Attacks on NIST PQC 3rd Round Candidates

Daniel Apon and James Howe
Why do we care about attacks on PQC?
Welcome to PQC Side-Channel Attacks

This is a talk about side-channel attacks (SCA) against NIST Post-Quantum Cryptography candidates.

NIST has repeatedly stated the importance of SCA and countermeasures:

- From the original NIST PQC call for proposals in 2016:
  “Schemes that can be made resistant to side-channel attacks at minimal cost are more desirable than those whose performance is severely hampered by any attempt to resist side-channel attacks.”

- To the latest PQC summary document (NISTIR 8309):
  “NIST hopes to see more and better data for performance in the third round. This performance data will hopefully include implementations that protect against side-channel attacks, such as timing attacks, power monitoring attacks, fault attacks, etc.”
Attacks on NIST PQC Schemes

A survey of attacks on NIST PQC 3rd Round Candidates
Types of Attacks Considered

- **Classical cryptanalysis** mathematically analyzes a cryptosystem.
- **Timing analysis** exploits variable runtime of an algorithm.
- **Fault attacks** are semi-invasive methods to intentionally induce faults to reveal cryptographic internal states.
- **Simple, differential, correlation power analysis** non-invasively exploits variations in power consumption of a cryptographic algorithm.
- **Electromagnetic attacks** exploit radiation from a cryptographic algorithm.
- **Template attacks** profiles a sensitive device to gain access to the secret.
- **Cold-boot attacks** exploit memory remanence to read data out of a computer’s memory after the computer has been powered off.
- **Countermeasures** protect/hinder attacks via hiding or masking methods.
A Disclaimer to the Survey

The purpose of this talk is to motivate more attacks.

By showing gaps in the state-of-the-art.

We try to remain as unbiased/neutral as possible:
- Presenting papers that directly attack a candidate.
- Thus, we only focus on KEMs and signatures.
- We try to assume implementations are correct.
- Oracle timings attacks [GJN20], etc., *should* be fixed by now.
# A Summary of the Attacks on NIST PQC Candidates

Table 1: Attacks on NIST PQC third round candidates categorized by classical cryptanalysis (CC), static timing analysis (STA), fault attacks (FA), simple power analysis (SPA), advanced (correlation/differential) power analysis (APA), electromagnetic (EM) attacks, template attacks (TA), cold-boot attacks (CB), and countermeasures (CM).

<table>
<thead>
<tr>
<th>PQC Scheme</th>
<th>CC</th>
<th>STA</th>
<th>FA</th>
<th>SPA</th>
<th>APA</th>
<th>EM</th>
<th>TA</th>
<th>CB</th>
<th>CM</th>
<th>References</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Finalists KEMs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Classic McEliece [BCL+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[CCD+20],[LNPS19],[Poi18]</td>
</tr>
<tr>
<td>Kyber [SAB+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[RRB+19],[XPRO20],[RRB20],[RRB20],[PP19],[ADP18], [PV17],[Poi18]</td>
</tr>
<tr>
<td>NTRU [ZCH+19]</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[RRB20],[RRB20],[Ver19],[BDK+20]</td>
</tr>
<tr>
<td>Saber [DKRV19]</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[BD18],[RJH+19],[RBB+19],[MTG19], [MHS+19]</td>
</tr>
<tr>
<td>Dilithium [LDK+19]</td>
<td></td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[Bone20],[PSKH18],[Poi18]</td>
</tr>
<tr>
<td>Falcon [PFH+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[GJN20],[CCD+20]</td>
</tr>
<tr>
<td>Rainbow [DCP+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[GJN20],[CCD+20],[BFM+19],[RRB20],[RRB+19],[HMOR19],[ATT+18]</td>
</tr>
<tr>
<td><strong>Alternatives KEMs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BIKE [ABB+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[GJN20],[CCD+20]</td>
</tr>
<tr>
<td>FrodoKEM [NAB+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[GJN20],[CCD+20],[BFM+19],[RRB20],[RRB+19],[HMOR19],[ATT+18]</td>
</tr>
<tr>
<td>HQC [AAB+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[GJN20],[CCD+20]</td>
</tr>
<tr>
<td>NTRU Prime [BCLv19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[WZW13],[HCY20], [CLN+20],[GJN20]</td>
</tr>
<tr>
<td>SIKE [JAC+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[TPD20],[PSKH18]</td>
</tr>
<tr>
<td><strong>Signs</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GeMSS [CFM+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[TDP20],[PSKH18]</td>
</tr>
<tr>
<td>Picnic [ZCD+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[DN19],[LLM20],[GSE20], [CMP18]</td>
</tr>
<tr>
<td>SPHINCS+ [HBD+19]</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>[TDP20],[PSKH18]</td>
</tr>
</tbody>
</table>
Recent Highlights

Some selected attacks relevant to the NIST PQC standardization project.
“When a side-channel attack fails, what can you still do with it?” – Léo Ducas.

- Lattice reduction (e.g. [ACD+18]) is a common method for deriving security levels.
- [DDGR20] propose a tool¹ to integrate “hints” from side-channels to use in lattice reduction.
- e.g. if you discover via side-channels $\text{HW}(s_0) = 2 \rightarrow s_0 \in \{3, 5\}$, for $s_i \in \{-5, ..., 5\}$.
- There are four types of hints:

  - **Perfect**: $\langle s, v \rangle = l$
  - **Modular**: $\langle s, v \rangle = l \mod k$
  - **Approximate**: $\langle s, v \rangle = l + \epsilon$
  - **Short Vector Hints**: $v \in \Lambda$

- These hints can reduce the BKZ block size, making attacks easier.

¹. https://github.com/lducas/leaky-LWE-Estimator
LWE With Side Information

“When a side-channel attack fails, what can you still do with it?” – Léo Ducas.

This work may have impacts in the future:

- This could potentially affect certifications of cryptographic modules.
- Especially certifiers for Common Criteria, perhaps even FIPS 140-3 and more.
- Some certifiers (for symmetric) may require $2^{80}$, or $2^{100}$ remaining keys after SCA.
- Thus, if this is unsatisfied, the implementation can ‘fail’.

- **But** having this a priori knowledge could help certifications.
- One may now set cryptographic parameters with side channels in mind.
Masked Ring-LWE (à la NewHope) [OSPG18] has large performance differences:

- Overheads in sampling and A2B conversion due to prime modulus.
- Overall, the performance for 1st order protection is 5.7x slower than unprotected.
- Using [BPO+20] this could be reduced further.

The choice between Kyber and Saber may come down to side-channels and masking:

- Differences between these schemes is small, but their moduli differ.
- Masked Saber [Ver19,BDK+20] is 2.5x slower for 1st order protection.
- Recent work [LS19,CHK+20] shows Saber and NTRU can benefit from using NTTs.
This issue is also seen in lattice-based signature schemes:

- Masked Dilithium [MGTF20] 7-9x faster using power-of-two modulus.
- Dilithium 1\textsuperscript{st} order masking 5.6x slower compared to unprotected.
- Masking qTESLA [GR19] also gained efficiencies by changing modulus.

Masking Falcon will be slightly different:

- But now “isochronous” with constant runtime of Gaussian sampling [HPRR20].
- Its use of floating-point arithmetic makes masking an open problem.
- But has been considered before in MPC [ABZS13, GHK+20].
Active Side-Channel Attacks: QuantumHammer (LWEHammer?)

- Didn’t have a large impact on the process, because of a concurrent work -- computational attack on LUOV [DDSVZ20].

Intuition:
1. Profile a victim device (e.g. in the cloud, like AWS) in an offline, pre-processing step.
2. Online physical manipulation of the device while LUOV signature scheme is running. This causes the device to output malformed signatures that are secret key dependent.
3. Computationally recover the secret key given enough malformed signatures.

- Is this more broadly applicable? (Upcoming work, joint NIST PQC + Univ of AR team – “LWEHammer”).
- Rowhammer is an “old attack” (2014); modern cloud architectures ought to have defenses.
- Are the Rowhammer defenses enough in practice? Can Rowhammer be replaced by fault attacks, etc.?
Takeaways for the Future

What can we learn from these attacks?
What can we learn from these attacks?

Implementation complexity will significantly increase with these standards.
- NIST set the focus on ARM Cortex M4 and Xilinx Artix-7.
- Will attacks or countermeasures we’ve seen be as effective on other devices?
- Will we see more attack vectors? Will we need more countermeasures?

Complexity also increased by the large number of fragile/sensitive operations.
- Past attacks highlight many sensitive/fragile operations that can break schemes.
- Even constant-time; e.g. how will ctgrind work with rejection sampling?

In order to learn the relevance of these attacks you should consider your use case.
- Some candidates will not fit on these devices, let alone smart cards.
05
Conclusions
We hope this survey will motivate further evaluations of NIST PQC candidates.

NIST will take these into consideration in their final decisions for standardization.

NIST PQC will add implementation complexities.
- We have seen many novel attacks due to the fragility of some operations.
- Many attacks were also enabled by implementation errors.

We discuss some open questions and implications of the attacks found.

Start your PQC transition now and consider the attacks that will affect your use cases.
Thanks

Does anyone have any questions?

daniel.apon@nist.gov
james.howe@pqshield.com

With thanks to: Elisabeth Oswald, Mélissa Rossi, and Dustin Moody.
References


References


References


References


[HMOR19] James Howe, Marco Martinoli, Elisabeth Oswald, and Francesco Regazzoni. Optimised Lattice-Based Key Encapsulation in Hardware. NIST’s Second PQC Standardization Conference, 2019.

References


References


References


References
