Single-Trace Side-Channel Attacks on Masked Lattice-Based Encryption

Robert Primas, Peter Pessl, Stefan Mangard
IAIK, Graz University of Technology, Austria

CHES 2017, September 28

Outlook

- Single-trace SCA on masked asymmetric lattice-based encryption
- Combination of template attack (TA) with:
  - Belief Propagation
  - Lattice Decoding

Outlook

- Single-trace SCA on masked asymmetric lattice-based encryption
- Combination of template attack (TA) with:
  - Belief Propagation
  - Lattice Decoding

$\Rightarrow$ Full private key recovery

Motivation

- Lattice-based cryptography is a promising PQ candidate
  - Quantum computer resistant
  - Many efficient schemes available
- Not a lot analysis of implementation security

Motivation

- Lattice-based cryptography is a promising PQ candidate
  - Quantum computer resistant
  - Many efficient schemes available
- Not a lot analysis of implementation security

$\Rightarrow$ First single-trace SCA for lattice-based crypto

Ring-LWE Encryption

- Proposed by Lyubashevsky, Peikert and Regev[LPR10]
- Based on Learning with Errors Problem
- Operates on polynomials in the ring: $\mathbb{Z}_q[x]/(x^n + 1)$
  - In our setting: $q = 7681, n = 256$

Ring-LWE Encryption

$\mathbf{r}_2$
(private key)

$(\mathbf{a},\mathbf{p})$
(public key)

$\overline{\mathbf{m}}$
(encoded message)

alice

bob

* calculations are in $\mathbb{Z}_q[x]/(x^n+1)$

Ring-LWE Encryption

$\mathbf{r}_2$
(private key)

$(\mathbf{a},\mathbf{p})$
(public key)

$\overline{\mathbf{m}}$
(encoded message)

$\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3 \leftarrow \mathcal{X}^n$

alice

bob

* calculations are in $\mathbb{Z}_q[x]/(x^n+1)$

Ring-LWE Encryption

$\mathbf{r}_2$
(private key)

$(\mathbf{a},\mathbf{p})$
(public key)

$\overline{\mathbf{m}}$
(encoded message)

$\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3 \leftarrow \mathcal{X}^n$

alice

$$\mathbf{c}_1 = \mathbf{a}\mathbf{e}_1 + \mathbf{e}_2 \longleftarrow$$

(cipher text 1)

bob

* calculations are in $\mathbb{Z}_q[x]/(x^n+1)$

Ring-LWE Encryption

$\mathbf{r}_2$
(private key)

$(\mathbf{a},\mathbf{p})$
(public key)

$\overline{\mathbf{m}}$
(encoded message)

$\mathbf{e}_1,\mathbf{e}_2,\mathbf{e}_3 \leftarrow \mathcal{X}^n$

alice

$\mathbf{c}_1 = \mathbf{a}\mathbf{e}_1 + \mathbf{e}_2$
(cipher text 1)

$\mathbf{c}_2 = \mathbf{p}\mathbf{e}_1 + \mathbf{e}_3 + \overline{\mathbf{m}}$
(cipher text 2)

bob

* calculations are in $\mathbb{Z}_q[x]/(x^n+1)$

Ring-LWE Decryption

alice

$$\overline{\mathbf{m}} = \mathbf{c}_1\mathbf{r}_2 + \mathbf{c}_2$$

* calculations are in $\mathbb{Z}_q[x]/(x^n + 1)$

Ring-LWE Decryption

alice

$$\overline{\mathbf{m}} = \mathbf{c}_1\mathbf{r}_2 + \mathbf{c}_2$$

$\Rightarrow$ Inefficient: $> \mathcal{O}(n^2)$ due to polynomial division

* calculations are in $\mathbb{Z}_q[x]/(x^n + 1)$

Number Theoretic Transform (NTT)

- Efficient polynomial multiplication in certain rings, e.g.:

$$\mathbb{Z}_q[x]/(x^n + 1)$$

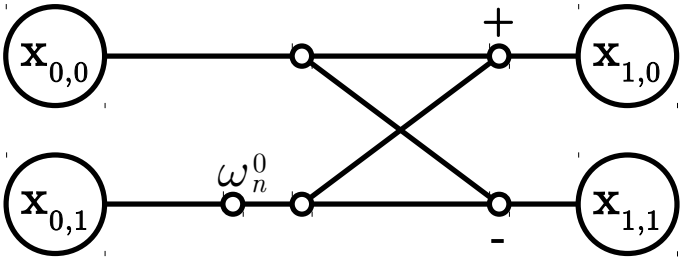
- Similar to FFT:

$$\mathbf{ab} = \text{INTT}(\ \text{NTT}(\mathbf{a})\ *\ \text{NTT}(\mathbf{b})\ )$$

- Features butterfly network

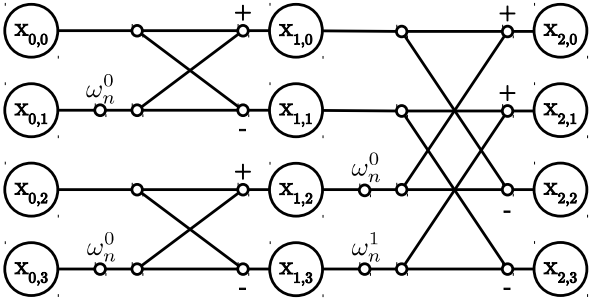
NTT - Butterfly

2-coefficients



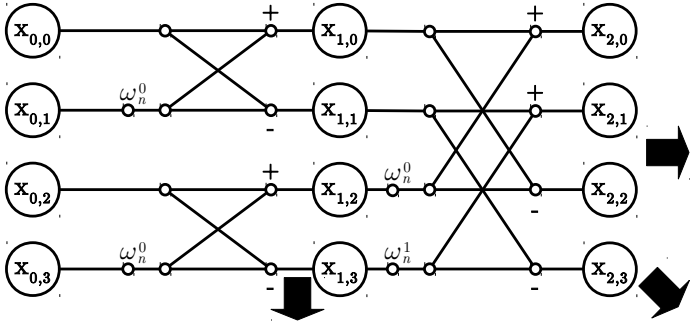
NTT - Butterfly Network

4-coefficients



NTT - Butterfly Network

256-coefficients



Efficient Ring-LWE Decryption


alice

$$\overline{\mathbf{m}} = \mathbf{c}_1\mathbf{r}_2 + \mathbf{c}_2$$

$$\overline{\mathbf{m}} = \mathrm{INTT}(\ \tilde{\mathbf{c}}_1 * \tilde{\mathbf{r}}_2 + \tilde{\mathbf{c}}_2\ )$$

* calculations are in $\mathbb{Z}_q[x]/(x^n + 1)$ * $\tilde{\mathbf{x}}$ is the NTT transformed of $\mathbf{x}$

Efficient Ring-LWE Decryption


alice

$$\overline{\mathbf{m}} = \mathbf{c}_1\mathbf{r}_2 + \mathbf{c}_2$$

$$\overline{\mathbf{m}} = \text{INTT}(\ \tilde{\mathbf{c}}_1 * \tilde{\mathbf{r}}_2 + \tilde{\mathbf{c}}_2\ )$$

$$\Rightarrow \text{Faster: } \mathcal{O}(n \log n)$$

* calculations are in $\mathbb{Z}_q[x]/(x^n + 1)$ * $\tilde{\mathbf{x}}$ is the NTT transformed of $\mathbf{x}$

Attack Idea

- Given the ciphertext $(\tilde{\mathbf{c}}_1, \tilde{\mathbf{c}}_2)$ and private key $\tilde{\mathbf{r}}_2$, decryption is defined as:

$$\overline{\mathbf{m}} = \mathrm{INTT}(\underbrace{\tilde{\mathbf{c}}_1 * \tilde{\mathbf{r}}_2 + \tilde{\mathbf{c}}_2}_{\mathcal{I}_{\mathrm{INTT}}}) \mod q$$

* public * $\tilde{\mathbf{x}}$ is the NTT transformed of $\mathbf{x}$

Attack Idea

- Given the ciphertext $(\tilde{\mathbf{c}}_1, \tilde{\mathbf{c}}_2)$ and private key $\tilde{\mathbf{r}}_2$, decryption is defined as:

$$\overline{\mathbf{m}} = \mathrm{INTT}(\underbrace{\tilde{\mathbf{c}}_1 * \tilde{\mathbf{r}}_2 + \tilde{\mathbf{c}}_2}_{\mathcal{I}_{\mathrm{INTT}}}) \mod q$$

- Thus $\tilde{\mathbf{r}}_2$ can be expressed as:

$$\tilde{\mathbf{r}}_2 = (\mathcal{I}_{\mathrm{INTT}} - \tilde{\mathbf{c}}_2) * \tilde{\mathbf{c}}_1^{-1} \mod q$$

* public * $\tilde{\mathbf{x}}$ is the NTT transformed of $\mathbf{x}$

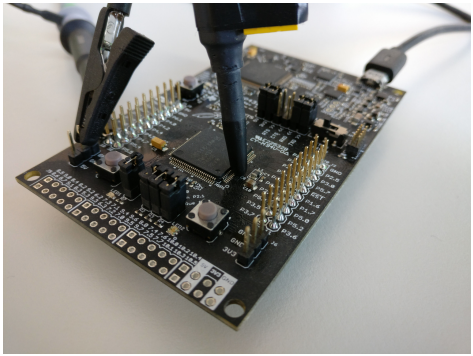
Attack Strategy

Steps:

1. Single-trace TA on the INTT operation
2. Leakage combination via Belief Propagation (BP)
3. Key recovery via lattice decoding

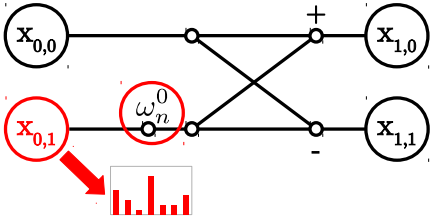
Step 1: Template Attack

- Efficient SW implementation by de Clercq et al. [dCRVV15]
- Texas Instruments MSP432 (ARM Cortex-M4F)
- EM-side-channel of power regulation circuitry
- Observed traces are expected to be close to power consumption



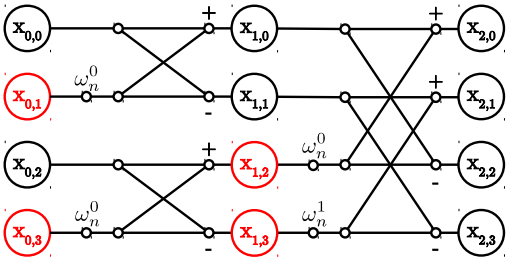
Step 1: Template Attack

- Target: Modular multiplication in each butterfly
- One factor of multiplication is always known (ω_n^x)
- Additional exploitation of timing information
- Goal: Probability distribution over each observed coefficient



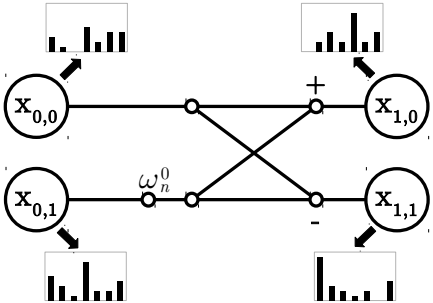
Step 1: Template Attack

- Target: Modular multiplication in each butterfly
- One factor of multiplication is always known (ω_n^x)
- Additional exploitation of timing information
- Goal: Probability distribution over each observed coefficient



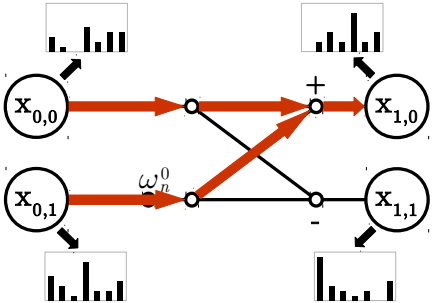
Step 2: Belief Propagation

- Iterative algorithm
- Calculate marginal distributions
- Combine leakage information
- Usage in SCA first proposed by Veyrat-Charvillon [VGS14]



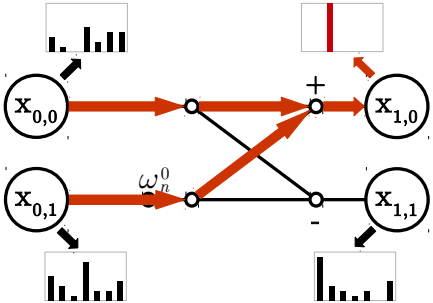
Step 2: Belief Propagation

- Iterative algorithm
- Calculate marginal distributions
- Combine leakage information
- Usage in SCA first proposed by Veyrat-Charvillon [VGS14]



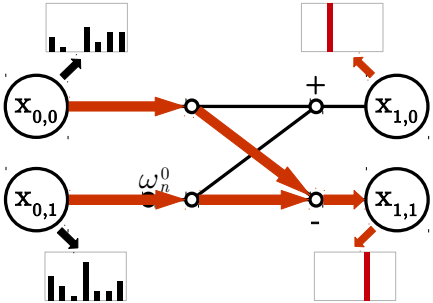
Step 2: Belief Propagation

- Iterative algorithm
- Calculate marginal distributions
- Combine leakage information
- Usage in SCA first proposed by Veyrat-Charvillon [VGS14]



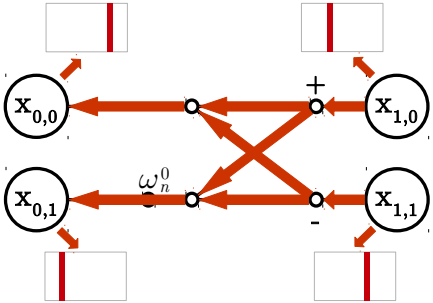
Step 2: Belief Propagation

- Iterative algorithm
- Calculate marginal distributions
- Combine leakage information
- Usage in SCA first proposed by Veyrat-Charvillon [VGS14]



Step 2: Belief Propagation

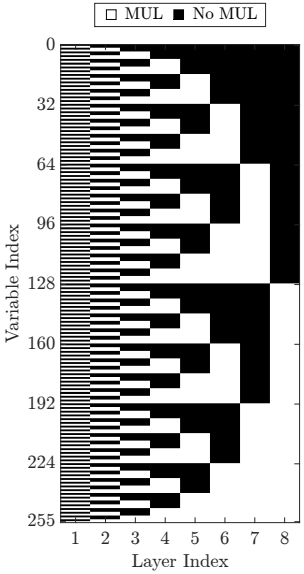
- Iterative algorithm
- Calculate marginal distributions
- Combine leakage information
- Usage in SCA first proposed by Veyrat-Charvillon [VGS14]



Step 2: Belief Propagation

Considerations:

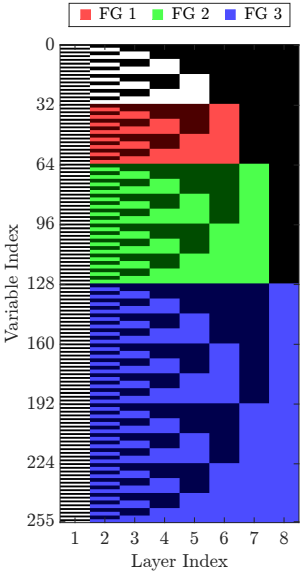
- Uneven distribution of side-channel information
- Bad TA performance in first layer ($\omega_n^0 = 1$)



Step 2: Belief Propagation

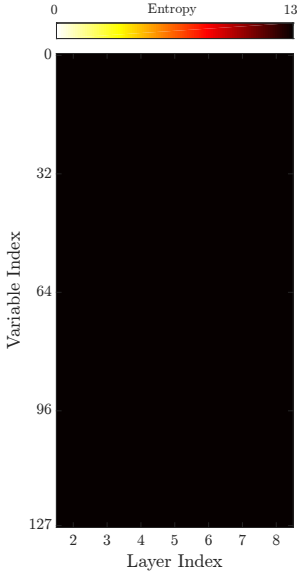
Solution:

- Perform BP on 3 Sub-Networks:
- Ignore areas with:
  - No / little side-channel information
  - Comparably noisy side-channel information
- Not all inputs can be recovered $\rightarrow$ Step 3:



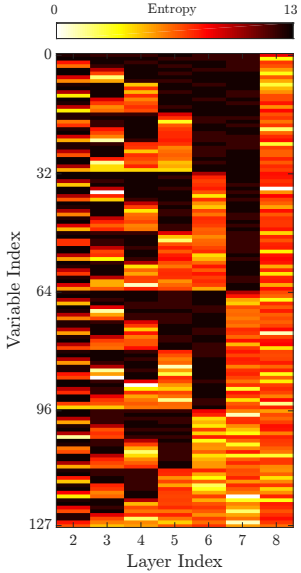
Step 2: Belief Propagation

Iteration 0



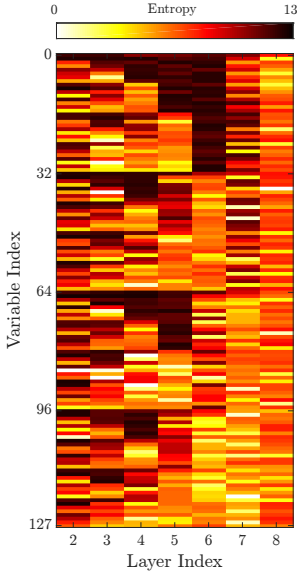
Step 2: Belief Propagation

Iteration 1



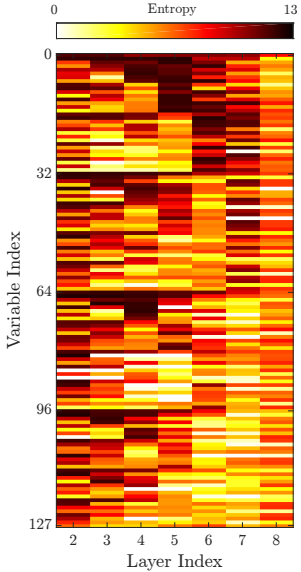
Step 2: Belief Propagation

Iteration 2



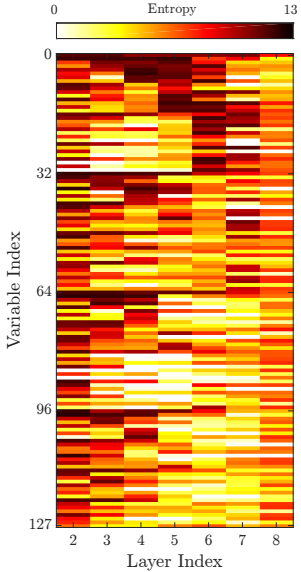
Step 2: Belief Propagation

Iteration 3



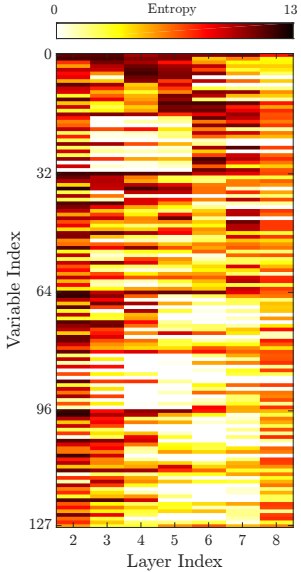
Step 2: Belief Propagation

Iteration 4



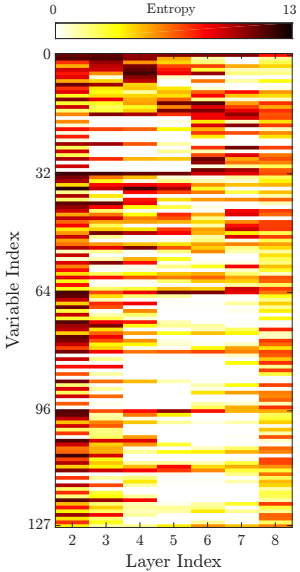
Step 2: Belief Propagation

Iteration 5



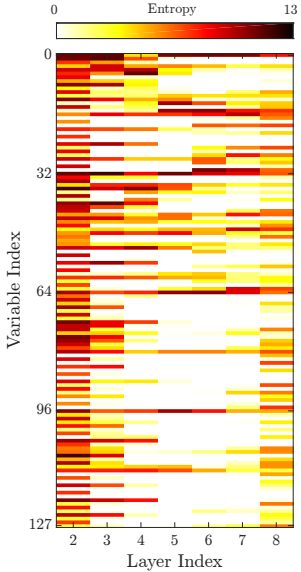
Step 2: Belief Propagation

Iteration 6



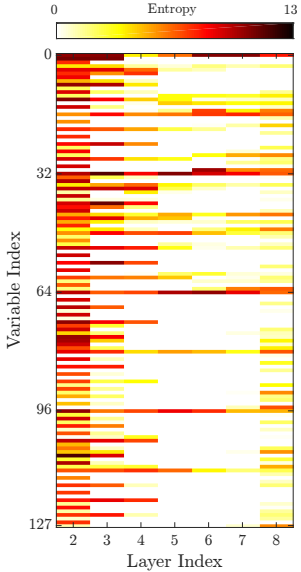
Step 2: Belief Propagation

Iteration 7



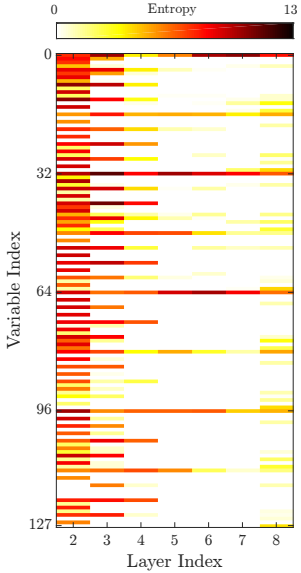
Step 2: Belief Propagation

Iteration 8



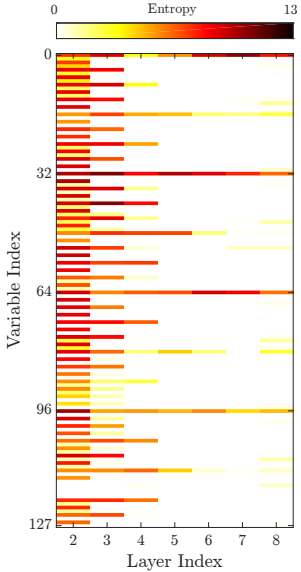
Step 2: Belief Propagation

Iteration 9



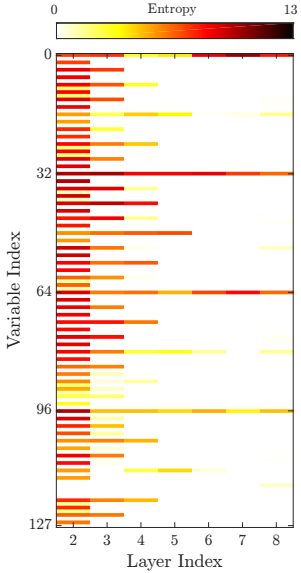
Step 2: Belief Propagation

Iteration 10



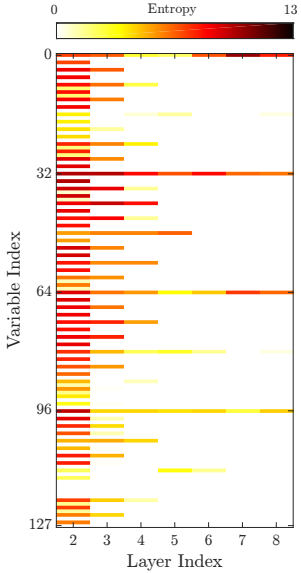
Step 2: Belief Propagation

Iteration 11



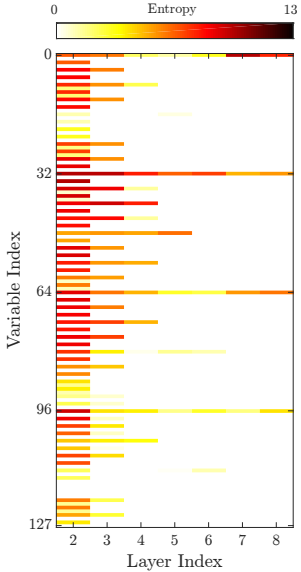
Step 2: Belief Propagation

Iteration 12



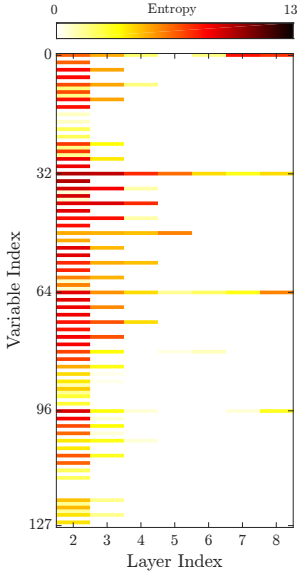
Step 2: Belief Propagation

Iteration 13



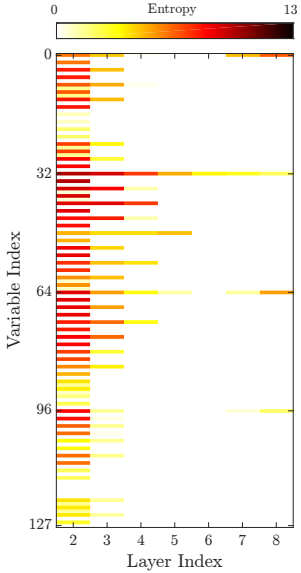
Step 2: Belief Propagation

Iteration 14



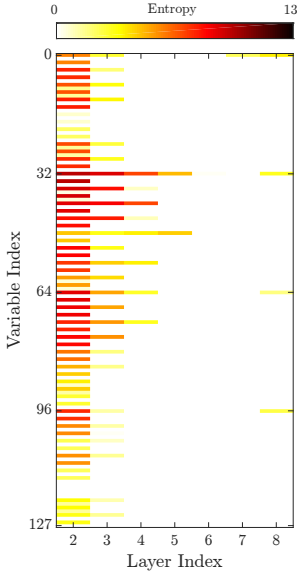
Step 2: Belief Propagation

Iteration 15



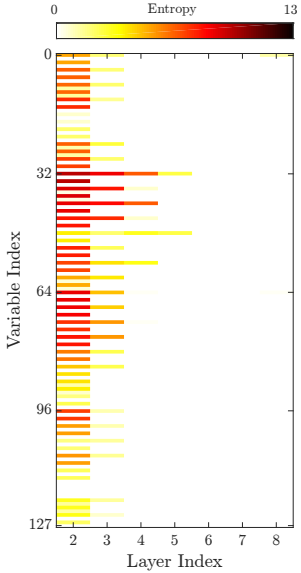
Step 2: Belief Propagation

Iteration 16



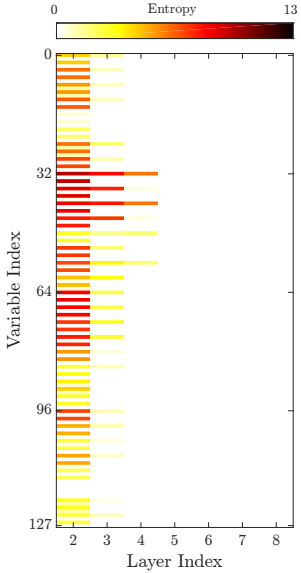
Step 2: Belief Propagation

Iteration 17



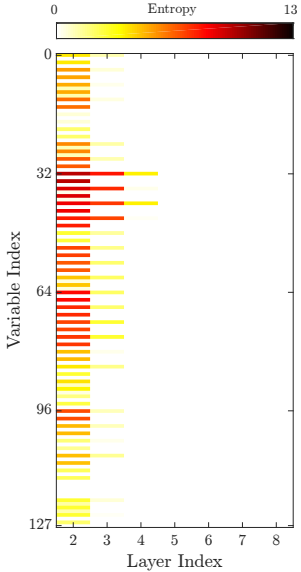
Step 2: Belief Propagation

Iteration 18



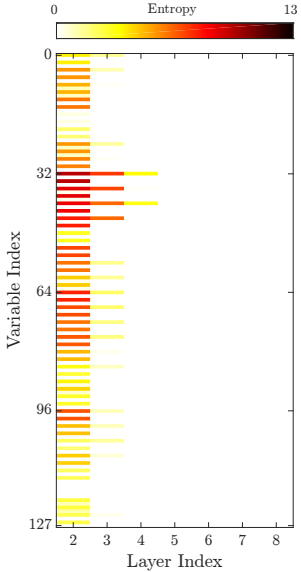
Step 2: Belief Propagation

Iteration 19



Step 2: Belief Propagation

Iteration ≥ 20

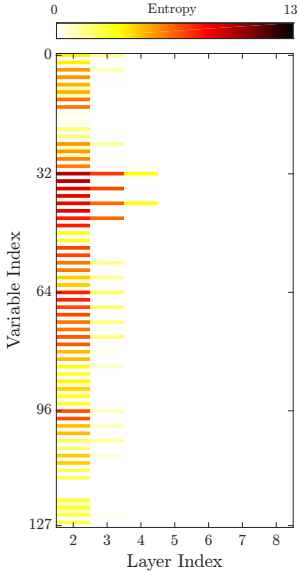


Step 2: Belief Propagation

- Still a lot of uncertainty in the input layer of all 3 Sub-Networks...
- We can exploit linearity of INTT to recover 192/256 inputs
- Brute forcing the remaining coefficients is still infeasible:

$$7681^{64} \approx 2^{826}$$

- Full key recovery still possible!



Step 3: Key Recovery

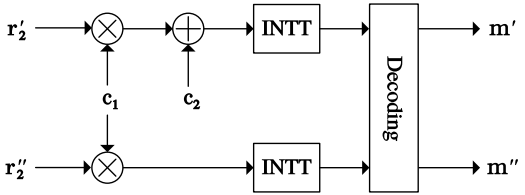
- Setup equation system that relates the 192 recovered coefficients to the private key $\mathbf{r}_2$
- Combine the equation system with the public key
- Recover $\mathbf{r}_2$ by solving a reduced rank $(256 - 192 = 64)$ SVP problem
  - BKZ Basis Reduction
- Success rate of lattice decoding is 1

Attack on masked implementation

- Proposed by Reparaz [RRdC+16]
- Private key $\mathbf{r}_2$ is split into $\mathbf{r}_2'$ and $\mathbf{r}_2''$ s.t.:

$$\mathbf{r}_2 = \mathbf{r}_2' + \mathbf{r}_2'' \mod q$$

- Recover 192 coefficients of one layer for both INTTs
- Perform pairwise addition of coefficients
- Proceed with Step 3 in unmasked scenario



Results

- Step 1: Obtain leakage of intermediate coefficients
- Step 2: Reliable recovery of coefficients in Sub-Networks
- Step 3: Lattice-decoding success rate is 1
- $\Rightarrow$ Attack success rate is 1
- Same holds for masked implementations
- Also evaluated for simulated noisy-HW leakage model

Single-Trace Side-Channel Attacks on Masked Lattice-Based Encryption

Robert Primas, Peter Pessl, Stefan Mangard
IAIK, Graz University of Technology, Austria

CHES 2017, September 28

Bibliography I

[dCRVV15] Ruan de Clercq, Sujoy Sinha Roy, Frederik Vercauteren, and Ingrid Verbauwhede. Efficient software implementation of ring-lwe encryption. In Wolfgang Nebel and David Atienza, editors, *DATE 2015*, pages 339–344. ACM, 2015.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, 2010.

[RRdC+16] Oscar Reparaz, Sujoy Sinha Roy, Ruan de Clercq, Frederik Vercauteren, and Ingrid Verbauwhede. Masking ring-lwe. *J. Cryptographic Engineering*, 6(2):139–153, 2016. Extended journal version of [RRVV15].

[RRVV15] Oscar Reparaz, Sujoy Sinha Roy, Frederik Vercauteren, and Ingrid Verbauwhede. A masked ring-lwe implementation. In Tim Güneysu and Helena Handschuh, editors, *CHES 2015*, volume 9293 of *LNCS*, pages 683–702. Springer, 2015.

[VGS14] Nicolas Veyrat-Charvillon, Benoît Gérard, and François-Xavier Standaert. Soft analytical side-channel attacks. In Palash Sarkar and Tetsu Iwata, editors, *ASIACRYPT 2014*, volume 8873 of *LNCS*, pages 282–296. Springer, 2014.