VEILCOMM SECURE MESSAGING PROTOCOL (VCSP)

Authenticated ECDH-Based Hybrid Encryption using AES-256-GCM

A Comprehensive Technical Whitepaper on Modern Cryptographic Architecture

Document Type	Technical Whitepaper
Version	1.0
Classification	Confidential
Protocol	VCSP

Abstract

This document presents the VeilComm Secure Messaging Protocol (VCSP), a cryptographic framework built on Authenticated Elliptic Curve Diffie-Hellman (ECDH) key agreement, HKDF-based key derivation, and AES-256-GCM authenticated encryption. VCSP achieves forward secrecy through ephemeral key pairs, eliminates MAC complexity via AEAD primitives, and aligns architecturally with industry standards including the Signal Protocol, WhatsApp, and TLS 1.3. This whitepaper details the complete cryptographic flow, security properties, and comparative analysis against traditional ECIES.

TABLE OF CONTENTS

1. High-Level Protocol Flow
2. Sender-Side Operations
3. Receiver-Side Operations
4. Transmission Payload
5. Detailed Cryptographic Flow
6. ECIES vs VCSP — Comparative Analysis
7. Industry Alignment
8. Security Properties

9. Final Statement

1. HIGH-LEVEL PROTOCOL FLOW

VCSP operates as a hybrid encryption scheme. The sender generates an ephemeral EC key pair, computes a shared secret via ECDH with the receiver's static public key, derives a symmetric encryption key through HKDF, and encrypts the payload with AES-256-GCM. The receiver reverses this process using their private key to recover the shared secret, derive the same symmetric key, and decrypt and verify the ciphertext.

2. SENDER-SIDE OPERATIONS

1 Ephemeral Key Generation
Generate ephemeral EC key pair (sk_E, pk_E)
Elliptic curve: X25519 or P-256

2 Key Agreement (ECDH)
Compute shared secret S using receiver's static public key (pk_R)
$S = ECDH(sk_E, pk_R)$

3 Key Derivation (HKDF)
Derive 256-bit symmetric key K
K = HKDF(S, salt, info)

4 Authenticated Encryption
Encrypt payload with AES-256-GCM using key K
Output: IV, Ciphertext, GCM Tag

5 Package Message
Assemble transmission bundle
Send: pk_E, IV, Ciphertext, GCM Tag

3. RECEIVER-SIDE OPERATIONS

1 Receive Message
Receive pk_E, IV, Ciphertext, GCM Tag

2 Key Agreement (ECDH)
Compute shared secret using own private key (sk_R)
$S = ECDH(sk_R, pk_E)$

3 Key Derivation (HKDF)
Derive 256-bit symmetric key K
K = HKDF(S, salt, info)

4 Authenticated Decryption
Decrypt & verify using AES-256-GCM with key K, IV and GCM Tag
If tag valid → plaintext recovered
Else → reject (tampered)

5 Output
Original plaintext message recovered

4. TRANSMISSION PAYLOAD

pk■ — Ephemeral Public Key
The sender's ephemeral EC public key. Enables the receiver to perform ECDH key agreement and reproduce the shared secret S.

IV — Initialization Vector
A random nonce used by AES-256-GCM. Must be unique per encryption operation to prevent keystream reuse.

Ciphertext

The AES-256-GCM encrypted payload. Confidentiality is guaranteed provided K and IV remain secret and unique.

GCM Authentication Tag

A 128-bit MAC produced by GCM. Verifies both integrity and authenticity of the ciphertext; any tampering causes tag verification failure and rejection.

5. DETAILED CRYPTOGRAPHIC FLOW

The following describes the complete six-stage cryptographic pipeline executed for every message transmitted via VCSP.

1	EC Key Gen	**Ephemeral Key Generation**	Generate (sk_E, pk_E) on elliptic curve (e.g., X25519 / P-256).
2	ECDH	**ECDH Key Agreement**	S = ECDH(sk_E, pk_R) [Sender] S = ECDH(sk_R, pk_E) [Receiver]
3	HKDF	**Key Derivation (HKDF)**	K = HKDF(S, salt, info) → 256-bit symmetric key
4	AES-GCM	**AES-256-GCM Encrypt**	Input: Plaintext, K, IV, AAD Output: Ciphertext, GCM Tag
5	Network	**Transmission**	Send: pk_E, IV, Ciphertext, GCM Tag over network
6	Verify	**Decrypt & Verify**	Verify GCM Tag If valid → Decrypt Else → Reject

6. ECIES vs VCSP — COMPARATIVE ANALYSIS

Aspect	Traditional ECIES	This Implementation (VCSP)
Key Agreement	ECDH	ECDH
Key Derivation	KDF (e.g., HKDF)	KDF (e.g., HKDF)
Encryption	Symmetric Encryption (e.g., AES)	Symmetric Encryption (e.g., AES)
Authentication	Separate MAC (e.g., HMAC-SHA256)	**AEAD (AES-GCM) — Built-in Authentication**
Auth Method	Encrypt-then-MAC	**AEAD (Single Primitive)**
Complexity	Higher (extra MAC layer)	**Lower (simpler)**
Security	Secure	**Secure (Modern AEAD)**
Error Prone	Higher risk (MAC misuse possible)	**Lower risk (AEAD misuse-resistant)**
Performance	Slightly more overhead	**More efficient**
Industry Usage	Legacy systems	**Modern protocols (Signal, TLS 1.3, etc.)**

7. INDUSTRY ALIGNMENT

VCSP's architectural pattern is consistent with the world's most widely deployed secure messaging systems. The following shows alignment across three major implementations:

S

Signal Protocol

End-to-end encrypted messaging. Uses ephemeral ECDH, HKDF key derivation, and AES-GCM or ChaCha20-Poly1305 for AEAD encryption.

W

WhatsApp (via Signal Protocol)

Implements Signal Protocol wholesale. Provides identical architectural guarantees: forward secrecy, AEAD encryption, and HKDF-based key schedules.

T

TLS 1.3

The latest Transport Layer Security standard. Mandates ephemeral ECDH for key exchange and AEAD ciphers (AES-GCM / ChaCha20-Poly1305). Eliminates legacy MAC constructs.

Common Architectural Pattern

✓ **Ephemeral ECDH**	Forward Secrecy — each session uses a fresh key pair
✓ **HKDF-based Key Derivation**	Deterministic, secure stretching of shared secrets
✓ **AEAD Encryption**	AES-GCM or ChaCha20-Poly1305 — combined confidentiality + integrity

8. SECURITY PROPERTIES

C

Confidentiality — AES-256-GCM Encryption

The payload is encrypted with AES-256-GCM. With a 256-bit key, brute-force attacks are computationally infeasible under current and foreseeable quantum threat models.

I

Integrity — GCM Authentication Tag

The 128-bit GCM tag authenticates both the ciphertext and any associated data (AAD). Any single-bit modification to the ciphertext causes tag verification failure.

F

Forward Secrecy — Ephemeral ECDH Keys

A fresh ephemeral EC key pair is generated per session. Compromise of any long-term key does not expose past session ciphertexts.

R

Replay / Tamper Resistance — Authentication Failure on Modification

The GCM tag binds IV, ciphertext, and AAD together. Replayed or modified messages are detected and rejected during tag verification.

9. FINAL STATEMENT

This system implements authenticated ECDH-based hybrid encryption using HKDF-derived AES-256-GCM, aligned with modern AEAD-based cryptographic design principles used in contemporary secure communication systems.

"These systems use the same architectural pattern (ECDH + HKDF + AEAD), though with additional protocol layers."

Key Takeaways

- VCSP achieves equivalent security to ECIES while reducing complexity through AEAD primitives.
- Ephemeral ECDH guarantees forward secrecy — past sessions remain secure even after key compromise.
- AES-256-GCM replaces the separate MAC construction of traditional ECIES with a single, misuse-resistant primitive.
- The architecture mirrors Signal Protocol, WhatsApp, and TLS 1.3 — the dominant secure messaging standards.
- The GCM authentication tag provides tamper and replay resistance without additional protocol overhead.

VeilComm Technologies · VCSP Whitepaper v1.0 · Authenticated ECDH-Based Hybrid Encryption using AES-256-GCM · *CONFIDENTIAL*