TCP-ENO: Encryption Negotiation Option

A. Bittau, D. Boneh, D. Giffin, M. Handley, D. Mazières, E. Smith
Stanford University and University College London

November 4, 2015
IETF 93 (and 92, and...) 

- Do we do tcpcrypt? Do we do TLS? Do we do TCP AO-Encryption? Do we do tcpcrypt? Do we do TLS? Do we do tcpcrypt? ...

- What’s clear:
  - Need a mechanism to signal encryption support.
  - Security requirements: Session ID for application-level authentication, forward secrecy, etc.
  - Provide fixed target for APIs across encryption specs.
TCP ENO: a framework for bootstrapping security protocols

SYN, ENO(tcpcrypt, TLS, X)
Do you support tcpcrypt? TLS? X?

(plain) SYN ACK
plain-text connection

Or:
SYN-ACK, ENO(tcpcrypt)
encrypted connection
Architecture: three layers

Encryption spec (e.g., tcpcrypt)
Setup and perform encryption.

Session ID

Application (optional)
perform authentication

ENO
Negotiate encryption spec

ENO transcript

SYN, ENO(tcpcrypt)

SYN-ACK, ENO(tcpcrypt)

ACK, ENO, INIT1
INIT2

sign(Session ID, …)

sign(Session ID, …)
TCP ENO: a framework for bootstrapping security protocols

**SYN-ENO**

- Encryption: Does the kernel support tcpcrypt (or TLS, or X)?
- Authentication: Does the application support ENO?
  - Disable ENO if the application doesn’t support it?

- (plain) SYN ACK - plain-text
- ENO SYN ACK - authenticated encryption
- ENO (app-support) SYN ACK - auth. encryption and app-awareness
TCP ENO option

**ENO (type)**

**Length**

**General suboption (Optional)**

**Encryption spec ID 1 (Optional)**

**Encryption spec ID … (Optional)**

```
000000 A AB

V S S S S S S S S
```

A: 2 application aware bits (aware, mandatory)
B: simultaneous open tie-breaker
V: 1 if suboption spans multiple bytes (must be last)
S: 7 bits to specify encryption protocol ID
ENO handshake

Output: ENO transcript - concatenation of all ENO SYN options, separated by empty ENO options.
Abstracting TCP-level encryption for applications

• Expose a Session ID.
  - First byte must be spec ID. Min 33 bytes total.
  - Must depend on fresh data from both client and sever, depend on public DH parameters, and ENO transcript.
  - Must not be confidential and must be random.

• Authenticated encryption.

• Forward secrecy.

• Protect end-of-file marker (FIN).
Open Issues

• Address mailing list feedback (M. Scharf).
  • Make option kind sharing non-normative.
• Is current support for simultaneous open adequate?
• Extractors for quantities other than Session ID?