Evolving Attack Tactics against Messaging Applications

Shakhzod Yuldoshkhujaev*, Suyeon Lee*, Hyungjoon Koo**

Department of Computer Science and Engineering,
Sungkyunkwan University (Graduate Student*, Professor**)

Abstract

The adoption of End-to-End Encryption (E2EE) in messengers such as Signal, WhatsApp, and Telegram has reshaped the security landscape of digital communication. While cryptographic primitives like the Signal Protocol provide strong confidentiality and forward secrecy, real-world vulnerabilities arise from how these primitives are composed, implemented, and deployed. This survey reviews academic research on attacks against secure messaging systems across five areas: protocol pitfalls, implementation errors and insecure deployment, metadata exposure, phishing threats, and ecosystem and supply-chain risks. The collected studies show that weaknesses originate less from cryptography itself than from its integration into complex systems and user environments. Tracing these works reveals four consistent trends: a shift from primitives to composite protocols, from content to metadata, from implementation/deployment to human factors, and from an application to the ecosystem. We suggest future work toward reliable session and group management, practical metadata-leakage mitigations, safer in-app user defenses, and stronger distribution integrity to secure messaging applications against evolving attacks.

I. Introduction

Over the past decade, digital communication platforms have foregrounded End-to-End Encryption (E2EE) as a core security property. Applications such as Signal and WhatsApp deploy E2EE by default, whereas Telegram offers it only in Secret Chats. In their E2EE modes (e.g., Signal, WhatsApp, and Telegram Secret Chats), messages are designed so that only the communicating parties can read the content, precluding even the service provider from access. The Signal Protocol in particular underpins these guarantees and is widely deployed, offering strong security properties.

Despite strong cryptographic foundations, security in practice hinges on components beyond the cipher suite. Weaknesses in protocol composition can lead to post-compromise security; implementation defects surface in deployments; metadata artifacts enable sender–recipient linkability; user-level attacks scale via phishing kits; and unofficial “mod” clients broaden supply-chain exposure. Accordingly, we analyze five axes of evolving attack tactics: (i) protocol pitfalls, (ii) implementation errors and insecure deployment, (iii) metadata exposure, (iv) phishing threats, and (v) ecosystem and supply-chain risks.

The primary contribution of our paper is a structured analysis of the evolving threat

This work was supported by the 2025 Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (Ministry of Science and ICT) (No. RS-2024-00398745, Proofs and Responses against Evidence Tampering in the New Digital Environment).

landscape. By organizing prior research, this work traces the shifting attack surfaces: from cryptographic primitives to composite protocols, from content to metadata, from implementation and deployment to human factors, and from individual applications to the ecosystem. It also outlines key directions for future work, including reliable session and group management, practical metadata protection measures, safer in-app user defenses, and stronger distribution integrity for the surrounding ecosystem.

II. Background

Modern secure messaging is built on the principle of E2EE, where messages are encrypted on the sender's device and can only be decrypted by the recipient's device. The service provider, which relays the messages, only has access to the encrypted ciphertext and cannot read the message content. This model is designed to protect communications from surveillance by the platform provider, network eavesdroppers, and state actors. The Signal Protocol is the most prominent E2EE protocol and is used by major applications like Signal and WhatsApp. It provides several key security properties:

- **Confidentiality:** Ensures that message content is unreadable to anyone other than the intended recipients.
- **Integrity:** Guarantees that messages cannot be tampered with in transit without detection.
- **Perfect Forward Secrecy (PFS):** Protects past messages even if a user's long-term keys are compromised. Even if an attacker obtains a user's long-term key, past messages remain secure since session keys are generated through ephemeral Diffie-Hellman exchanges and never reused.
- **Post-Compromise Security (PCS):** Also known as "future secrecy" this property allows a protocol to restore confidentiality after a key compromise. It applies when a lost device exposes its keys—once the legitimate user resumes messaging, the protocol automatically establishes new session keys and excludes the attacker from future communication.

While these properties provide a robust cryptographic foundation, the security of the entire system depends on more than just the core encryption algorithms. The application logic, software implementation, communication patterns, and user behavior all present potential avenues for attack. In this work, we use the E2EE model and its guarantees as the baseline for classifying and analyzing attacks beyond the cryptographic core.

III. Attack Surfaces on Messengers

Existing studies consistently demonstrate that vulnerabilities in secure messaging rarely originate from cryptographic primitives themselves. Instead, they emerge in how these primitives are composed into protocols, implemented and deployed, exposed through metadata, exploited through user manipulation, and extended across the untrusted ecosystem.

Protocol Pitfalls. Group messaging remains fragile when administrative actions (add/remove, role changes) are not cryptographically authorized. Balbás et al. [1] formalize cryptographic administration for secure group messaging and give efficient, secure constructions. This enforces administrator control and prevents inconsistent group views—capturing PFS/PCS goals and aligning with modern standards efforts. Another pitfall arises from group logic. Cremers et al. [2] show that application-level session handling can

undermine conversation-level guarantees, where device cloning and inconsistent session management may break expected security properties until recovery.

Implementation Errors and Insecure Deployment. Even a perfectly designed protocol can be undermined by errors in its software implementation or insecure deployment. Kobeissi et al. [3] formally analyze a JavaScript implementation of a Signal-variant, uncovering replay and key-compromise impersonation attacks. Their analysis shows that missing message-state validation and inconsistent handling of ephemeral keys allowed adversaries to reuse or replay protocol messages.

Metadata Exposure. As E2EE makes message content inaccessible, adversaries have turned their attention to metadata that is information about who is communicating with whom, when, and for how long. Signal's "Sealed-Sender" feature was designed to hide the sender's identity from the Signal server. However, Martiny et al. [4] show a statistical disclosure attack, amplified by delivery receipts, which can link sealed-sender conversations in a handful of messages; the paper proposes deployable mitigations. The attack exploits the automatic and predictable timing of delivery receipts, allowing the server to correlate an anonymous message to a recipient with the subsequent receipt sent back to the original sender, thereby de-anonymizing the conversation.

Phishing Threats. Messaging apps amplify trust and immediacy, which adversaries exploit with phishing kits distributed via social platforms and secure messengers. A longitudinal study by Bijmans et al. [5] collected 70 kits, clustered them into 10 families, and fingerprinted deployments across 1,363 phishing domains; campaigns exhibited a median domain uptime of 24 hours, illustrating rapid, kit-driven operations that are easily seeded and coordinated through messaging ecosystems. These results show that cryptography alone does not mitigate user-level manipulation; effective defenses require in-app safeguards such as safer link handling, sender cues, and recovery workflows informed by real attacker behavior.

Ecosystem and Supply-Chain Threats. Security boundaries extend beyond the official client. Munyendo et al. [6] show that unofficial "mod" clients, adopted for extra features, shift risk to distribution and provenance: they are typically sideloaded from third-party sites, request broader permissions than the official app, and in some cases contain malware. Consequently, the threat model expands from a single application to the surrounding app-distribution and maintenance ecosystem, calling for provenance checks, permission hygiene, and caution against mod use in sensitive contexts.

IV. Evolving Attack Tactics

Our survey indicates a consistent migration of attack surfaces. As E2EE primitives have standardized and hardened, attention has shifted from the cryptographic core to surrounding layers of the system. This shift unfolds along four axes.

From Primitives to Composite Protocols. Research has moved from analyzing cryptographic designs themselves to examining how primitives are composed within protocols. Works on group messaging and session handling show that weak composition can undermine post-compromise and forward secrecy, even when underlying algorithms remain secure.

From Content to Metadata. With message content now strongly protected by E2EE,

adversaries increasingly exploit metadata exposure. Studies on sealed-sender mechanisms demonstrate that timing and delivery receipts can reveal communication links despite encrypted content.

From Implementation/Deployment to Human Factors. Vulnerabilities that once originated in implementation and deployment are now compounded by user-level manipulation through phishing threats. Research on phishing kits highlights how adversaries exploit trust and immediacy within messaging platforms rather than protocol design.

From an Application to the Ecosystem. The scope of security challenges has expanded from a single application to the ecosystem surrounding it. Analyses of unofficial or modified clients reveal that additional features often come at the cost of weakened provenance and broader permission.

V. Future Work

Building on evolving attack tactics, we propose four directions to strengthen the security of online messengers. First, composition methods require to be verified for consistent session and group management. Second, practical metadata-protection techniques should be deployed to balance privacy and usability. Third, verification on implementation is needed to close the gaps between formal models and deployed code. Finally, user-level defenses and ecosystem integrity should be reinforced through trusted distribution and permission control.

VI. Conclusion

This survey examines attacks on secure messaging systems across five dimensions: protocol pitfalls, implementation errors and insecure deployment, metadata exposure, phishing threats, and ecosystem risks. Over time, the attack tactics have evolved from primitives to composite protocols, from content to metadata, from implementation/deployment to human factors, and from an application to the ecosystem. Hence, in the future, we need to focus on reliable session and group management, metadata-leakage mitigations, safer in-app user protections, and secure ecosystem distribution.

[References]

[1] Balbás, D., Collins, D., & Vaudenay, S. (2023). Cryptographic administration for secure group messaging. In Proceedings of the 32nd USENIX Security Symposium.

[2] Cremers, C., Jacomme, C., & Naska, A. (2023). Formal Analysis of Session-Handling in Secure Messaging: Lifting Security from Sessions to Conversations. In Proceedings of the 32nd USENIX Security Symposium.

[3] Kobeissi, N., Bhargavan, K., & Blanchet, B. (2017). Automated verification for secure messaging protocols and their implementations: A symbolic and computational approach. In Proceedings of the 2017 IEEE European Symposium on Security and Privacy (EuroS&P).

[4] Martiny, I., Kaptchuk, G., Aviv, A., Roche, D., & Wustrow, E. (2021). Improving Signal's Sealed Sender. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium.

[5] Bijmans, H., Booij, T., Schwedersky, A., Nedgabat, A., & Wegberg, R. (2021). Catching phishers by their bait: Investigating the Dutch phishing landscape through phishing kit detection. In Proceedings of the 30th USENIX Security Symposium.

[6] Munyendo, C., Owens, K., Strong, F., Wang, S., Aviv, A., Kohno, T., & Roesner, F. (2025). "You Have to Ignore the Dangers": User Perceptions of the Security and Privacy Benefits of WhatsApp Mods. In Proceedings of the 2025 IEEE Symposium on Security and Privacy (S&P).