International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

The Role of Cryptography in Network Security: A Systematic Review and Emerging Trends

Daniel Makolo, Obafemi Babatunde Desmond, Dauda Shaibu Anibe, Ejiga Timothy Ikoojo, Lawal Lukman Adinoyi, Patience Ngozi Okoli, Idakwoji Joan Ojoache

Department of Computer Science, Faculty of Natural Sciences,

Prince Abubakar Audu University, P.M.B. 1008, Anyigba, Kogi State, Nigeria

Abstract - Cryptography is the backbone of modern network security, providing confidentiality, integrity, authentication, and non-repudiation for digital communication. However, the rapid evolution of cyber threats, particularly the looming arrival of large-scale quantum computers, poses serious challenges to the cryptographic algorithms that protect today's networks. This paper presents a systematic review of cryptography in network security, following the PRISMA 2020 guidelines. A total of 68 studies published between 2016 and 2025 were selected from five major academic databases: IEEE Xplore, ACM Digital Library, Scopus, Web of Science, and ScienceDirect. The review covers classical symmetric and asymmetric algorithms, widely deployed cryptographic protocols such as TLS 1.3, IPsec, and SSH, and the growing body of work on post-quantum cryptography (PQC). Key findings include the following: NIST finalized three post-quantum cryptographic standards (FIPS 203, 204, and 205) in August 2024; lightweight cryptography standards for IoT devices were published in 2025 with the selection of ASCON; and real-world deployment of hybrid classical/post-quantum schemes has already begun in major web browsers and messaging applications. This paper also examines emerging trends in homomorphic encryption, zero-knowledge proofs, and AI-driven cryptanalysis. Based on the findings, this review identifies critical gaps in PQC migration strategies, IoT security, and the integration of cryptography with artificial intelligence, and proposes directions for future research.

Keywords: Cryptography, Network Security, Symmetric Encryption, Asymmetric Encryption, Post-Quantum Cryptography, Lightweight Cryptography

INTRODUCTION

The growth of the internet and connected digital systems has made network security one of the most important concerns in computer science. Every day, billions of transactions, messages, and data transfers take place across networks that are, by design, open and shared. Cryptography is the primary tool used to protect this data from unauthorized access, tampering, and eavesdropping (Stallings, 2022). At its core, cryptography transforms readable data (plaintext) into an unreadable form (ciphertext) using mathematical algorithms and secret keys. Only authorized parties who hold the correct key can reverse the process and read the original data.

The history of modern cryptography can be traced back to the mid-1970s, when Diffie and Hellman (1976) introduced the concept of public-key cryptography, and Rivest, Shamir, and Adleman (1978) developed the RSA algorithm. Since then, cryptographic techniques have become deeply embedded in virtually every layer of network communication. Protocols such as the Transport Layer Security (TLS) protocol secure web traffic, the Internet Protocol Security (IPsec) framework protects virtual private networks (VPNs), and the Secure Shell (SSH) protocol enables safe remote access to servers. These protocols rely on a combination of symmetric encryption (for fast bulk data encryption), asymmetric encryption (for secure key exchange), and hash functions (for data integrity verification).

However, the cryptographic landscape is shifting. On one hand, attackers are becoming more sophisticated, exploiting implementation flaws such as side-channel vulnerabilities and protocol weaknesses. On the other hand, quantum computing is advancing rapidly. Shor's (1994) algorithm, if run on a sufficiently powerful quantum computer, can break RSA and elliptic curve cryptography (ECC) in polynomial time. Grover's (1996) algorithm effectively halves the security of symmetric ciphers. While a cryptographically relevant quantum computer (CRQC) capable of breaking current public-key systems does not yet exist, researchers estimate it could arrive within the next 10 to 15 years (Mosca & Piani, 2023). The threat is made more urgent by the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

"harvest now, decrypt later" (HNDL) strategy, where adversaries collect encrypted data today with the intention of decrypting it once quantum computers become available (Mascelli & Rodden, 2025).

In response to these threats, the National Institute of Standards and Technology (NIST) launched a post-quantum cryptography (PQC) standardization process in 2016. After eight years of evaluation involving 69 initial submissions, NIST published three final standards in August 2024: FIPS 203 (ML-KEM, a key encapsulation mechanism based on lattices), FIPS 204 (ML-DSA, a digital signature scheme based on lattices), and FIPS 205 (SLH-DSA, a hash-based signature scheme). These standards forms the beginning of a global migration from classical to quantum-resistant cryptography (NIST, 2024a, 2024b, 2024c).

At the same time, the explosive growth of the Internet of Things (IoT) has created a need for lightweight cryptographic algorithms that can run on devices with very limited processing power, memory, and battery life. NIST addressed this need by selecting ASCON as its lightweight cryptography standard in 2023, with the final standard (SP 800-232) published in August 2025 (Turan et al., 2025).

Given these rapid developments, there is a clear need for a comprehensive and up-to-date review that brings together the current state of cryptography in network security, the challenges posed by quantum computing and other threats, and the emerging solutions being developed and deployed. This paper addresses that need by conducting a systematic review following the PRISMA 2020 guidelines (Page et al., 2021). The review covers the period from 2016 to 2025 and addresses the following research questions:

RQ1: What are the current cryptographic algorithms and protocols used in network security, and how do they compare in terms of security and performance?

RQ2: What are the major threats and challenges facing traditional cryptographic systems?

RQ3: What is the current state of post-quantum cryptography standardization and deployment?

RQ4: What emerging cryptographic technologies and trends are shaping the future of network security?

LITERATURE REVIEW

The relationship between cryptography and network security has been the subject of extensive academic study over the past several decades. This section reviews the key literature that provides the theoretical and practical foundations for the present study.

Foundational Works in Cryptography

The field of modern public-key cryptography began with the landmark paper by Diffie and Hellman (1976), which introduced the idea that two parties could establish a shared secret over an insecure channel without needing to meet in advance. Shortly after, Rivest, Shamir, and Adleman (1978) developed the RSA algorithm, which became the first widely used public-key cryptosystem. On the symmetric side, the Data Encryption Standard (DES) served as the primary encryption algorithm for decades until it was replaced by the Advanced Encryption Standard (AES), which NIST standardized in 2001 (NIST, 2001). Stallings (2022) and Katz and Lindell (2021) provide comprehensive treatments of both classical and modern cryptographic techniques, covering symmetric ciphers, asymmetric systems, hash functions, and their applications in network security protocols.

Prior Surveys and Reviews

Several surveys have examined specific aspects of cryptography in network security. Thakor et al. (2021) reviewed lightweight cryptographic algorithms for resource-constrained IoT devices, comparing gate area, throughput, and energy consumption across multiple algorithm families. Lara-Nino et al. (2018) focused on elliptic curve lightweight cryptography, evaluating its suitability for embedded systems. Lou et al. (2022) surveyed microarchitectural side-channel vulnerabilities and the defenses available to protect cryptographic implementations. In the domain of post-quantum cryptography, Alagic et al. (2022) documented the NIST standardization process through its third round, while Lyubashevsky (2024) provided a tutorial on the lattice problems that underpin the ML-KEM and ML-DSA standards. Marcolla et al. (2022) surveyed fully homomorphic encryption schemes and their practical applications, and Mehic et al. (2020) examined quantum key distribution from a networking perspective.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Identified Gap

While these individual surveys cover their respective domains thoroughly, no single review brings together all of these topics, namely classical algorithms, network security protocols, quantum threats, post-quantum standardization, IoT cryptography, and emerging applications, into a unified and current assessment. The rapid pace of developments in 2024 and 2025, including the finalization of NIST's PQC standards and the ASCON lightweight standard, means that much of the existing survey literature is already outdated on key points. This systematic review fills that gap by providing a comprehensive, PRISMA-compliant assessment of the entire cryptographic landscape in network security as of 2025.

METHODOLOGY

This systematic review follows the PRISMA 2020 statement (Page et al., 2021) and adapts the guidelines for software engineering systematic literature reviews proposed by Kitchenham and Charters (2007) and later updated by Kitchenham, Madeyski, and Budgen (2023) in the SEGRESS framework. The methodology consists of five steps: defining the search strategy, setting inclusion and exclusion criteria, selecting studies, extracting data, and assessing quality.

Search Strategy

The literature search was conducted across five major academic databases: IEEE Xplore, ACM Digital Library, Scopus, Web of Science, and ScienceDirect. These databases were chosen because they index the leading journals and conference proceedings in computer science, cybersecurity, and information technology. The search was supplemented by backward snowballing (checking the reference lists of selected papers) and forward snowballing (checking papers that cited selected works), following the procedure described by Wohlin (2014).

The search string was constructed using Boolean operators to combine terms related to the key concepts of the review. The primary search string was: ("cryptography" OR "encryption" OR "cryptographic") AND ("network security" OR "cybersecurity" OR "information security") AND ("review" OR "survey" OR "analysis" OR "framework" OR "protocol" OR "algorithm"). The search was applied to titles, abstracts, and keywords. The search was conducted in January 2025 and covered publications from January 2016 to December 2025.

Table 1. Summary of Search Strategy

Component	Details
Databases	IEEE Xplore, ACM Digital Library, Scopus, Web of Science, ScienceDirect
Time period	January 2016 to December 2025
Language	English
Document types	Journal articles, conference papers, and standards documents
Supplementary	Backward and forward snowballing (Wohlin, 2014); NIST and IETF standards

Inclusion and Exclusion Criteria

Studies were included if they met all of the following criteria: (a) the study addresses cryptographic algorithms, protocols, or systems used in or relevant to network security; (b) the study is published in a peer-reviewed journal, conference, or as an official standards document; (c) the study is written in English; and (d) the full text is accessible. Studies were excluded if they: (a) focus exclusively on cryptography for non-network applications such as database encryption without a network component; (b) are duplicates across databases; (c) are editorials, opinion pieces, or abstracts without full text; or (d) were published before 2016, unless they are foundational works that are essential for context (such as the original RSA, Diffie-Hellman, or Shor's algorithm papers).

Study Selection Process

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

The study selection followed a three-stage screening process. In the first stage, the initial database search returned 1,247 records. After removing 389 duplicates, 858 unique records remained. In the second stage, titles and abstracts were screened against the inclusion and exclusion criteria, which eliminated 682 records. In the third stage, the full texts of the remaining 176 articles were assessed for eligibility. Of these, 108 were excluded for reasons including: insufficient focus on network security (n = 41), lack of peer review (n = 27), full text not available (n = 19), and being superseded by a more recent version of the same work (n = 21). This process resulted in 68 primary studies included in the review. An additional 14 foundational works and standards documents were included as supplementary sources. The selection process is illustrated in Figure 1.

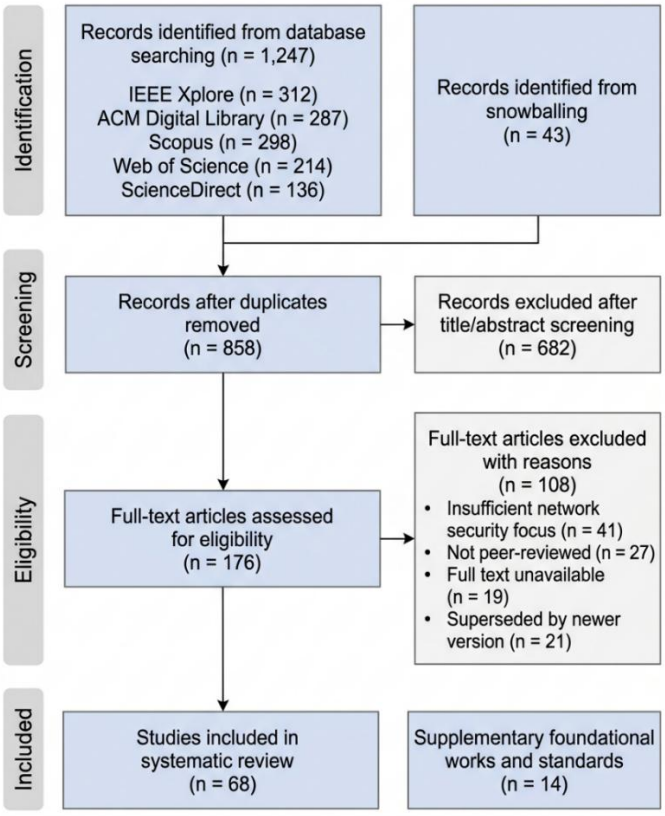

Figure 1. *PRISMA 2020 flow diagram showing the study selection process*

Data Extraction and Synthesis

A data extraction form was developed to capture key information from each included study. The extracted data included: author(s), year, title, publication venue, study type (empirical, theoretical, survey, or implementation), cryptographic domain (symmetric, asymmetric, hash, protocol, PQC, lightweight, or emerging), key findings, and limitations. The extracted data were organized thematically and synthesized using a narrative synthesis approach. Quantitative data, such as algorithm performance metrics and key sizes, were tabulated for comparison.

Quality Assessment

The quality of each included study was assessed using a checklist adapted from Dyba and Dingsoyr (2008), which evaluates studies on three dimensions: rigor (is the research methodology clearly described and appropriate?), credibility (are the findings supported by the data?), and relevance (is the study relevant to the review questions?). Each study was scored on a scale of 0 (not met), 0.5 (partially met), or 1 (fully met) for each criterion. Studies scoring below 50% on the quality assessment were flagged for further scrutiny but were not automatically excluded, following the recommendations of Kitchenham and Charters (2007).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

RESULTS

This section presents the findings of the systematic review, organized thematically across six areas: fundamentals of cryptography, cryptographic protocols, challenges to traditional cryptography, post-quantum cryptography, cryptography in IoT and resource-constrained devices, and emerging applications and trends.

Fundamentals of Cryptography in Network Security

Cryptography can be broadly divided into three categories: symmetric key cryptography, asymmetric key cryptography, and hash functions. Each serves a different purpose in network security, and modern systems typically combine all three to achieve a balanced trade-off between security and performance (Katz & Lindell, 2021). Figure 2 provides a visual taxonomy of the major cryptographic techniques discussed in this section.

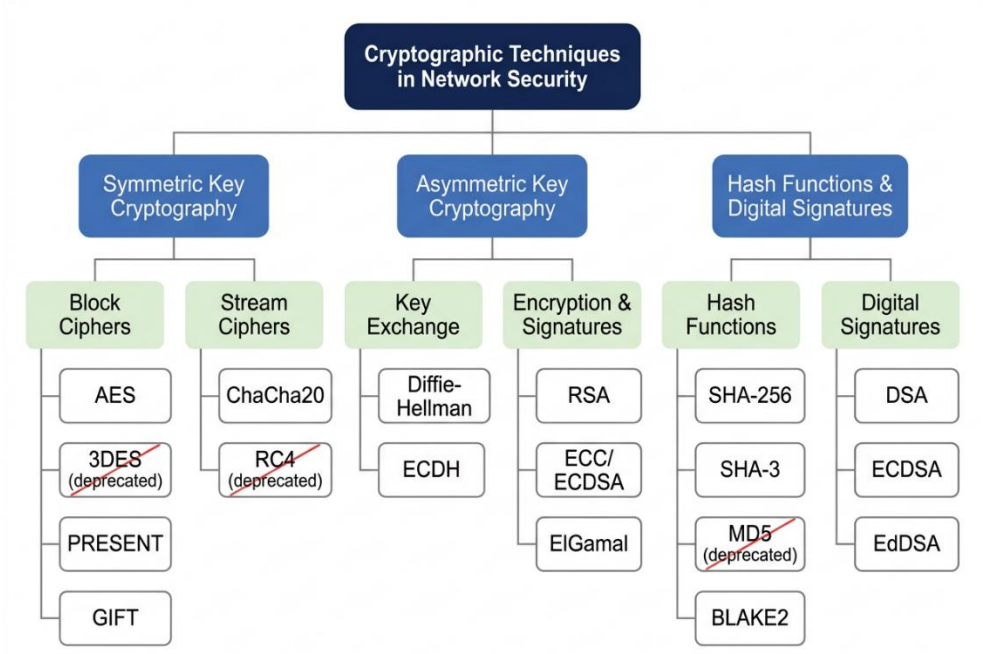

Figure 2. *Taxonomy of cryptographic techniques in network security*

Symmetric Key Cryptography

Symmetric key cryptography uses a single shared secret key for both encryption and decryption. Because both the sender and receiver must know the same key, distributing that key securely is a fundamental challenge. However, symmetric algorithms are very fast, which makes them the standard choice for encrypting large volumes of data in real time (Stallings, 2022).

The most widely used symmetric algorithm today is the Advanced Encryption Standard (AES), specified in FIPS 197 (NIST, 2001). AES operates on 128-bit blocks and supports three key sizes: 128, 192, and 256 bits, with 10, 12, and 14 rounds of processing respectively. AES with a 128-bit key provides approximately 128 bits of security, which is considered sufficient for most applications. On modern processors with hardware acceleration (Intel AES-NI or ARM Cryptographic Extensions), AES can achieve throughput of several gigabytes per second (Paar & Pelzl, 2010).

The Data Encryption Standard (DES), once the dominant symmetric cipher, was officially withdrawn by NIST in 2005 due to its short 56-bit key, which is vulnerable to brute-force attacks (NIST, 2019). Triple DES (3DES), which applies DES three times with two or three different keys, extended the effective key length to 112 or 168 bits. However, 3DES is also being phased out; NIST disallowed its use for new applications after 2023 because of its small 64-bit block size, which creates vulnerabilities when encrypting large amounts of data (NIST, 2019).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Another notable symmetric cipher is ChaCha20, designed by Bernstein (2008) and standardized in RFC 8439 (Nir & Langley, 2018). ChaCha20 paired with the Poly1305 authenticator is widely used in TLS 1.3 as an alternative to AES-GCM. Its design uses only addition, rotation, and XOR (ARX) operations, making it particularly efficient on devices that lack dedicated AES hardware, such as older smartphones and embedded systems.

Table 2. Comparison of Symmetric Encryption Algorithms

Algorithm	Key Size (bits)	Block Size (bits)	Rounds	Status	Primary Use Case
AES	128/192/256	128	10/12/14	Active standard	Bulk data encryption, TLS, disk encryption
3DES	112/168	64	48	Deprecated (2023)	Legacy banking systems
ChaCha20	256	Stream	20	Active (RFC 8439)	TLS 1.3, mobile devices without AES-NI
DES	56	64	16	Withdrawn (2005)	Historical only

Asymmetric Key Cryptography

Asymmetric (public-key) cryptography uses a pair of mathematically related keys: a public key that can be shared openly, and a private key that must be kept secret. The concept was introduced by Diffie and Hellman (1976), and the first practical implementation was the RSA algorithm by Rivest, Shamir, and Adleman (1978). RSA's security depends on the difficulty of factoring large integers. A typical RSA key today is 2,048 or 4,096 bits long. While RSA is well understood and widely trusted, it is relatively slow compared to symmetric encryption, so it is typically used only for key exchange and digital signatures rather than bulk data encryption.

Elliptic Curve Cryptography (ECC), independently proposed by Koblitz (1987) and Miller (1986), offers the same level of security as RSA but with much smaller key sizes. For example, a 256-bit ECC key provides roughly equivalent security to a 3,072-bit RSA key (Johnson et al., 2001; Hankerson et al., 2004). This efficiency advantage makes ECC the preferred choice for environments where bandwidth, storage, or processing power are limited, including mobile devices and IoT systems (Lara-Nino et al., 2018). The Elliptic Curve Digital Signature Algorithm (ECDSA) and Elliptic Curve Diffie-Hellman (ECDH) are the most widely used ECC-based schemes in practice.

Table 3. Comparison of Asymmetric Encryption Algorithms

Algorithm	Key Size (bits)	Security (bits)	Hard Problem	Quantum Resistant?
RSA-2048	2,048	~112	Integer factoring	No (broken by Shor's algorithm)
RSA-4096	4,096	~140	Integer factoring	No
ECDSA P-256	256	~128	ECDLP	No (broken by Shor's algorithm)
ECDH P-384	384	~192	ECDLP	No
DH-2048	2,048	~112	DLP	No

Note. ECDLP = Elliptic Curve Discrete Logarithm Problem; DLP = Discrete Logarithm Problem.

Hash Functions

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

A cryptographic hash function takes an input of any size and produces a fixed-length output (called a digest or hash value) that acts as a fingerprint of the input data. Good hash functions are one-way (it is computationally infeasible to recover the input from the output) and collision-resistant (it is extremely difficult to find two different inputs that produce the same output). These properties make hash functions essential for verifying data integrity, storing passwords, and building digital signatures (Katz & Lindell, 2021).

SHA-256, part of the SHA-2 family specified in FIPS 180-4 (NIST, 2015a), is the most widely used hash function today. It produces a 256-bit digest and is used in TLS certificates, blockchain (Bitcoin), code signing, and many other applications. SHA-3, standardized in FIPS 202 (NIST, 2015b), is based on an entirely different design (the Keccak sponge construction) and serves as a backup in case vulnerabilities are found in SHA-2. Older hash functions such as MD5 and SHA-1 are considered broken for security purposes. Wang and Yu (2005) demonstrated practical collision attacks on MD5, and Stevens et al. (2017) showed that SHA-1 collisions can be produced at a cost of roughly $110,000 in cloud computing resources.

Digital Signatures

Digital signatures combine asymmetric cryptography with hash functions to provide authentication, integrity, and non-repudiation. To sign a message, the sender computes a hash of the message and then encrypts the hash with their private key. The recipient can verify the signature by decrypting it with the sender's public key and comparing the result to their own hash of the message. If they match, the signature is valid (Stallings, 2022). Common digital signature algorithms include RSA signatures, DSA, ECDSA (Johnson et al., 2001), and the more recent EdDSA (Bernstein et al., 2012), which is based on twisted Edwards curves and is used in protocols like SSH and Signal.

Cryptographic Protocols in Network Security

Cryptographic algorithms do not work in isolation. They are combined into protocols that handle the complexities of real-world communication: negotiating which algorithms to use, exchanging keys securely, authenticating parties, and encrypting data in transit. This subsection reviews the major cryptographic protocols used in network security today.

Transport Layer Security (TLS)

TLS is the most widely deployed cryptographic protocol on the internet. It secures communication between web browsers and servers (HTTPS), email clients and servers, and many other application-layer protocols. TLS 1.3, published as RFC 8446 (Rescorla, 2018), is a major overhaul compared to TLS 1.2. The key improvements include the removal of insecure legacy features such as RSA key exchange (which does not provide forward secrecy), static Diffie-Hellman, and cipher suites using CBC mode or RC4. TLS 1.3 mandates the use of Authenticated Encryption with Associated Data (AEAD) ciphers such as AES-GCM and ChaCha20-Poly1305. It also reduces the handshake from two round trips to one (1-RTT), and supports an optional zero round trip (0-RTT) mode for resumption (Dowling et al., 2021).

The security of TLS 1.3 has been formally analyzed in multiple works. Dowling et al. (2021) provided a cryptographic proof of the TLS 1.3 handshake in a multi-stage key exchange model, covering full 1-RTT, PSK, and PSK-ECDHE modes. Adoption has been strong: as of mid-2024, approximately 70% of websites support TLS 1.3, and over 94% of Google's web traffic uses encrypted connections (Zhou et al., 2024).

Internet Protocol Security (IPsec)

IPsec operates at the network layer (Layer 3 of the OSI model) and is the standard framework for securing VPN connections. It consists of two main sub-protocols: the Authentication Header (AH), which provides integrity and authentication, and the Encapsulating Security Payload (ESP), which adds encryption. IPsec can operate in transport mode (encrypting only the payload) or tunnel mode (encrypting the entire original IP packet, including headers). Key exchange is handled by the Internet Key Exchange version 2 (IKEv2) protocol, specified in RFC 7296 (Kaufman et al., 2014). NIST SP 800-77 Rev. 1 (Barker et al., 2020) provides comprehensive guidance on IPsec VPN deployment, including algorithm selection and configuration. Abbas et al. (2023) offer a detailed security assessment framework for VPN technologies in their ACM Computing Surveys paper.

Secure Shell (SSH)

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

SSH is the standard protocol for secure remote login and command execution on servers. It uses a combination of asymmetric cryptography for authentication and key exchange, symmetric encryption for session data, and MACs for integrity. Recent security research has identified notable vulnerabilities. The Terrapin attack (Baumer et al., 2024), disclosed in late 2023, exploited a flaw in SSH's handshake sequence numbering to truncate encrypted messages. The researchers found that approximately 77% of SSH servers supported at least one vulnerable cipher mode. In July 2024, the regreSSHion vulnerability (CVE-2024-6387) was disclosed, which allowed unauthenticated remote code execution in OpenSSH server versions 8.5p1 through 9.7p1 (Qualys, 2024). This was the first unauthenticated RCE in OpenSSH in roughly 18 years, highlighting that even mature, well-audited protocols can contain critical flaws.

Wireless Security Protocols

WPA3, introduced by the Wi-Fi Alliance in 2018, replaced WPA2 as the standard for wireless network security. Its key improvement is the use of the Simultaneous Authentication of Equals (SAE) handshake, also known as Dragonfly, which replaces WPA2's Pre-Shared Key (PSK) handshake. SAE is based on a password-authenticated key exchange protocol that resists offline dictionary attacks. WPA3 also mandates 128-bit encryption for personal networks and 192-bit encryption for enterprise networks (Halbouni et al., 2023). However, WPA3 has not been immune to attacks. Vanhoef and Ronen (2020) identified downgrade attacks and side-channel vulnerabilities in the Dragonfly handshake (the "Dragonblood" vulnerabilities), and Chatzoglou et al. (2022) demonstrated denial-of-service attacks against WPA3-SAE on Wi-Fi 6 equipment.

Challenges to Traditional Cryptography

The Quantum Computing Threat

The most significant long-term threat to traditional cryptography comes from quantum computing. In 1994, Peter Shor published a quantum algorithm that can factor large integers and compute discrete logarithms in polynomial time (Shor, 1994). If implemented on a quantum computer with enough stable qubits, Shor's algorithm would completely break RSA, Diffie-Hellman, and ECC, which together form the basis of nearly all public-key cryptography used on the internet today.

Grover's algorithm (Grover, 1996) poses a different kind of threat. It provides a quadratic speedup for brute-force search, which effectively halves the security level of symmetric ciphers. For example, AES-128 would offer only 64 bits of security against a quantum adversary, making it potentially vulnerable. AES-256, however, would retain 128 bits of security, which is still considered safe.

How close are we to a quantum computer that can actually break current encryption? The honest answer is: probably not close, but the timeline is uncertain. Gidney and Ekera (2021) estimated that breaking RSA-2048 would require approximately 20 million noisy physical qubits running for about 8 hours. A 2025 preprint by Gidney reduced this estimate to under 1 million noisy qubits (Gidney, 2025), which is significant because several hardware roadmaps target that range by around 2030. Meanwhile, Google's Willow processor (105 qubits) demonstrated the first below-threshold quantum error correction in late 2024 (Google Quantum AI and Collaborators, 2025), and IBM's Condor processor reached 1,121 qubits in December 2023, though these are still far from the scale needed for cryptanalysis (AbuGhanem, 2025). A 2023 survey by the Global Risk Institute found that roughly 23% of experts believe there is a greater than 50% chance that RSA-2048 will be breakable by a quantum computer within 10 years (Mosca & Piani, 2023).

Harvest Now, Decrypt Later

Even though a CRQC does not exist yet, the threat is not purely theoretical. Nation-state adversaries and other well-resourced attackers can intercept and store encrypted communications today, planning to decrypt them in the future when quantum computers become available. This strategy, known as "harvest now, decrypt later" (HNDL), is particularly concerning for data that must remain confidential for decades, such as state secrets, medical records, financial data, and intellectual property. A 2025 Federal Reserve working paper analyzed the HNDL risk specifically for blockchain networks, concluding that it represents a present and ongoing threat because blockchain transaction data is permanently stored on public ledgers (Mascelli & Rodden, 2025). NIST's draft transition guidance (IR 8547) explicitly acknowledges the HNDL threat as a reason to begin PQC migration now rather than waiting for quantum computers to arrive (NIST, 2024d).

Side-Channel Attacks

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Even when a cryptographic algorithm is mathematically secure, its implementation may leak information through unintended physical channels. Side-channel attacks exploit measurable physical properties such as execution time, power consumption, electromagnetic emissions, or cache access patterns to extract secret keys. Kocher (1996) introduced timing attacks, showing that small variations in the time taken to perform cryptographic operations can reveal private keys. Kocher et al. (1999) later introduced differential power analysis (DPA), which uses statistical analysis of power traces to extract keys from hardware implementations.

More recent work has shown that these attacks remain a serious concern. Lou et al. (2022) surveyed microarchitectural side-channel attacks, including cache-timing attacks (Flush+Reload, Prime+Probe) and speculative execution attacks (Spectre). Wang et al. (2022) demonstrated the Hertzbleed attack, which exploits CPU frequency scaling to turn power side-channel leakage into remotely observable timing differences. Importantly, Ravi et al. (2024) showed that post-quantum algorithms like Kyber and Dilithium are also vulnerable to side-channel and fault-injection attacks, meaning that the transition to PQC does not automatically solve implementation security problems.

Key Management Challenges

Managing cryptographic keys throughout their lifecycle, from generation and distribution to storage, rotation, and destruction, is one of the most difficult operational aspects of network security. Poor key management has been the cause of numerous real-world breaches. NIST SP 800-57 (Barker, 2020) provides comprehensive guidance on key management practices, but many organizations struggle to implement these practices consistently, especially across large-scale distributed networks with thousands of endpoints. The upcoming transition to PQC could potentially compound these challenges, as organizations will need to manage both classical and post-quantum keys during the migration period (NIST, 2024d).

Post-Quantum Cryptography

Post-quantum cryptography (PQC) refers to cryptographic algorithms that are designed to be secure against attacks by both classical and quantum computers. Unlike quantum cryptography (such as QKD), which uses quantum physics to distribute keys, PQC algorithms run on ordinary classical computers. The goal is to replace vulnerable RSA, DH, and ECC schemes with new algorithms based on mathematical problems that are believed to be hard even for quantum computers.

NIST PQC Standardization

NIST initiated its PQC standardization effort in December 2016 with a public call for proposals. The process received 82 submissions, of which 69 were accepted as complete for the first round of evaluation. Over the next six years, the candidates were narrowed through successive rounds of public analysis, culminating in July 2022 with the selection of four algorithms: CRYSTALS-Kyber (key encapsulation), CRYSTALS-Dilithium (digital signatures), FALCON (digital signatures), and SPHINCS+ (digital signatures). On August 13, 2024, NIST published three final standards (Alagic et al., 2022; NIST, 2024a, 2024b, 2024c):

FIPS 203 (ML-KEM): Module-Lattice-Based Key-Encapsulation Mechanism, derived from CRYSTALS-Kyber. It offers three parameter sets (ML-KEM-512, 768, and 1024) targeting NIST security levels 1, 3, and 5 respectively. ML-KEM is the recommended algorithm for key exchange in protocols like TLS.

FIPS 204 (ML-DSA): Module-Lattice-Based Digital Signature Algorithm, derived from CRYSTALS-Dilithium. It is the primary PQC digital signature standard, also based on Module-LWE and Module-SIS problems (Ducas et al., 2018).

FIPS 205 (SLH-DSA): Stateless Hash-Based Digital Signature Algorithm, derived from SPHINCS+. Its security relies entirely on the security of the underlying hash function, making it the most conservative choice (Bernstein et al., 2019).

FALCON, a lattice-based signature scheme with the smallest combined public key and signature sizes among the NIST candidates, is still being developed as FIPS 206 under the name FN-DSA (Fouque et al., 2020). Additionally, in March 2025, NIST selected HQC (a code-based KEM) as a backup to ML-KEM, providing algorithmic diversity.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

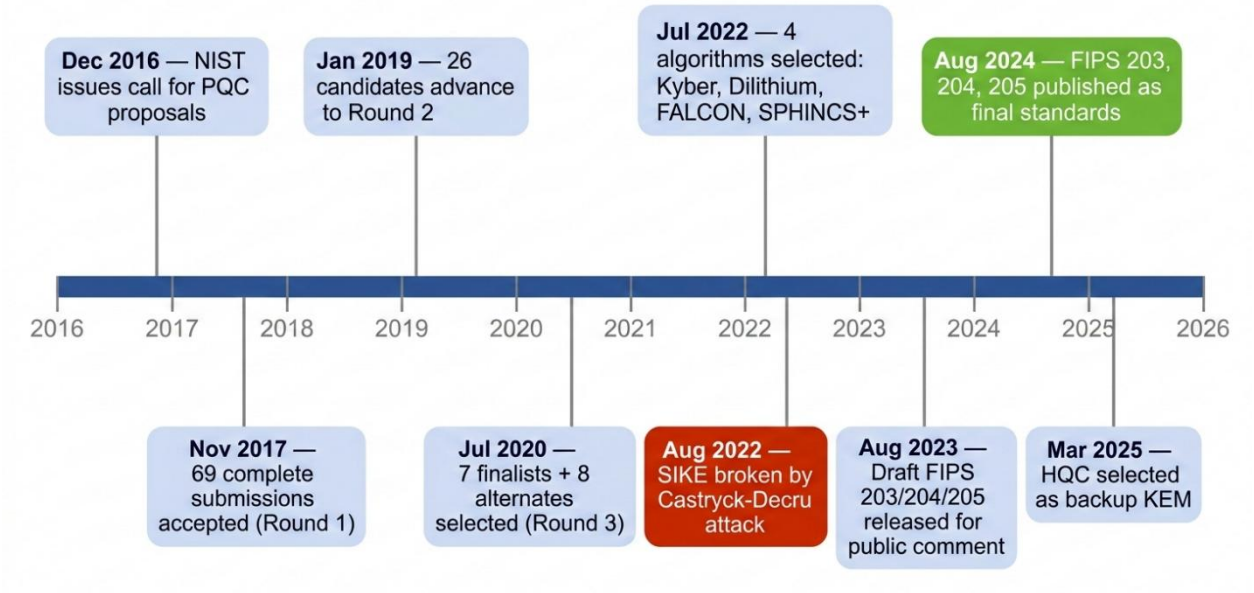

Figure 3. *Timeline of the NIST Post-Quantum Cryptography standardization process (2016 to 2025)*

Table 4. Comparison of NIST Post-Quantum Cryptographic Standards

Algorithm	Standard	Type	Public Key	Sig/CT Size	NIST Level	Basis
ML-KEM-768	FIPS 203	KEM	1,184 B	1,088 B	3	Module-LWE
ML-DSA-65	FIPS 204	Signature	1,952 B	3,293 B	3	Module-LWE/SIS
SLH-DSA-128f	FIPS 205	Signature	32 B	17,088 B	1	Hash functions
Falcon-512	Draft 206	Signature	897 B	~666 B	1	NTRU lattice
RSA-2048*	(classical)	Sig/Enc	256 B	256 B	~112-bit	Factoring
ECDSA P-256*	(classical)	Signature	64 B	64 B	~128-bit	ECDLP

Note. B = bytes; CT = ciphertext; KEM = Key Encapsulation Mechanism. Asterisk (*) denotes classical (non-quantum-resistant) algorithms included for comparison. Data from NIST (2024a, 2024b, 2024c) and Alagic et al. (2022).

PQC Algorithm Families

PQC algorithms can be grouped into several families based on the mathematical problems they rely on. Lattice-based cryptography, which underpins both ML-KEM and ML-DSA, is based on the hardness of problems like Learning With Errors (LWE) and Short Integer Solution (SIS) on lattices. These problems are well-studied and believed to be hard for both classical and quantum computers. Lattice-based schemes generally offer good performance and moderate key sizes, making them the most practical choice for widespread deployment (Lyubashevsky, 2024).

Hash-based signatures, such as SLH-DSA (SPHINCS+), rely only on the security of hash functions, which makes them the most conservative option from a security standpoint. However, they produce significantly larger signatures (17 KB for SLH-DSA-128f compared to 3.3 KB for ML-DSA-65), which can be a drawback for bandwidth-constrained applications (Bernstein et al., 2019).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Code-based cryptography, based on the difficulty of decoding random linear codes, has a long history going back to the McEliece cryptosystem of 1978. NIST selected HQC, a code-based KEM, in March 2025 as a backup to the lattice-based ML-KEM, providing important algorithmic diversity in case lattice problems turn out to be weaker than expected.

It is worth noting that not all PQC candidates survived the evaluation process. Isogeny-based cryptography, represented by SIKE, was a Round 4 candidate until Castryck and Decru (2023) published a devastating polynomial-time key recovery attack that broke SIKE's security entirely. The attack, which could recover keys from SIKEp434 in roughly one hour on a single CPU core, serves as a reminder that the security of any cryptographic scheme is only as strong as the hardness of its underlying mathematical problem.

Hybrid Approaches and Crypto-Agility

Given the uncertainty about the long-term security of newly standardized PQC algorithms, many organizations are adopting a hybrid approach that combines a classical algorithm with a PQC algorithm. In a hybrid key exchange, for example, a TLS connection might use both ECDH (classical) and ML-KEM (post-quantum) in parallel. The connection is secure as long as at least one of the two algorithms remains unbroken. Several major technology companies have already begun deploying hybrid PQC in production. Google Chrome enabled the X25519Kyber768 hybrid key exchange by default starting in Chrome 124 (April 2024) and transitioned to ML-KEM768 combined with X25519 in Chrome 131 (November 2024). Signal adopted the PQXDH protocol, which pairs X25519 with ML-KEM for its end-to-end encrypted messaging. AWS deployed hybrid ECDHE combined with Kyber for its Key Management Service and Secrets Manager.

Closely related to hybrid deployment is the concept of crypto-agility: the ability of a system to swap out cryptographic algorithms without requiring major changes to the underlying infrastructure. NIST's NCCoE has published guidance on achieving crypto-agility as a key enabler for the PQC transition (NIST NCCoE, 2024). The importance of crypto-agility is underscored by the SIKE break; organizations that had experimentally deployed SIKE needed to be able to replace it quickly.

Cryptography in IoT and Resource-Constrained Devices

The Internet of Things comprises billions of connected devices, many of which have severe constraints on processing power, memory, energy, and bandwidth. Standard cryptographic algorithms like AES (which requires approximately 15,000 gate equivalents per round) may be too resource-intensive for the smallest IoT devices such as RFID tags, smart sensors, and medical implants (Thakor et al., 2021). This has driven the development of lightweight cryptographic algorithms specifically designed for constrained environments.

Lightweight Cryptographic Algorithms

The lightweight cryptography research community has produced a large number of block ciphers, stream ciphers, hash functions, and authenticated encryption schemes optimized for low-resource devices. PRESENT (Bogdanov et al., 2007) is one of the earliest and most studied lightweight block ciphers, operating on 64-bit blocks with 80 or 128-bit keys and implementable in approximately 1,570 gate equivalents. It was standardized in ISO/IEC 29192. The SIMON and SPECK families (Beaulieu et al., 2015), designed by the NSA, offer flexible block and key sizes. SIMON is optimized for hardware (using AND, rotation, and XOR operations), while SPECK is optimized for software. GIFT (Banik et al., 2017) improved on PRESENT by offering better energy efficiency and formed the basis for the GIFT-COFB authenticated encryption scheme, which was a finalist in the NIST lightweight cryptography competition.

The NIST Lightweight Cryptography Standard: ASCON

In 2018, NIST launched a dedicated lightweight cryptography standardization project, receiving 57 submissions. After multiple rounds of evaluation, NIST selected ASCON as the winning algorithm in February 2023, and the final standard was published as SP 800-232 in August 2025 (Turan et al., 2025). ASCON, designed by Dobraunig et al. (2021), uses a sponge construction with a 320-bit internal state and provides 128-bit security for both authenticated encryption (ASCON-AEAD128) and hashing (ASCON-Hash256). It can be implemented in as few as 2,600 gate equivalents while achieving throughput of 4.9 to 7.3 Gbps in hardware. This makes it well suited for everything from tiny RFID tags to more capable IoT gateways. On an ARM Cortex-M3 microcontroller at 84 MHz, ASCON ranks among the top performers for authenticated encryption (Rana et al., 2022).

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Table 5. *Comparison of Selected Lightweight Cryptographic Algorithms*

Algorithm	Block (bits)	Key (bits)	Type	Gate Area (GE)	Standard	Optimized For
ASCON	Sponge	128	AEAD + Hash	~2,600	NIST SP 800-232	Hardware + software
PRESENT	64	80/128	Block cipher	~1,570	ISO/IEC 29192	Hardware (RFID)
GIFT-128	128	128	Block cipher	~1,700	Research	Energy efficiency
SIMON-64	64	96/128	Block cipher	~1,300	NSA design	Hardware (AND/XOR)
SPECK-64	64	96/128	Block cipher	~2,400	NSA design	Software (ARX)
AES-128*	128	128	Block cipher	~15,000	FIPS 197	General purpose

Note. GE = Gate Equivalents (area in hardware). Asterisk (*) denotes AES included for comparison. Data compiled from Thakor et al. (2021) and Rana et al. (2022).

Emerging Applications and Trends

Homomorphic Encryption

Homomorphic encryption (HE) allows computations to be performed directly on encrypted data without first decrypting it. The result, when decrypted, is identical to what would have been obtained by performing the same computation on the plaintext. Fully homomorphic encryption (FHE) supports both addition and multiplication on ciphertexts, enabling arbitrary computations. Since Gentry's (2009) breakthrough construction, practical FHE schemes such as BGV, BFV, TFHE, and CKKS have been developed, along with open-source libraries including Microsoft SEAL, HElib, and OpenFHE (Marcolla et al., 2022). While FHE remains orders of magnitude slower than plaintext computation, recent work on hardware acceleration using FPGAs and GPUs, as well as algorithmic optimizations, has significantly improved performance (Gong et al., 2024). Key applications include privacy-preserving machine learning, secure cloud computing, and confidential data analytics.

Zero-Knowledge Proofs

A zero-knowledge proof (ZKP) allows one party (the prover) to convince another party (the verifier) that a statement is true without revealing any information beyond the truth of the statement itself. ZKPs have become a cornerstone technology in blockchain and cryptocurrency systems, where they enable transaction privacy while maintaining verifiability. The two main families of non-interactive ZKPs are zk-SNARKs (used in Zcash and various Ethereum Layer 2 solutions) and zk-STARKs (which do not require a trusted setup and are post-quantum resistant). Oude Roelink et al. (2024) provide a systematic review of 41 studies comparing zk-SNARKs, zk-STARKs, and Bulletproofs across financial, medical, and business domains.

AI and Machine Learning in Cryptography

The intersection of artificial intelligence and cryptography is a growing area of research. On the offensive side, Gohr (2019) demonstrated that deep neural networks could be trained to perform differential cryptanalysis on the SPECK32/64 block cipher, outperforming traditional statistical methods. This seminal work, published at CRYPTO 2019, has spawned over 200 follow-up publications exploring neural cryptanalysis on various ciphers (Gerault et al., 2024). On the defensive side, machine learning is being explored for tasks such as automated detection of side-channel leakage, optimization of cryptographic implementations, and classification of encrypted network traffic. Privacy-preserving machine learning, which combines cryptographic techniques (HE, MPC, or secure enclaves) with ML models to enable inference and training on sensitive data without exposing it, is another active area of research (Ng & Chow, 2023).

Quantum Key Distribution

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

Quantum Key Distribution (QKD) uses the principles of quantum mechanics to establish shared secret keys between two parties with information-theoretic security. The BB84 protocol, proposed by Bennett and Brassard (1984), is the most well-known QKD protocol. Unlike PQC, which relies on computational hardness assumptions, QKD's security is based on the laws of physics: any attempt to intercept the quantum key exchange disturbs the quantum states and is detectable. Real-world QKD networks have been deployed, including China's 2,000-kilometer Beijing-Shanghai backbone and various European metropolitan networks. However, QKD has significant practical limitations, including the need for dedicated fiber optic or satellite links, limited key rates over long distances, and high equipment costs (Mehic et al., 2020; Dervisevic et al., 2025). For these reasons, PQC and QKD are generally viewed as complementary technologies rather than competitors.

DISCUSSION

This systematic review reveals several important findings about the current state and trajectory of cryptography in network security.

First, the classical cryptographic foundations remain solid for the time being, but the clock is ticking. AES, SHA-256, and ECC continue to provide strong security against classical attacks, and protocols like TLS 1.3 have significantly raised the bar by removing legacy weaknesses. However, the quantum threat is not something that can be addressed later. The HNDL strategy means that sensitive data encrypted today with RSA or ECC could be compromised retroactively once quantum computers arrive. This makes the PQC transition not just a future concern but a present imperative.

Second, the NIST PQC standardization process has reached a critical milestone with the publication of FIPS 203, 204, and 205. The standards are final and ready for implementation. Real-world deployment has already begun, with hybrid PQC integrated into major web browsers (Chrome, Firefox), messaging applications (Signal), and cloud services (AWS, Cloudflare). The challenge now shifts from algorithm design to operational migration: conducting cryptographic inventories, updating libraries and protocols, managing hybrid key material, and ensuring backward compatibility. NIST's recommendation to deprecate all traditional public-key algorithms by 2035 (NIST, 2024d) provides a clear timeline but also reflects the scale of the effort required.

Third, IoT security remains a major concern. The sheer diversity of IoT devices, from powerful edge gateways to tiny sensors with kilobytes of memory, means there is no one-size-fits-all cryptographic solution. The ASCON standard addresses the need for lightweight authenticated encryption, but the integration of PQC into IoT devices remains largely an open problem. Post-quantum key sizes (1 to 2 KB for public keys, compared to 32 to 64 bytes for ECC) may be prohibitive for the most constrained devices.

Fourth, emerging technologies like homomorphic encryption, zero-knowledge proofs, and AI-driven cryptanalysis are opening new possibilities and creating new challenges simultaneously. FHE, while still too slow for many real-time applications, is becoming practical for specific use cases like privacy-preserving analytics. ZKPs are already deployed at scale in blockchain systems. And neural cryptanalysis, though still limited to reduced-round ciphers, suggests that AI could eventually play a meaningful role in both attacking and defending cryptographic systems.

CONCLUSIONS AND RECOMMENDATIONS

Cryptography is the foundation on which network security is built. This systematic review has examined the current state of cryptographic algorithms, protocols, threats, and emerging technologies based on 68 primary studies and 14 supplementary sources published between 2016 and 2025. The findings confirm that while classical cryptography continues to serve well against current threats, the approaching reality of quantum computing requires urgent action. NIST's publication of the first three post-quantum cryptographic standards in August 2024 marks a turning point. The transition from classical to post-quantum cryptography is no longer a theoretical exercise; it is an active, ongoing process with real-world deployments already underway in major browsers, messaging platforms, and cloud services.

At the same time, this review highlights that cryptography in network security extends well beyond just replacing algorithms. Securing IoT devices with lightweight cryptography, protecting data in use with homomorphic encryption, enabling privacy on blockchains with zero-knowledge proofs, and defending against AI-powered attacks are all critical areas that will shape this field in

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

the coming years. The research community, standards bodies, and industry must work together to ensure that these transitions happen securely, efficiently, and in time.

Based on the findings of this review, the following directions for future research are recommended:

PQC migration strategies for legacy systems. While NIST has published transition guidance, practical migration strategies for large organizations with diverse legacy systems remain underdeveloped. Research is needed on automated cryptographic inventory tools, migration cost models, and risk-based prioritization frameworks.

Lightweight PQC for IoT. Adapting post-quantum algorithms for resource-constrained IoT devices is a critical open problem. Current PQC key sizes and computational requirements are too large for many IoT devices. Research into hardware-optimized PQC implementations, PQC-aware lightweight protocols, and hybrid classical/PQC schemes for IoT is needed.

Side-channel resistance of PQC implementations. As Ravi et al. (2024) have shown, PQC algorithms are not inherently resistant to side-channel attacks. Developing and verifying constant-time, side-channel-resistant implementations of ML-KEM, ML-DSA, and other PQC standards is essential before widespread deployment.

Practical homomorphic encryption. Reducing the computational overhead of FHE to make it practical for a broader range of applications, particularly in healthcare, finance, and government, is an important research direction. Hardware acceleration using specialized processors, FPGAs, or ASICs is a promising avenue.

AI in cryptanalysis and defense. Expanding neural cryptanalysis beyond reduced-round toy ciphers to full-strength algorithms, and developing AI-based tools for automated vulnerability detection in cryptographic implementations, are emerging research frontiers.

Crypto-agility standards and tooling. Developing standardized frameworks and tools that enable organizations to achieve true crypto-agility, allowing them to swap algorithms quickly in response to new attacks or standards, is essential for long-term security resilience.

IMPLICATIONS

The findings of this review carry important implications for both research and practice. For the research community, the review highlights several underexplored areas that require urgent attention. The integration of post-quantum cryptographic algorithms into resource-constrained IoT devices remains an open problem, and further work is needed on hardware-optimized PQC implementations. The vulnerability of PQC algorithms to side-channel attacks, as demonstrated by Ravi et al. (2024), underscores the need for implementation-level security research alongside algorithm-level design. Additionally, the growing capability of AI-driven cryptanalysis tools suggests that future cipher designs may need to account for machine learning-based attacks from the outset.

For practitioners, including network administrators, security architects, and IT decision-makers, the review provides a clear call to action. Organizations should begin conducting cryptographic inventories to identify where vulnerable algorithms are deployed, prioritize systems that handle long-lived sensitive data (which are most exposed to the HNDL threat), and start testing hybrid classical/post-quantum configurations in their environments. The NIST 2035 deprecation timeline for traditional public-key algorithms provides a concrete planning horizon, but given the complexity of large-scale cryptographic migrations, early action is strongly advisable.

For policymakers and standards bodies, the review reinforces the importance of continued investment in PQC standardization, lightweight cryptography research, and the development of crypto-agility frameworks. The SIKE break demonstrated that even carefully vetted candidates can fail, making algorithmic diversity and the ability to rapidly swap algorithms essential features of any long-term security strategy.

ACKNOWLEDGEMENT

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

The authors acknowledge the Department of Computer Science, Faculty of Natural Sciences, Prince Abubakar Audu University, Anyigba, for providing the academic environment and resources that supported this research.

DECLARATIONS

Conflict of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Informed Consent

Not applicable. This study is a systematic literature review that did not involve human participants or primary data collection from individuals.

Ethics Approval

Not applicable. This study is a systematic literature review that did not involve human participants, animal subjects, or primary data collection requiring ethics approval.

REFERENCES

[1] Abbas, H., Emmanuel, N., Amjad, M. F., Ahmed, S., Junaid, M., & Iqbal, Z. (2023). Security assessment and evaluation of VPNs: A comprehensive survey. *ACM Computing Surveys*, 55(13s), Article 273, 1–47. https://doi.org/10.1145/3579162

[2] AbuGhanem, M. (2025). IBM quantum computers: Evolution, performance, and future directions. *Journal of Supercomputing*, 81, 687. https://doi.org/10.1007/s11227-025-07047-7

[3] Alagic, G., Apon, D., Cooper, D., Dang, Q., Dang, T., Kelsey, J., Lichtinger, J., Liu, Y.-K., Miller, C., Moody, D., Peralta, R., Perlner, R., Robinson, A., & Smith-Tone, D. (2022). *Status report on the third round of the NIST post-quantum cryptography standardization process* (NIST IR 8413-upd1). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.IR.8413-upd1

[4] Banik, S., Pandey, S. K., Peyrin, T., Sasaki, Y., Sim, S. M., & Todo, Y. (2017). *GIFT: A small present. In Cryptographic Hardware and Embedded Systems* (CHES 2017) (LNCS 10529, pp. 321–345). Springer. https://doi.org/10.1007/978-3-319-66787-4_16

[5] Barker, E. (2020). *Recommendation for key management: Part 1 – General* (NIST SP 800-57 Part 1 Rev. 5). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-57pt1r5

[6] Barker, E., Dang, Q., Frankel, S., Scarfone, K., & Wouters, P. (2020). *Guide to IPsec VPNs* (NIST SP 800-77 Rev. 1). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-77r1

[7] Baumer, F., Brinkmann, M., & Schwenk, J. (2024). *Terrapin attack: Breaking SSH channel integrity by sequence number manipulation*. In Proceedings of the 33rd USENIX Security Symposium.

[8] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., & Wingers, L. (2015). *The SIMON and SPECK families of lightweight block ciphers* (IACR ePrint 2015/585).

[9] Bennett, C. H., & Brassard, G. (2014). Quantum cryptography: Public key distribution and coin tossing. *Theoretical Computer Science*, 560, 7–11. https://doi.org/10.1016/j.tcs.2014.05.025

[10] Bernstein, D. J. (2008). *ChaCha, a variant of Salsa20*. Workshop Record of SASC 2008. https://cr.yp.to/chacha/chacha-20080128.pdf

[11] Bernstein, D. J., Duif, N., Lange, T., Schwabe, P., & Yang, B.-Y. (2012). High-speed high-security signatures. *Journal of Cryptographic Engineering*, 2(2), 77–89. https://doi.org/10.1007/s13389-012-0027-1

[12] Bernstein, D. J., Hulsing, A., Kolbl, S., Niederhagen, R., Rijneveld, J., & Schwabe, P. (2019). *The SPHINCS+ signature framework*. In Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security (pp. 2129–2146). https://doi.org/10.1145/3319535.3363229

[13] Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J. B., Seurin, Y., & Vikkelsoe, C. (2007). *PRESENT: An ultra-lightweight block cipher. In Cryptographic Hardware and Embedded Systems* (CHES 2007) (LNCS 4727, pp. 450–466). Springer. https://doi.org/10.1007/978-3-540-74735-2_31

[14] Castryck, W., & Decru, T. (2023). *An efficient key recovery attack on SIDH. In Advances in Cryptology – EUROCRYPT 2023* (LNCS 14008, pp. 423–447). Springer. https://doi.org/10.1007/978-3-031-30589-4_15

[15] Chatzoglou, E., Kambourakis, G., & Kolias, C. (2022). How is your Wi-Fi connection today? DoS attacks on WPA3-SAE. *Journal of Information Security and Applications*, 64, 103058. https://doi.org/10.1016/j.jisa.2021.103058

[16] Dervisevic, E., Tankovic, A., Fazel, E., Mehic, M., Maurhart, O., & Schrenk, B. (2025). *Quantum key distribution networks – Key management: A survey*. ACM Computing Surveys. https://doi.org/10.1145/3730575

[17] Diffie, W., & Hellman, M. (1976). New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6), 644–654. https://doi.org/10.1109/TIT.1976.1055638

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

[18] Dobraunig, C., Eichlseder, M., Mendel, F., & Schlaffer, M. (2021). Ascon v1.2: Lightweight authenticated encryption and hashing. *Journal of Cryptology*, 34, Article 33. https://doi.org/10.1007/s00145-021-09398-9

[19] Dowling, B., Fischlin, M., Gunther, F., & Stebila, D. (2021). A cryptographic analysis of the TLS 1.3 handshake protocol. *Journal of Cryptology*, 34(4), Article 37, 1–69. https://doi.org/10.1007/s00145-021-09384-1

[20] Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G., & Stehle, D. (2018). CRYSTALS-Dilithium: A lattice-based digital signature scheme. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 2018(1), 238–268. https://doi.org/10.13154/tches.v2018.i1.238-268

[21] Dyba, T., & Dingsoyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9–10), 833–859. https://doi.org/10.1016/j.infsof.2008.01.006

[22] Fouque, P.-A., Hoffstein, J., Kirchner, P., Lyubashevsky, V., Pornin, T., Prest, T., Ricosset, T., Seiler, G., Whyte, W., & Zhang, Z. (2020). *Falcon: Fast-Fourier lattice-based compact signatures over NTRU*. NIST PQC Submission v1.2. https://falcon-sign.info/falcon.pdf

[23] Gentry, C. (2009). *Fully homomorphic encryption using ideal lattices*. In Proceedings of the 41st ACM Symposium on Theory of Computing (STOC '09) (pp. 169–178). https://doi.org/10.1145/1536414.1536440

[24] Gerault, D., Peyrin, T., Tan, Q. Q., & Udovenko, A. (2024). *SoK: Six years of neural differential cryptanalysis* (IACR ePrint 2024/1300).

[25] Gidney, C. (2025). *How to factor 2048 bit RSA integers with less than a million noisy qubits*. arXiv preprint, arXiv:2505.15917.

[26] Gidney, C., & Ekera, M. (2021). How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits. *Quantum*, 5, 433. https://doi.org/10.22331/q-2021-04-15-433

[27] Gohr, A. (2019). *Improving attacks on round-reduced Speck32/64 using deep learning*. In Advances in Cryptology – CRYPTO 2019 (LNCS 11693, pp. 150–179). Springer. https://doi.org/10.1007/978-3-030-26951-7_6

[28] Gong, Y., Chang, X., Mišić, J., Mišić, V. B., Bao, L., & Wang, J. (2024). Practical solutions in fully homomorphic encryption: A survey analyzing existing acceleration methods. *Cybersecurity*, 7, 5. https://doi.org/10.1186/s42400-023-00187-4

[29] Google Quantum AI and Collaborators. (2025). Quantum error correction below the surface code threshold. *Nature*, 638, 920–926. https://doi.org/10.1038/s41586-024-08449-y

[30] Grover, L. K. (1996). *A fast quantum mechanical algorithm for database search*. In Proceedings of the 28th ACM Symposium on Theory of Computing (pp. 212–219). https://doi.org/10.1145/237814.237866

[31] Halbouni, A., Ong, L.-Y., & Leow, M. C. (2023). Wireless security protocols WPA3: A systematic literature review. *IEEE Access*, 11, 112438–112450. https://doi.org/10.1109/ACCESS.2023.3322931

[32] Hankerson, D., Menezes, A., & Vanstone, S. (2004). *Guide to elliptic curve cryptography*. Springer. https://doi.org/10.1007/b97644

[33] Johnson, D., Menezes, A., & Vanstone, S. (2001). The elliptic curve digital signature algorithm (ECDSA). *International Journal of Information Security*, 1(1), 36–63. https://doi.org/10.1007/s102070100002

[34] Katz, J., & Lindell, Y. (2021). *Introduction to modern cryptography (3rd ed.)*. CRC Press. https://doi.org/10.1201/9781351133036

[35] Kaufman, C., Hoffman, P., Nir, Y., Eronen, P., & Kivinen, T. (2014). *Internet Key Exchange Protocol Version 2* (IKEv2) (RFC 7296). https://doi.org/10.17487/RFC7296

[36] Kitchenham, B. A., & Charters, S. (2007). G*uidelines for performing systematic literature reviews in software engineering* (Technical Report EBSE-2007-01). Keele University.

[37] Kitchenham, B. A., Madeyski, L., & Budgen, D. (2023). SEGRESS: Software engineering guidelines for reporting secondary studies. *IEEE Transactions on Software Engineering*, 49(3), 1273–1298. https://doi.org/10.1109/TSE.2022.3174092

[38] Koblitz, N. (1987). Elliptic curve cryptosystems. *Mathematics of Computation*, 48(177), 203–209. https://doi.org/10.1090/S0025-5718-1987-0866109-5

[39] Kocher, P. C. (1996). *Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems*. In Advances in Cryptology – CRYPTO '96 (LNCS 1109, pp. 104–113). Springer. https://doi.org/10.1007/3-540-68697-5_9

[40] Kocher, P., Jaffe, J., & Jun, B. (1999). *Differential power analysis. In Advances in Cryptology – CRYPTO '99* (LNCS 1666, pp. 388–397). Springer. https://doi.org/10.1007/3-540-48405-1_25

[41] Lara-Nino, C. A., Diaz-Perez, A., & Morales-Sandoval, M. (2018). Elliptic curve lightweight cryptography: A survey. *IEEE Access*, 6, 72514–72550. https://doi.org/10.1109/ACCESS.2018.2881444

[42] Lou, X., Zhang, T., Jiang, J., & Zhang, Y. (2022). A survey of microarchitectural side-channel vulnerabilities, attacks, and defenses in cryptography. *ACM Computing Surveys*, 54(6), Article 122, 1–37. https://doi.org/10.1145/3456629

[43] Lyubashevsky, V. (2024). *Basic lattice cryptography: The concepts behind Kyber* (ML-KEM) and Dilithium (ML-DSA) (IACR ePrint 2024/1287).

[44] Marcolla, C., Sucasas, V., Manzano, M., Bassoli, R., Fitzek, F. H. P., & Aaraj, N. (2022). *Survey on fully homomorphic encryption, theory, and applications*. Proceedings of the IEEE, 110(10), 1572–1609. https://doi.org/10.1109/JPROC.2022.3205665

[45] Mascelli, J., & Rodden, M. (2025). *"Harvest now decrypt later": Examining post-quantum cryptography and the data privacy risks for distributed ledger networks* (FEDS Working Paper No. 2025-093). Federal Reserve Board. https://doi.org/10.17016/FEDS.2025.093

[46] Mehic, M., Niemiec, M., Rass, S., Ma, J., Peev, M., Aguado, A., Martin, V., Schauer, S., Poppe, A., Pacher, C., & Voznak, M. (2020). Quantum key distribution: A networking perspective. *ACM Computing Surveys*, 53(5), Article 96. https://doi.org/10.1145/3402192

[47] Miller, V. S. (1986). *Use of elliptic curves in cryptography. In Advances in Cryptology – CRYPTO '85* (LNCS 218, pp. 417–426). Springer. https://doi.org/10.1007/3-540-39799-X_31

[48] Mosca, M., & Piani, M. (2023). *2023 quantum threat timeline report*. Global Risk Institute.

[49] Ng, L. K. L., & Chow, S. S. M. (2023). *SoK: Cryptographic neural-network computation*. In IEEE Symposium on Security and Privacy (S&P 2023) (pp. 497–514). https://doi.org/10.1109/SP46215.2023.10179483

[50] Nir, Y., & Langley, A. (2018). *ChaCha20 and Poly1305 for IETF protocols* (RFC 8439). https://doi.org/10.17487/RFC8439

[51] NIST NCCoE. (2024). *Migration to post-quantum cryptography* (NIST CSWP 39). https://www.nccoe.nist.gov/crypto-agility-considerations-migrating-post-quantum-cryptographic-algorithms

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 05 , May - 2026

[52] NIST. (2001, updated 2023). *Advanced Encryption Standard* (AES) (FIPS PUB 197). https://doi.org/10.6028/NIST.FIPS.197-upd1

[53] NIST. (2015a). *Secure Hash Standard* (SHS) (FIPS PUB 180-4). https://doi.org/10.6028/NIST.FIPS.180-4

[54] NIST. (2015b). *SHA-3 Standard: Permutation-based hash and extendable-output functions* (FIPS PUB 202). https://doi.org/10.6028/NIST.FIPS.202

[55] NIST. (2019). *Transitioning the use of cryptographic algorithms and key lengths* (SP 800-131A Rev. 2). https://doi.org/10.6028/NIST.SP.800-131Ar2

[56] NIST. (2024a). *Module-Lattice-Based Key-Encapsulation Mechanism Standard* (FIPS 203). https://doi.org/10.6028/NIST.FIPS.203

[57] NIST. (2024b). *Module-Lattice-Based Digital Signature Standard* (FIPS 204). https://doi.org/10.6028/NIST.FIPS.204

[58] NIST. (2024c). *Stateless Hash-Based Digital Signature Standard* (FIPS 205). https://doi.org/10.6028/NIST.FIPS.205

[59] NIST. (2024d). *Transition to post-quantum cryptography standards* (IR 8547, Initial Public Draft).

[60] Oude Roelink, B., El-Hajj, M., & Sarmah, D. (2024). Systematic review: Comparing zk-SNARK, zk-STARK, and Bulletproof protocols for privacy-preserving authentication. *Security and Privacy*, 7(5), e401. https://doi.org/10.1002/spy2.401

[61] Paar, C., & Pelzl, J. (2010). *Understanding cryptography: A textbook for students and practitioners*. Springer. https://doi.org/10.1007/978-3-642-04101-3

[62] Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., Hoffmann, T. C., Mulrow, C. D., Shamseer, L., Tetzlaff, J. M., Akl, E. A., Brennan, S. E., Chou, R., Glanville, J., Grimshaw, J. M., Hróbjartsson, A., Lalu, M. M., Li, T., Loder, E. W., Mayo-Wilson, E., McDonald, S., ... Moher, D. (2021). *The PRISMA 2020 statement: An updated guideline for reporting systematic reviews*. BMJ, 372, n71. https://doi.org/10.1136/bmj.n71

[63] Qualys TRU. (2024). regreSSHion: An unauthenticated remote code execution in OpenSSH server (CVE-2024-6387). https://www.qualys.com/regresshion-cve-2024-6387

[64] Rana, M., Mamun, Q., & Islam, R. (2022). Lightweight cryptography in IoT networks: A survey. *Future Generation Computer Systems*, 129, 77–89. https://doi.org/10.1016/j.future.2021.11.011

[65] Ravi, P., Chattopadhyay, A., D'Anvers, J. P., & Baksi, A. (2024). Side-channel and fault-injection attacks over lattice-based post-quantum schemes (Kyber, Dilithium): Survey and new results. *ACM Transactions on Embedded Computing Systems*, 23(2), Article 35. https://doi.org/10.1145/3603170

[66] Rescorla, E. (2018). *The Transport Layer Security (TLS) Protocol Version 1.3* (RFC 8446). https://doi.org/10.17487/RFC8446

[67] Rivest, R. L., Shamir, A., & Adleman, L. M. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2), 120–126. https://doi.org/10.1145/359340.359342

[68] Shor, P. W. (1994). *Algorithms for quantum computation: Discrete logarithms and factoring*. In Proceedings of the 35th IEEE Symposium on Foundations of Computer Science (pp. 124–134). https://doi.org/10.1109/SFCS.1994.365700

[69] Stallings, W. (2022). *Cryptography and network security: Principles and practice (8th ed.)*. Pearson.

[70] Stevens, M., Bursztein, E., Karpman, P., Albertini, A., & Markov, Y. (2017). *The first collision for full SHA-1. In Advances in Cryptology – CRYPTO 2017* (LNCS 10401, pp. 570–596). Springer. https://doi.org/10.1007/978-3-319-63688-7_19

[71] Thakor, V. A., Razzaque, M. A., & Khandaker, M. R. A. (2021). Lightweight cryptography algorithms for resource-constrained IoT devices: A review, comparison and research opportunities. *IEEE Access*, 9, 28177–28193. https://doi.org/10.1109/ACCESS.2021.3052867

[72] Turan, M. S., McKay, K. A., Chang, D., Kang, J., & Kelsey, J. (2025). *Ascon-based lightweight cryptography standards for constrained devices* (NIST SP 800-232). https://doi.org/10.6028/NIST.SP.800-232

[73] Vanhoef, M., & Ronen, E. (2020). *Dragonblood: Analyzing the Dragonfly handshake of WPA3 and EAP-pwd*. In IEEE Symposium on Security and Privacy (S&P 2020).

[74] Wang, X., & Yu, H. (2005). *How to break MD5 and other hash functions. In Advances in Cryptology – EUROCRYPT 2005* (LNCS 3494, pp. 19–35). Springer. https://doi.org/10.1007/11426639_2

[75] Wang, Y., Paccagnella, R., He, E. T., Shacham, H., Fletcher, C. W., & Kohlbrenner, D. (2022). *Hertzbleed: Turning power side-channel attacks into remote timing attacks on x86*. In Proceedings of the 31st USENIX Security Symposium.

[76] Wohlin, C. (2014). *Guidelines for snowballing in systematic literature studies*. In Proceedings of EASE '14, Article 38. https://doi.org/10.1145/2601248.2601268

[77] Zhou, J., Fu, W., Hu, W., Sun, Z., He, T., & Zhang, Z. (2024). Challenges and advances in analyzing TLS 1.3-encrypted traffic: A comprehensive survey. *Electronics*, 13(20), 4000. https://doi.org/10.3390/electronics13204000