

SecureStream 256 Pro — Technical Datasheet

Document version: 2.3 • Date: 30.09.2025

Software version: 2.3 • File format: ENCV2 / SGCM2 / SGCM2F / EMETA2

Author: Paweł Pawlikowski • Source: www.securestream256pro.com • Contact: support@securestream256pro.com

Cryptographic suite (summary)

Data algorithm: AES-256-GCM, 128-bit tag.

KDF (password): Argon2id; defaults t=5, m=256 MiB, p=6; salt 16 B.

File-key (alternative mode): 32-byte key (exactly 32 bytes).

HKDF — key separation & labels (exact values from code)

HKDF_INFO_AEAD = "SecureStream256Pro AEAD key" — base AEAD key.

HKDF_INFO_KCV = "SecureStream256Pro KCV key" — KCV key (early check).

HKDF_INFO_FILEKEY = "SecureStream256Pro master from filekey" — master from file-key.

HKDF_INFO_FILEMAC = "SecureStream256Pro file-mac key" — per-file MAC key.

Per-file AEAD: salt = BLAKE2s(32) over header_bytes || nonce_prefix; info="SecureStream256Pro AEAD per-file".

KCV (Key Check Value)

Early key verification (immediate abort on wrong password/key).

KCV length = 32 bytes.

Stream and AAD

GCM Nonce: 12 B = 8-byte prefix + 4-byte counter; unique per chunk.

AAD (common + per-chunk):

AAD_common = header_bytes || "SGCM2" || chunk_bytes || nonce_prefix

AAD = AAD_common || chunk_index

Header / Footer

ENCV2 header: includes MAGIC, mode (password/file-key + meta flag), KDF=Argon2id, name/ext lengths, t/m/p, salt, KCV; full header_bytes reused as AAD.

SGCM2F footer: total_chunks (uint32), total_plain (uint64), BLAKE2s(32) over all ciphertexts, and HMAC-SHA256 (truncated to 32 B) over AAD_common || total_chunks || total_plain || digest. Footer verified before atomic commit.

Metadata & privacy

Hide-metadata mode: original name/extension stored in EMETA2 (internal META, up to 65 535 B).

Privacy: fully offline; no telemetry. Local logs contain no secrets.

Filtering: skips symlinks, junctions, and tmp_secure / backup directories.

I/O and consistency

Atomic write: temp file → fsync → os.replace → fsync(dir) after footer verification.

TMP directory: hidden, auto-cleaned on program start.

Backup (optional): executed per configuration; errors don't abort the operation.

Unicode: full support (emoji, complex scripts).

Limits & behavior (Windows / general)

Chunk size: up to 256 MiB (MAX_CHUNK_BYTES).

Max chunks: $2^{32} - 1$.

Max plaintext: 2 TiB (strict limit enforced during prescan and streaming).

Windows path budget: ~240 characters; names are truncated and enumerated on collision.

Secure overwrite: none (intentional; ineffective on SSD/flash).

Binary checksum (for integrity verification)

SHA-256 (binary file):

F75104844460F74EC476F2960FC1A3EA48877033579E0F30A5EFE653BD068F50 SecureStream 256 Pro EN.exe

"How to verify": PowerShell: (Get-FileHash -Algorithm SHA256 'path\SecureStream 256 Pro.exe').Hash