reliable and Secure Banking Management System Thru Machine Learning," Journal of Applied Science and Computations, vol. 6, no. 4, pp. 377–382, April 2019.
22. P. Ramesh Babu, "Automated Student Document Analysis and Query System Using OCR and Conversational AI," pp. 358–366, 2026.
23. Security Engineering: A Handbook for Developing Dependable Distributed Systems, R. Anderson, Wiley, 2020, 3rd ed.
24. Handbook of Applied Cryptography, A. Menezes, P. van Oorschot, and S. Vanstone. CRC Press, 1996.
25. B. Schneier, Digital Security in a Networked World: Secrets and Lies, 2nd ed. Wiley (2015).