NTNU | Norwegian University of Science and Technology

SIDE-CHANNEL ATTACKS 4: POST-QUANTUM CRYPTO

TTM4205 – Lecture 15

Tjerand Silde

13.10.2025

Contents

Contents

ChipWhisperer Lab

There will be a 4 hour lab session on Friday, next week we go back to 2 hours.

Exercises sessions can be used for any of the assignments, also the CW lab.

Do you prefer an extra guest lecture or an extra exercise / lab session later?

Technical Essay

You should start thinking about groups and topics for the technical essay.

Deadline for proposal is October 31st, but I recommend you to start earlier.

How many are busy with the Advanced Ethical Hacking exam on Nov 17th?

Reference Group Meeting

We will schedule another reference group meeting soon.

Please provide feedback about the course to the members.

Anything in particular you would like us to discuss?

Contents

Black Box Crypto

We design the security of a cryptographic scheme to follow Kerckhoff's principle: if everything about the scheme, except for the key, is known, then the scheme should be secure.

We analyze the scheme mathematically as black-box algorithms that take some (public or secret) input and give some (public or secret) output, and prove it secure concerning the algorithm description and the public data.

However, security depends on your model. In practice, it matters how these algorithms are implemented and what kind of information the *physical* system leaks about the inner workings of the algorithm computing on secret data.

Leakage

- The time it takes to compute...
- The power usage while computing...
- The electromagnetic radiation...
- The temperature variation...
- The memory pattern accessed...
- The sounds your laptop makes...

Exploiting Leakage

- Timing or power traces can leak secret bits
- Fault injection might leak dummy operations
- Differential analysis allow statistical attacks
- The adversary can choose the input (adaptively)
- The secret key might be static and re-used

Attack Categories

- Remote vs physical attacks
- Software and hardware attacks
- Passive vs active attacks
- Invasive vs non-invasive attacks

Preventing Leakage

- Constant time operations and algorithms
- The result must depend on all operations
- Randomize input and/or secrets each time
- Split secrets into random additive shares

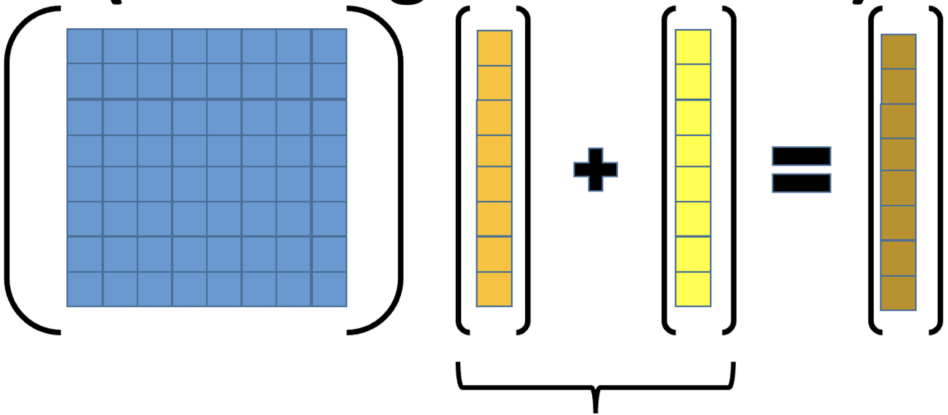
Contents

Learning With Errors (LWE)

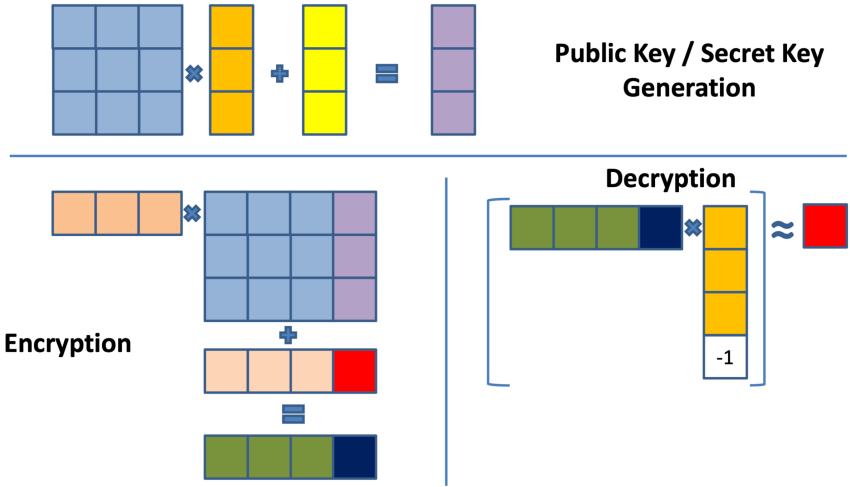
Definition 1. For positive integers m, n, q, and $\beta < q$, the $\mathsf{LWE}_{n,m,q,\beta}$ problem asks to distinguish between the following two distributions:

1. $(\mathbf{A}, \mathbf{As} + \mathbf{e})$, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}, \mathbf{s} \leftarrow [\beta]^m, \mathbf{e} \leftarrow [\beta]^n$
2. $(\mathbf{A}, \mathbf{u})$, where $\mathbf{A} \leftarrow \mathbb{Z}_q^{n \times m}$ and $\mathbf{u} \leftarrow \mathbb{Z}_q^n$.

(Learning With Errors)



Visualization of ML-KEM



ML-KEM KGen and Enc

$$\mathsf{sk}: \mathbf{s} \leftarrow [\beta]^m, \ \mathsf{pk}: (\mathbf{A} \leftarrow \mathbb{Z}_q^{m \times m}, \mathbf{t} = \mathbf{A}\mathbf{s} + \mathbf{e}_1), \text{ where } \mathbf{e}_1 \leftarrow [\beta]^m. \tag{6}$$

To encrypt a message $\mu \in \{0,1\}$, the encryptor chooses $\mathbf{r}, \mathbf{e}_2 \leftarrow [\beta]^m$ and $e_3 \leftarrow [\beta]$, and outputs

$$\left(\mathbf{u}^T = \mathbf{r}^T\mathbf{A} + \mathbf{e}_2^T, v = \mathbf{r}^T\mathbf{t} + e_3 + \left\lceil \frac{q}{2} \right\rfloor \mu\right). \tag{7}$$

Figure: Q: Which operations might leak information?

ML-KEM Dec

To decrypt, one computes $v - \mathbf{u}^{\tilde{T}}\mathbf{s}$. But rather than this cleanly giving us the message μ as in (4), we instead obtain

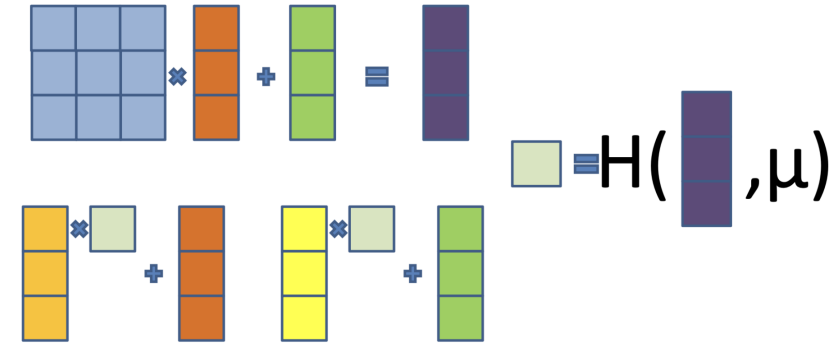
$$v - \mathbf{u}^T\mathbf{s} = \mathbf{r}^T(\mathbf{A}\mathbf{s} + \mathbf{e}_1) + e_3 + \frac{q}{2}\mu - \left(\mathbf{r}^T\mathbf{A} + \mathbf{e}_2^T\right)\mathbf{s} \tag{8}$$

$$= \mathbf{r}^T\mathbf{e}_1 + e_3 + \frac{q}{2}\mu - \mathbf{e}_2^T\mathbf{s} \tag{9}$$

Figure: Q: Which operations might leak information?

Visualization of ML-DSA

Public Key / Secret Key Generation



ML-DSA

Private information: $\mathbf{s}_1 \in [\beta]^m, \mathbf{s}_2 \in [\beta]^n$
Public information: $\mathbf{A} \in \mathcal{R}_{q,f}^{n \times m}, \mathbf{t} = \mathbf{A}\mathbf{s}_1 + \mathbf{s}_2 \in \mathcal{R}_{q,f}^n$

Prover		Verifier
$\mathbf{y}_1 \leftarrow [\gamma + \bar{\beta}]^m$ $\mathbf{y}_2 \leftarrow [\gamma + \bar{\beta}]^n$, $\mathbf{w} := \mathbf{A}\mathbf{y}_1 + \mathbf{y}_2$		
	$\xrightarrow{\mathbf{w}}$	
		$c \leftarrow \mathcal{C}$
	$\xleftarrow{c}$	
$\mathbf{z}_1 := c\mathbf{s}_1 + \mathbf{y}_1$ $\mathbf{z}_2 := c\mathbf{s}_2 + \mathbf{y}_2$ if $\mathbf{z}_1 \notin [\bar{\beta}]^m$ or $\mathbf{z}_2 \notin [\bar{\beta}]^n$ then $(\mathbf{z}_1, \mathbf{z}_2) := \perp$		
	$\xrightarrow{(\mathbf{z}_1, \mathbf{z}_2)}$	
		Accept iff $\mathbf{z}_1 \in [\bar{\beta}]^m$ and $\mathbf{z}_2 \in [\bar{\beta}]^n$ and $\mathbf{A}\mathbf{z}_1 + \mathbf{z}_2 - c\mathbf{t} = \mathbf{w}$

Figure: Q: Which operations might leak information?

Leakage

- Norm leakage from sampling keys or masks
- Hamming weight of secrets multiplied with **A**
- Norm leakage of rejected signature samples
- Fault injection to avoid rejection sampling

LWE Hints

Definition 2.10 (LWE Hints). *Let* $\mathbf{s} \in \mathbb{Z}_q^n$ *be an LWE secret. We define the following* LWE hints *for* $\mathbf{s}$.

1. *A tuple* $\overline{\mathbf{v}} = (\mathbf{v}, \ell) \in \mathbb{Z}^n \times \mathbb{Z}$ *with*

$$\langle \mathbf{v}, \mathbf{s} \rangle = \ell$$

   is called a perfect hint.
2. *A tuple* $\overline{\mathbf{v}} = (\mathbf{v}, \ell, m) \in \mathbb{Z}^n \times \mathbb{Z} \times \mathbb{N}$ *with*

$$\langle \mathbf{v}, \mathbf{s} \rangle \equiv \ell \mod m$$

   is called a modular hint. *If* $m = q$, *we call* $\overline{\mathbf{v}}$ *a* mod-q hint.

LWE Hints

	Kyber 512	Falcon 512	NTRU-HRSS 701	Kyber 768	Dilithium 1024
mod-q	449 (88%)	452 (88%)	622 (89%)	702 (91%)	876 (85%)
Time	20 mins	20 mins	45 mins	35 mins	10 hours
perfect	234 (46%)	233 (46%)	332 (47%)	390 (51%)	463 (45%)
Time	3 hours	3 hours	11 hours	1 day	7 days

Table 1. Minimal amount k of mod-q/perfect hints required for solving instances with LLL. Time includes both lattice basis construction and LLL reduction.

Contents

NIST PQC Submission Requirements

Another case where security and performance interact is resistance to side-channel attacks. Schemes that can be made resistant to side-channel attack at minimal cost are more desirable than those whose performance is severely hampered by any attempt to resist side-channel attacks. We further note that optimized implementations that address side-channel attacks (e.g., constant-time implementations) are more meaningful than those which do not.

Protection Techniques

- Constant time sampling of secret vectors
- Avoid rejection sampling in signatures
- Masking multiplication of secret values

Raccoon DSA

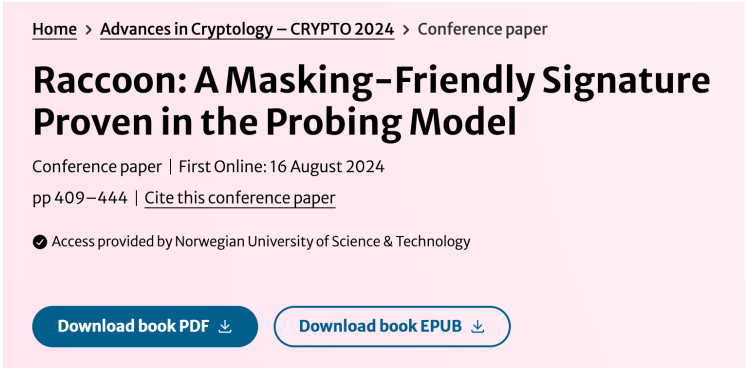


Figure: https://eprint.iacr.org/2024/1291.pdf

Trade-offs PQC DSA

Signature schemes strike a balance between:

- Sizes (verification key and signatures)
- Speed (signing, verification)
- Portability
- Conservative assumptions
- **Resistance against side-channel attacks**

And so on…

Criteria					
Dilithium	★★½	★★★	★★★	★★	
Falcon	★★★	★★★	★★	★★	
SPHINCS+	★½	★★	★★	★★★	
Raccoon	★★	★★★	★★★	★★	★★★

t-Probing Model

t-probing model

- Adversary can probe t circuit values at runtime
- Unrealistic but a good starting point

Masking

- Each sensitive value x is split in d shares:

$$[\![x]\!] = (x_0, x_1, \ldots, x_{d-1}) \quad (1)$$

such that

$$x_0 + x_1 + \cdots + x_{d-1} = x \quad (2)$$

- In t-probing model, ideally 0 leakage if $d > t$
- In "real life", security is exponential in d
- What about computations?

Difficulty of Masking

How difficult are operations to mask?

- 😀 **Addition ($[\![c]\!] = [\![a + b]\!]$)?**
  - Compute $[\![c]\!] = (a_0 + b_0, \ldots, a_{d-1} + b_{d-1})$, simple and fast: $\Theta(d)$ operations
- ☹ **Multiplication ($[\![c]\!] = [\![a \cdot b]\!]$)?**
  - Complex and slower: $\Theta(d^2)$ operations
- 😱 **More complex operations?**
  - Use so-called *mask conversions*, very slow: $\gg \Theta(d^2)$ operations

Masking Dilithium

Dilithium follows the Fiat-Shamir **with aborts** paradigm.

Sign($\mathsf{sk} = \mathbf{s}, \mathsf{vk} = (\mathbf{A}, \mathbf{t}), \mathsf{msg}) \to \mathsf{sig}$

1	Generate a short ephemeral secret $\mathbf{r}$	▷ Slow
2	Compute the commitment $\mathbf{w} = \mathbf{A} \cdot \mathbf{r}$	▷ Fast
3	Compute the challenge $c = \mathsf{H}(\mathbf{w}, \mathsf{msg}, \mathsf{vk})$	▷ No mask
4	Compute the response $\mathbf{z} = \mathbf{s} \cdot c + \mathbf{r}$	▷ Fast
5	Check that $\mathbf{z}$ is in a given interval. If not, restart.	▷ Slow
6	Signature is $\mathsf{sig} = (c, \mathbf{z})$	

Masking bottlenecks:

- Short secret generation (1) requires B2A.
- Rejection sampling (5) requires A2B and B2A.

Total masking overhead: $\Theta(d^2 \log q)$

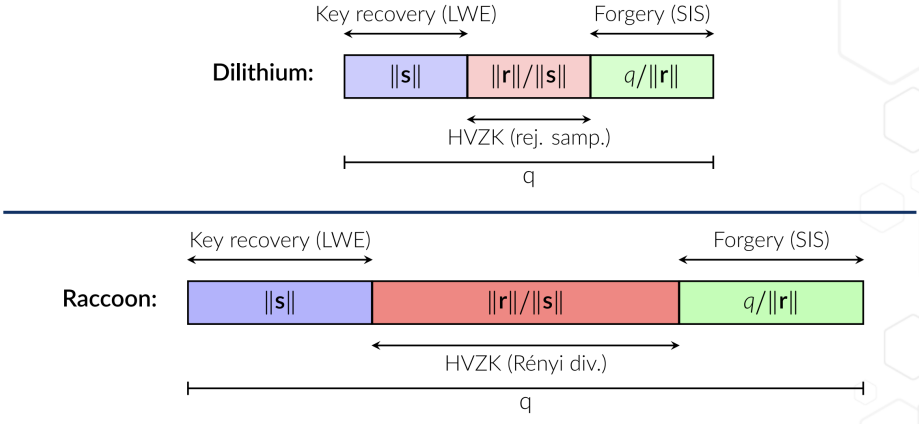
Masking Raccoon

Sign$(\mathsf{sk} = [\![\mathbf{s}]\!], \mathsf{vk} = (\mathbf{A}, \mathbf{t}), \mathsf{msg}) \to \mathsf{sig}$

1. Generate a masked short ephemeral secret $[\![\mathbf{r}]\!]$ using "AddRepNoise" ▷ Fast
2. Compute the commitment $[\![\mathbf{w}]\!] = \mathbf{A} \cdot [\![\mathbf{r}]\!]$ ▷ Fast
3. Unmask $[\![\mathbf{w}]\!]$ to obtain $\mathbf{w}$ ▷ Fast
4. Compute the challenge $c = \mathsf{H}(\mathbf{w}, \mathsf{msg}, \mathsf{vk})$ ▷ No mask
5. Compute the response $[\![\mathbf{z}]\!] = [\![\mathbf{s}]\!] \cdot c + [\![\mathbf{r}]\!]$ ▷ Fast
6. Unmask $[\![\mathbf{z}]\!]$ to obtain $\mathbf{z}$ ▷ Fast
7. (No more rejection sampling!)
8. Signature is $\mathsf{sig} = (c, \mathbf{z})$

Total masking overhead: $O(d \log d)$

Impact on Modulus



1. Removing rejection sampling increases $\|\mathbf{r}\|/\|\mathbf{s}\|$ from $\Theta(\dim \mathbf{s})$ to $\Theta\left(\|c\|\sqrt{\text{Queries}}\right)$
2. The increased q in turn requires increasing $\|\mathbf{s}\|$, $q/\|\mathbf{r}\|$ and/or the dimensions.

Comparison

Raccoon is a specific-purpose scheme aimed at high side-channel resistance:

- ☺ Same assumptions as Dilithium
- ☺ Simpler
- ☺ Verification key size is similar
- ☹ Signature is 4x larger
- 🤩 **When masked, orders of magnitude faster than other schemes are**

Comparison

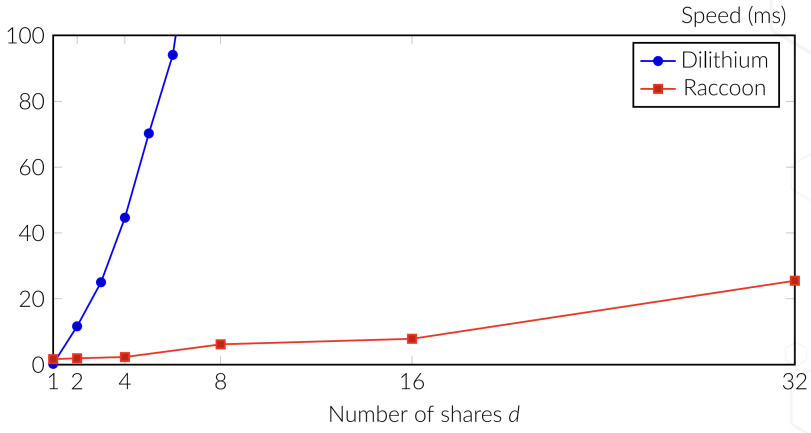


Figure: `https://raccoonfamily.org`

Further Work

Side-channel analysis and protection for PQC is an active research area.

It is a great topic for a master's project to study a PQC KEM or DSA in detail, conduct experiments, test protection mechanisms, and provide analysis.

It is also possible to do a lightweight analysis for your technical essay.

Questions?