Crunching the Numbers: Post Quantum Algorithm Performance

Tomas Gustavsson
Chief PKI Officer at Keyfactor
Crunching the Numbers: The Reality of Quantum Algorithm Performance and Security

Tomas Gustavsson, Chief PKI Officer
Post-Quantum Algorithm Metrics

How does it compare to today's world?
Signature Size

<table>
<thead>
<tr>
<th>Signature Algorithm</th>
<th>Signature Size (bytes)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRYSTALS-Dilithium</td>
<td>2420</td>
</tr>
<tr>
<td>FALCON-512</td>
<td>666</td>
</tr>
<tr>
<td>SPHINCS+128s</td>
<td>7856</td>
</tr>
<tr>
<td>HSS/LMS</td>
<td>2964</td>
</tr>
<tr>
<td>XMSS^MT</td>
<td>4963</td>
</tr>
<tr>
<td>ECDSA-256</td>
<td>64</td>
</tr>
<tr>
<td>RSA-2048</td>
<td>256</td>
</tr>
</tbody>
</table>

Thanks to Verisign for the graph

\(^{1}\)with example parameters
Compare Apples with Apples

<table>
<thead>
<tr>
<th>Level</th>
<th>Definition, as least as hard to break as...</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>To recover the key of AES-128 by exhaustive search</td>
</tr>
<tr>
<td>2</td>
<td>To find collision in SHA256 by exhaustive search</td>
</tr>
<tr>
<td>3</td>
<td>To recover the key of AES-192 by exhaustive search</td>
</tr>
<tr>
<td>4</td>
<td>To find collision in SHA384 by exhaustive search</td>
</tr>
<tr>
<td>5</td>
<td>To recover the key of AES-258 by exhaustive search</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Security Strength</th>
<th>Symmetric Key Algorithms</th>
<th>FFC (DSA, DH, MQV)</th>
<th>IFC* (RSA)</th>
<th>ECC* (ECDSA, EdDSA, DH, MQV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>128</td>
<td>AES-128</td>
<td>L = 3072</td>
<td>K = 3072</td>
<td>f = 256-383</td>
</tr>
<tr>
<td>192</td>
<td>AES-192</td>
<td>L = 7680</td>
<td>K = 7680</td>
<td>f = 384-511</td>
</tr>
<tr>
<td>256</td>
<td>AES-256</td>
<td>L = 15360</td>
<td>K = 15360</td>
<td>f = 512+</td>
</tr>
</tbody>
</table>

Key sizes
- Not obvious with PQC
- "security strength"; FIPS 800-57
- Solution "security level"
Public Key Size

- SPHINCS+128
- LMS H20
- Dilithium2
- RSA3072
- P-256
- SPHINCS+192
- Dilithium3
- P-384
- SPHINCS+256
- Dilithium5
- P-521
Certificate Size

- SPHINCS+128
- LMS H20
- Dilithium2
- RSA3072
- P-256
- SPHINCS+192
- Dilithium3
- P-384
- SPHINCS+256
- Dilithium5
- P-521
Private Key Size

- SPHINCS+128
- LMS H20
- Dilithium2
- RSA3072
- P-256
- SPHINCS+192
- Dilithium3
- P-384
- SPHINCS+256
- Dilithium5
- P-521
## Size Table

<table>
<thead>
<tr>
<th>ALGORITHM</th>
<th>PUBLIC</th>
<th>PRIVATE</th>
<th>CERTIFICATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SHA256WithRSA 3072</td>
<td>294</td>
<td>1217</td>
<td>1173</td>
</tr>
<tr>
<td>SHA256WithECDSA P-256</td>
<td>191</td>
<td>150</td>
<td>522</td>
</tr>
<tr>
<td>Dilithium2</td>
<td>1336</td>
<td>3902</td>
<td>4123</td>
</tr>
<tr>
<td>SPHINCS+ 128</td>
<td>58</td>
<td>101</td>
<td>8279</td>
</tr>
<tr>
<td>LMS_SHA256_M32_H20</td>
<td>82</td>
<td>164</td>
<td>5389</td>
</tr>
</tbody>
</table>
Speed!
HSM Status

5 HSMs tested with Dilithium

• 3 on Round3 version ✔
• 2 still on Round2 version ❌

(none on FIPS Draft specs)
Certificate Issuance

Software Crypto

BC 1.75

Test

- 10 threads
- 1 minute per CA
- 2 rounds
- Intel Corei7, 1TB SSD, 64GB RAM

<sample command>
Certificate Issuance - Software

- RSA3072
- P-256
- P-384
- P-521
- Dilithium2
- Dilithium3
- Dilithium5
Certificate Issuance

<sample command>

- 50 threads
- 10,000 certificates
Certificate Issuance - Software

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Issuance Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>RSA3072</td>
<td>Zero</td>
</tr>
<tr>
<td>P-256</td>
<td>13%</td>
</tr>
<tr>
<td>P-384</td>
<td>6%</td>
</tr>
<tr>
<td>P-521</td>
<td>8%</td>
</tr>
<tr>
<td>Dilithium2</td>
<td>-13%</td>
</tr>
<tr>
<td>Dilithium3</td>
<td>-14%</td>
</tr>
<tr>
<td>Dilithium5</td>
<td>-20%</td>
</tr>
</tbody>
</table>
HSM Signatures

15 threads
60 seconds
Network over cloud – high latency: +-10%

Signing Speed – HSM 1

- RSA3072
- P-256
- P-384
- P-521
- Dilithium2
- Dilithium3
- Dilithium5

Zero

-29%

-29%

-20%

-86%

-87%

-91%
Signing Speed – HSM 2

- **RSA3072**: 0% decrease
- **P-256**: 7% decrease
- **P-384**: -17% decrease
- **P-521**: -20% decrease
- **Dilithium2**: -13% decrease
- **Dilithium3**: -41% decrease
- **Dilithium5**: -55% decrease

Network over cloud – high latency: ±10%
Local installed – low (no) latency

Signing Speed – HSM 3

- RSA3072: Zero
- P-256: 671%
- P-384: 216%
- P-521: 55%
- Dilithium2: 291%
- Dilithium3: 286%
- Dilithium5: 192%
The other are "normal"

BC 1.76
Stateful Hash Based Signature Algorithms (SHBS)

<table>
<thead>
<tr>
<th>LMS TREE TYPE</th>
<th>HEIGHT</th>
<th>SIGNATURES</th>
<th>KEY GEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>LMS_SHA256_M32_H5</td>
<td>5</td>
<td>32</td>
<td>Fast (ms)</td>
</tr>
<tr>
<td>LMS_SHA256_M32_H10</td>
<td>10</td>
<td>1024</td>
<td>Fast (ms)</td>
</tr>
<tr>
<td>LMS_SHA256_M32_H15</td>
<td>15</td>
<td>32,768</td>
<td>Fast (s)</td>
</tr>
<tr>
<td>LMS_SHA256_M32_H20</td>
<td>20</td>
<td>1,048,576</td>
<td>Slow (m)</td>
</tr>
<tr>
<td>LMS_SHA256_M32_H25</td>
<td>25</td>
<td>33,554,432</td>
<td>Unbearable (h)</td>
</tr>
</tbody>
</table>
Ok, so what does this mean to me?

• Signing and verification will not be horribly slow
• Database size
  • 1M certificates – 1GB -> 4GB
  • 1B certificates – 1TB -> 4TB
  • Signed Transactions and Logs?
• Optimizations will come

LMS for firmware signing - no H25 expected (but maybe partitions) - BEWARE
Open Questions?

• Constrained Devices
• Hardware and Software Optimizations
• CloudHSM efficiency
• Which algorithms will be widely used?
• IT Eco Systems
  • How hard will the migration be? MD5 still seen...
Thanks!