Replacing Weary Crypto: Upgrading the I2P network with stronger primitives

str4d
https://geti2p.net
str4d@i2pmail.org
@str4d

2016-01-08
Tor and I2P have several similarities...

- Both started circa 2003

- Location anonymity
  - Onion routing

- Low-latency
  - Vulnerability to traffic confirmation attacks!
… but also significant differences

Tor
- Centralized*
- Asymmetric design
  - ~8,000 relays
  - Millions of users
- TCP
- Bidirectional tunnels

I2P
- Decentralized*
- Symmetric design
  - ~40,000 routers
- TCP, UDP, RAW, ...
- Unidirectional tunnels
Tunnel layout
I2P uses three layers of crypto

Outbound

Inbound

Application

Destination

Cryptography

RouterInfo
Link encryption

NTCP (2006) - TCP
- 2048-bit DH
- 2-way auth
- AES-256/CBC with last 16 bytes of prev. message as IV

SSU (2005) - UDP
- 2048-bit DH
- 2-way auth
- AES-256/CBC with random IV and MAC (HMAC-MD5-128*)
Tunnel encryption

AES-256/CBC + truncated SHA256

Packet: 4-byte Tunnel ID + 16-byte IV + Ciphertext

IV encrypted before and after each hop with AES-256/ECB (ie. one block)
End-to-end encryption

ElGamal/AES+SessionTags

First packet:
- 514-byte
  ElG(PK_B, (sk, pre-IV))
- AES-CBC(sk, SHA256(pre-IV)[:16], (list of 32-byte nonces + payload))

Subsequent packets:
- 32-byte nonce
- AES-CBC(sk, SHA256(nonce)[:16], payload)
Original primitives

- **ElGamal-2048**
  - Using Oakley primes
  - Use short exponent [1] on non-(64-bit x86) hardware

- **DSA-1024**

- **AES-256/CBC**

- **SHA256**

- **Non-standard HMAC-MD5-128** (only for SSU)

[1] On Diffie-Hellman Key Agreement with Short Exponents - van Oorschot, Weiner at EuroCrypt 96
We have good update propagation

- Automatic in-net updates since 2009
- Via in-net torrents since 2013/14
Legacy data structures...

Destination

<table>
<thead>
<tr>
<th>PK (unused)</th>
<th>SPK</th>
<th>Cert</th>
</tr>
</thead>
<tbody>
<tr>
<td>256B</td>
<td>128B</td>
<td>1B 2B</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PK

LeaseSet

<table>
<thead>
<tr>
<th>Dest</th>
<th>PK</th>
<th>SPK</th>
<th>Leases</th>
</tr>
</thead>
<tbody>
<tr>
<td>256B</td>
<td>128B</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Revocation (unused)

RouterIdentity

<table>
<thead>
<tr>
<th>PK</th>
<th>SPK</th>
<th>Cert</th>
</tr>
</thead>
<tbody>
<tr>
<td>256B</td>
<td>128B</td>
<td>1B 2B</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

RouterInfo

<table>
<thead>
<tr>
<th>RId</th>
<th>Date</th>
<th>Addresses</th>
<th>Options</th>
<th>Sig</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Length</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
</tbody>
</table>
Don't break third-party software!
We now have full flexibility for future key types (up to 64,000 each, 7 SPK defined)
(Relatively) good uptake

<table>
<thead>
<tr>
<th>Type</th>
<th>Usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSA_SHA1</td>
<td>73%</td>
</tr>
<tr>
<td>ECDSA_SHA256_P256</td>
<td>6%</td>
</tr>
<tr>
<td>EdDSA_SHA512_Ed25519</td>
<td>21%</td>
</tr>
</tbody>
</table>
We get router key upgrades for free!

- Can change signing and encryption type
  - (becomes “new” router)
- But no backup for routers without support for new types
  → Cut backwards compatibility
RI signature upgrade is rolling out

0.9.22

0.9.23
We are continuing the migration

- E2E crypto: LeaseSet has no free bits → LS2
  - Easy to handle, doesn't change address
  - Take opportunity to redesign both netDb and LS

- NTCP is very identifiable → NTCP2
  - Based on nTor? Ace?
  - We require 2WAKE

Design help appreciated!