Network Working Group
Internet-Draft
Intended status: Experimental
Expires: September 18, 2026

Z. Eli
Independent Researcher

March 17, 2026

StealthFlow Protocol (SFP)
draft-eli-stealthflow-protocol-01

Abstract

This document specifies the StealthFlow Protocol (SFP), a pre-transport security layer designed to operate before TLS or QUIC. SFP introduces a zero-handshake mechanism with reduced observability and asymmetric cost properties to mitigate denial-of-service attacks and traffic analysis, while remaining compatible with existing PKI infrastructure.

1. Introduction

Traditional TLS deployments expose identifiable handshake patterns (e.g., ClientHello, SNI, ALPN), which can be exploited by DPI systems and adversarial observers. Additionally, handshake asymmetry enables low-cost denial-of-service attacks.

SFP addresses these issues by introducing:

- A single-packet initiation model (zero handshake)
- Proof-of-Work (PoW) based admission control
- Low-observability packet structure
- Compatibility with existing PKI and TLS

SFP is not a replacement for TLS; it acts as a pre-layer.

2. Design Goals

The protocol is designed with the following objectives:

- Zero explicit negotiation (no handshake rounds)
- Minimal network observability
- Asymmetric cost enforcement (attacker > defender)
- Compatibility with existing PKI
- Fail-silent behavior
- Strong integrity protection

3. Threat Model

SFP assumes:

- Advanced DPI and traffic analysis capabilities
- Active probing and packet injection
- High-concurrency denial-of-service attacks
- Partial network manipulation

Out of scope:

- Fully compromised endpoints
- Global PKI failure

4. Protocol Overview

The client sends a single "armored" packet containing:

- Dynamic guide identifier
- Ephemeral public key
- Proof-of-Work
- Timestamp
- Nonce
- Packet hash
- Signature
- Random padding

The server validates the packet statelessly and either:

- Silently drops invalid traffic
- Responds with a signed encrypted reply

TLS or QUIC proceeds only after successful validation.

5. First Packet Format

The first packet consists of:

1. Dynamic Guide Identifier (816 bytes)
2. Client Ephemeral Public Key (32 bytes, X25519)
3. Proof-of-Work:
    - SHA-256 hash with leading zeros D
    - Ed25519 signature (64 bytes)
4. Timestamp (8 bytes, millisecond precision)
5. Nonce (1632 bytes)
6. Whole Packet Hash (32 bytes, SHA-256)
7. Signature (Ed25519 over all above fields)
8. Random Padding (0256 bytes)

6. Proof-of-Work

PoW is defined as:

```
PoW = SHA256(IP || Timestamp || Nonce || C_eph_pub)
```

Constraint:

```
LeadingZeroBits(PoW)  D
```

Default difficulty:

D = 20 bits

The server MAY increase difficulty dynamically.

Timestamp MUST fall within an acceptable window of server time.

7. Server Processing

Upon receiving a packet:

1. Detect guide identifier
2. Extract client ephemeral public key
3. Verify signature
4. Verify packet hash
5. Validate PoW
6. Validate timestamp window
7. Optionally check Nonce uniqueness

On failure:

- Packet MUST be silently dropped

On success:

- Server generates a response

8. Response Packet

The response contains:

1. Server Public Key (X25519, plaintext)
2. Encrypted Payload:
   - Certificate chain
   - Session parameters
3. Whole Response Hash (SHA-256)
4. Server Signature (Ed25519)

Encryption uses a key derived via X25519.

9. Cryptographic Algorithms

The following algorithms are REQUIRED:

- Hash: SHA-256
- Signature: Ed25519
- Key exchange: X25519
- AEAD: ChaCha20-Poly1305

AES-128-GCM MAY be supported.

10. Security Properties

SFP provides:

- Resistance to traffic analysis
- Cost asymmetry via Proof-of-Work
- Replay protection via timestamp and nonce
- Integrity via full-packet signatures
- Mitigation of man-in-the-middle attacks via PKI

11. Integration with TLS/QUIC

SFP operates as a pre-layer:

- TLS handshake begins only after SFP success
- QUIC MAY use SFP as a UDP pre-filter

Existing TLS implementations require no modification.

12. Deployment Considerations

Deployment can be incremental:

- Internal infrastructure
- Public APIs
- High-value endpoints

Hardware acceleration (e.g., XDP, eBPF, SmartNIC) is OPTIONAL.

13. Limitations

- Does not hide IP addresses
- Does not prevent endpoint compromise
- Relies on existing PKI trust model
- May be subject to long-term traffic analysis

14. IANA Considerations

This document has no IANA actions.

15. Security Considerations

Security is the primary focus of this protocol.
Improper implementation of timestamp validation,
nonce handling, or signature verification may
weaken guarantees.

16. Informative References

[RFC8446] Rescorla, E., "The Transport Layer Security (TLS) Protocol Version 1.3", RFC 8446, DOI 10.17487/RFC8446, August 2018, <https://www.rfc-editor.org/info/rfc8446>.

Authors’ Addresses

Z. Eli
Email: li.xiaoming@tutamail.com