Post Quantum Noise

Yawning Angel  Benjamin Dowling  Andreas Hülsing  Peter Schwabe
Florian Weber

(author-list in alphabetical order)
Noise Protocol Framework

- Framework for Key Exchange Protocols
- Users include: WhatsApp, Wireguard, Lightning, I2P
- Diffie-Hellman based → not Quantum-safe
Noise

- Static and Ephemeral keys
- Fed into a hash-chain
- Hash-chain produces keys for symmetric encryption
- Messages can be sent early (with reduced security)

Patterns
- Used to describe specific exchanges
- \( s \rightarrow \) static key, \( e \rightarrow \) ephemeral key.
- Textual Format: For example:
  \[ XX: \]
  \[
  \begin{align*}
  &\rightarrow e \\
  &\leftarrow e, ee, s, es \\
  &\rightarrow s, se
  \end{align*}
  \]
Goal: Same Security as Noise, but in a Post-Quantum-Setting

Idea: Replace the DH-key-exchanges with KEMs
- e and s works as before, sending KEM-PKs
- ekem sends an unencrypt KEM-ciphertext for e.
- skem sends an, if possible encrypted, KEM-ciphertext for s.
Goal: Same Security as Noise, but in a Post-Quantum-Setting

Idea: Replace the DH-key-exchanges with KEMs

- e and s works as before, sending KEM-PKs
- $ekem$ sends an unencrypt KEM-ciphertext for e.
- $skem$ sends an, if possible encrypted, KEM-ciphertext for s.

Problem: DH allows for non-interactive KX (NIKE)
Problem: DH creates bidirectional authenticity
Problem: DH keys can be freely combined.
Some cases are trivial.
- “ee”, “→ es”, “← se”
- → “ekem”, “→ skem”, “← skem”.

Some are challenging
- “← es”, “→ se”
- switch parties and send the other way.
- potentially adds a roundtrip

Some are “impossible”
- “ss” (combination of two long-term keys)
- Cheat! (Next Slide)

Works but not always optimal.

⇒ We also provide (conjectured) optimal solutions for all standard patterns.
Static-Ephemeral Entropy-Combination (SEEC)

- Strengthen ephemeral randomness with static secret.

- Established: “NAXOS-trick”, “Twisted PRF-trick”, RFC 8937
  - Previous formalizations implicit or tied to instantiation.

- Intentionally weak notion to cover existing schemes.
  - Generic analysis with (insecure) identity possible
Analysis

- fACCE-model.
  - Used by previous analysis of Noise.

- Analyse Hash-chains as “Pseudorandom Hashobject”
  - Noise uses final state as output.
    → “Noise Pseudorandom Hashobject”
  - Allows for generic proof of all patterns in one go

- Previous Noise-analysis limited to specific patterns.
  - We match all proven and conjectured claims.

- KEMs all treated separately.
  - Mixed-KEM-hybrids are covered (compare PQWireGuard)
  - Applicable to Classic-Noise+PQNoise-hybrids
The following (generic!) statements also apply for non-composite hybrid patterns:

**Ephemeral KEM**
- All messages sent after $ekem$ are confidential, if:
  - Both ephemeral keys are uncorrupted. ($\rightarrow$ Forward Secrecy)

**Initiator/Responder KEM**
- All messages sent after $skem$ are confidential, if:
  - The sender’s ephemeral and the receivers static keys are uncorrupted.
- The sender is authentic, if s/he can continue for one more roundtrip, if:
  - The sender’s ephemeral and the receivers static keys are uncorrupted.

**SEEC**
If a party uses SEEC, an uncorrupted static key can act as an uncorrupted ephemeral key.
Performance

- **Implement in Nyquist**
- **Using Kyber-768 (Level 3) as KEM**

<table>
<thead>
<tr>
<th></th>
<th>Initiator (fast NW)</th>
<th>Responder (fast NW)</th>
<th>Initiator (slow NW)</th>
<th>Responder (slow NW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>KK</td>
<td>16.35ms</td>
<td>0.42ms</td>
<td>98.73ms</td>
<td>0.41ms</td>
</tr>
<tr>
<td>PQKK</td>
<td>16.07ms</td>
<td>0.25ms</td>
<td>100.28ms</td>
<td>0.27ms</td>
</tr>
<tr>
<td>XX</td>
<td>16.02ms</td>
<td>16.1ms</td>
<td>98.47ms</td>
<td>98.6ms</td>
</tr>
<tr>
<td>PQXX</td>
<td>31.83ms</td>
<td>16.1ms</td>
<td>199.31ms</td>
<td>100.36ms</td>
</tr>
</tbody>
</table>

- Comparable for patterns that are trivial translations.
- Worse but acceptable for patterns with additional messages.
Thanks for your Attention!

And to Trevor Perrin and Denisa Greconici for many helpful discussions.