Efficient Implementation of Lattice-based Cryptography for Embedded Devices

Tobias Oder
Ruhr-University Bochum

Workshop on Cryptography for the Internet of Things and Cloud 2017
Lattice-based Cryptography

- Set of vectors in $n$-dimensional space define a basis
Lattice-based Cryptography

• Efficiency

• Scalability

• Versatility
  – Encryption
  – Digital signatures
  – Key exchange
  – Advanced constructions (IBE, FHE, ...)

Implementation of Lattice Crypto | Tobias Oder | Ruhr-University Bochum | 09.11.2017
Learning with Errors

Given $A$ and $b = As$

Task: Find $s$

➢ Easy to solve
Learning with Errors

Given $A$ and $b = As$
Task: Find $s$
➢ Easy to solve

Given $A$ and $b = As + e$
Task: Find $s$
➢ Hard problem
Lattice Classes

Standard or random lattices
• Unstructured matrices
• Main Operation: matrix-vector multiplication
Lattice Classes

Standard or random lattices
• Unstructured matrices
• Main Operation: matrix-vector multiplication

Ring or ideal lattices
• Smaller parameters
• Faster implementations
• Smaller implementations
• Main Operation: polynomial multiplication

But less trust in security due to structure!
Idea: Find a trade-off between the advantages of both classes

Main operation: Matrix-vector multiplication

• But matrix elements are polynomials!
## Schemes

### Non-exhaustive list

<table>
<thead>
<tr>
<th>Schemes</th>
<th>Encryption</th>
<th>Signature</th>
<th>Key Exchange</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Standard Lattices</strong></td>
<td>LWE Encrypt</td>
<td>TESLA</td>
<td>Frodo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bai-Galbraith</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>GPV</td>
<td></td>
</tr>
<tr>
<td><strong>Ideal Lattices</strong></td>
<td>Ring-LWE Encrypt</td>
<td>BLISS</td>
<td>„A new hope“</td>
</tr>
<tr>
<td></td>
<td>NTRU Encrypt</td>
<td>GLP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ring-TESLA</td>
<td></td>
</tr>
<tr>
<td><strong>Module Lattices</strong></td>
<td>Kyber</td>
<td>Dilithium</td>
<td>CCA2-secure Kyber</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dilithium-G</td>
<td></td>
</tr>
</tbody>
</table>
## Schemes

*Non-exhaustive list*

<table>
<thead>
<tr>
<th></th>
<th>Encryption</th>
<th>Signature</th>
<th>Key Exchange</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Standard Lattices</strong></td>
<td>LWE Encrypt</td>
<td>TESLA</td>
<td>Frodo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bai-Galbraith</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>GPV</td>
<td></td>
</tr>
<tr>
<td><strong>Ideal Lattices</strong></td>
<td>Ring-LWE Encrypt</td>
<td>BLISS</td>
<td>&quot;A new hope&quot;</td>
</tr>
<tr>
<td></td>
<td>NTRU Encrypt</td>
<td>GLP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ring-TESLA</td>
<td></td>
</tr>
<tr>
<td><strong>Module Lattices</strong></td>
<td>Kyber</td>
<td>Dilithium</td>
<td>CCA2-secure</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dilithium-G</td>
<td>Kyber</td>
</tr>
</tbody>
</table>
Non-exhaustive list

<table>
<thead>
<tr>
<th>Schemes</th>
<th>Encryption</th>
<th>Signature</th>
<th>Key Exchange</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Lattices</td>
<td>LWE Encrypt</td>
<td>TESLA</td>
<td>Frodo</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bai-Galbraith GPV</td>
<td></td>
</tr>
<tr>
<td>Ideal Lattices</td>
<td><strong>Ring-LWE Encrypt</strong></td>
<td>BLISS</td>
<td>„A new hope“</td>
</tr>
<tr>
<td></td>
<td><strong>NTRU Encrypt</strong></td>
<td>GLP</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Ring-TESLA</td>
<td></td>
</tr>
<tr>
<td>Module Lattices</td>
<td>Kyber</td>
<td>Dilithium</td>
<td>CCA2-secure Kyber</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dilithium-G</td>
<td></td>
</tr>
</tbody>
</table>
Implementation on Embedded Devices

• What are the goals?
  – Throughput/latency
  – Code size/area
  – Power/energy
Implementation on Embedded Devices

• What are the goals?
  – Throughput/latency
  – Code size/area
  – Power/energy

• Cross-disciplinary work and interaction between engineers and cryptographers required
  – Parameter selection and design decisions can make schemes more efficient but also weaker
Implementation on Embedded Devices

• What are the goals?
  – Throughput/latency
  – Code size/area
  – Power/energy

• Cross-disciplinary work and interaction between engineers and cryptographers required
  – Parameter selection and design decisions can make schemes more efficient but also weaker

• Cover side-channels
  – Timing, Cache, Simple Power Analysis
  – Differential Power Analysis, EM
• Polynomial multiplication is a major building block for ideal and module lattice-based cryptography

• NTT is a fast Fourier transform in integer rings
  – Polynomial multiplication in $O(n \log n)$ instead of $O(n^2)$
• Polynomial multiplication is a major building block for ideal and module lattice-based cryptography

• NTT is a fast Fourier transform in integer rings
  – Polynomial multiplication in $O(n \log n)$ instead of $O(n^2)$

• Powers of primitive root of unitiy $\omega$ („twiddle factors“) required
  – Stored in tables
  – Computed on-the-fly
• Polynomial multiplication is a major building block for ideal and module lattice-based cryptography

• NTT is a fast Fourier transform in integer rings
  – Polynomial multiplication in $O(n \log n)$ instead of $O(n^2)$

• Powers of primitive root of unity $\omega$ („twiddle factors“) required
  – Stored in tables
  – Computed on-the-fly

• Core operation is a so-called „butterfly“
  – Gentleman-Sande
  – Cooley-Tukey
Gaussian Sampling

Rejection Sampling

Bernoulli Sampling

Cumulative Distribution Table (CDT) Sampling

Knuth-Yao Sampling
CCA2-Security

• Plain Ring-LWE encryption is only secure against chosen-plaintext attackers (CPA)

• Many use cases require security against chosen-ciphertext attackers (CCA)
  – Attacker has access to a decryption oracle

• Generic Fujisaki-Okamoto transform
  – Tweak by Targhi and Unruh for post-quantum security
  – Expensive re-encryption in decryption
Components to be masked in CCA2-secure Ring-LWE

- PRNG/Hash
- NTT
- Sampler
- Encoding/Decoding

See our implementation: ia.cr/2016/1109 together with Tobias Schneider, Thomas Pöppelmann, and Tim Güneysu
Identity-based Encryption (IBE)

- **Demand** for advanced security services (e.g., smart environments)
- **Concept**: Extend asymmetric encryption scheme based on public identifier \(ID_x\) (e.g., given name, MAC, e-mail address, etc.)
IBE Implementation

- Implementation of encryption and decryption of [DPL14] feasible on embedded devices

- Key generation memory-wise and computationally expensive

IBE Implementation

• Implementation of encryption and decryption of [DPL14] feasible on embedded devices

• Key generation memory-wise and computationally expensive

• Cortex-M4 microcontroller
  – Enc/Dec: 6/2 ms

• Spartan6 FPGA
  – Enc/Dec: 80/54 µs

Conclusion

Lattice-based cryptography is practical on embedded devices!

Future Work

• Side-channel security
• Efficient IBE key generation
• More cryptanalysis
Thank You For Your Attention!