Master's Thesis 要旨

A Side-channel Assisted Lattice Attack on Cryptographic Schemes based on Number Theoretic Transform

Dept. of Mathematical Informatics 48216235 Yen-Ting Kuo

Supervisor Lecturer Atsushi Takayasu

1 Introduction

CRYSTALS-Kyber (Kyber) [1] belongs to the category of lattice-based cryptography, and in particular a module Learning With Errors (mod-LWE) scheme. Let $\mathcal{R}_q$ be the quotient ring $\mathbb{Z}_q[x]/(x^n+1)$. In the mod-LWE of Kyber, the sample is given of form $(\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e} \mod q)$, where $\mathbf{A}$ is chosen uniformly from $\mathcal{R}_q^{k\times k}$, and $\mathbf{s}, \mathbf{e} \in \mathcal{R}_q^{k\times 1}$ are again sampled from some small distribution.

Kyber prescribes the usage of the Number Theoretic Transform (NTT) for efficient polynomial multiplication. Normally the schoolbook multiplication takes $O(n^2)$ time. Via point-wise multiplication of transformed polynomials $\mathrm{NTT}(a)$ and $\mathrm{NTT}(b)$, i.e., $ab = \mathrm{NTT}^{-1}(\mathrm{NTT}(a) \circ \mathrm{NTT}(b))$, multiplication of a and b can be performed in time $O(n \log n)$, where n is the degree of polynomial a and b.

Power analysis attack is a kind of side-channel attack (SCA) exploiting the fact that the instantaneous power consumption of a cryptographic device depends on the data it processes and on the operation it performs. There are two types of power analysis attack, namely the simple power analysis (SPA) [4] and correlation power analysis (CPA) [3]. These attacks had been proved to jeopardize the security of the classical cryptographic field like the unprotected version of RSA and ECC.

In this thesis, our goal is to combine correlation power analysis and lattice reduction to fully recover the secret key of lattice-based cryptographic schemes that utilize NTT as their intrinsic polynomial multiplication method. Our attack consists of two steps:

- First, by exploiting the correlation of Hamming weight of some intermediates and the power consumption of the decryption process, precisely the part where we multiply the secret key with ciphertext, we can recover some of the coefficients of the secret key in the NTT domain.
- Secondly, since there will be some ambiguity about whether the recovered coefficients are indeed correct, we sample part of the recovered coefficients and construct a lattice problem by Kannan's embedding method. Then one can recover the entire secret key by solving the lattice problem by using lattice reduction algorithms such as the BKZ algorithm [2].

2 Correlation Power Analysis

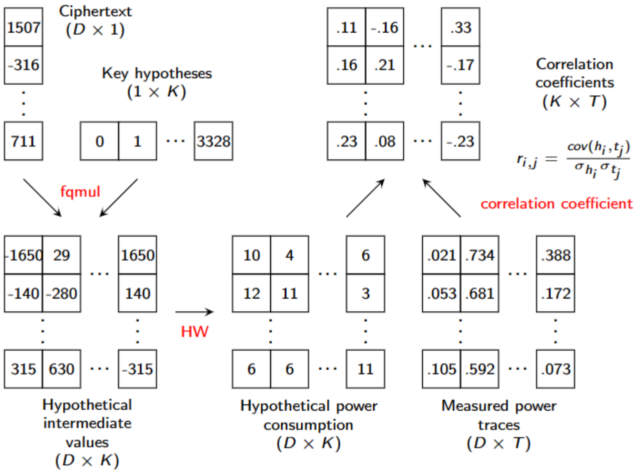

Fig. 1. The CPA process for recovering $\hat{\mathbf{s}}_0$

Our attack targets the NTT in the decryption process of Kyber, with the aim of recovering the victim's secret key $\hat{\mathbf{s}}$ in the NTT domain. To decrypt a message the recipient calculates $\mathrm{NTT}^{-1}(\hat{\mathbf{s}}^\top \circ \hat{\mathbf{u}})$, where $\hat{\mathbf{u}}$ is the decompressed ciphertext in the NTT domain and $\circ$ denotes the pairwise multiplication. The pairwise multiplication is operated in the quotient ring $\mathbb{Z}_q[x]/(x^2 - \zeta_i)$, where ζ_i are the primitive roots of unity of $\mathbb{Z}_q$. In such a ring, the product of two polynomials $a = a_0 + a_1 x$ and $b = b_0 + b_1 x$ can be easily computed as

$$ab = (a_0 b_0 + a_1 b_1 \zeta_i) + (a_0 b_1 + a_1 b_0)x \mod q.$$

Let x and y be two integers in the range $[-q+1, q-1]$, we refer to the output of $x \times y$ by the `REDC` algorithm as $\texttt{fqmul}(x, y)$. Then the product $r_0 + r_1 x = ab2^{-16}$ can be computed as follow:

$$\begin{aligned}
r_0 &\leftarrow \texttt{fqmul}(a_1, b_1)\\
r_0 &\leftarrow \texttt{fqmul}(r_0, \zeta_i 2^{16})\\
r_0 &\leftarrow \texttt{fqmul}(a_0, b_0) + r_0 \qquad (1)\\
r_1 &\leftarrow \texttt{fqmul}(a_1, b_0)\\
r_1 &\leftarrow \texttt{fqmul}(a_0, b_1) + r_1.
\end{aligned}$$

The unwanted constant can be dealt with in the inverse NTT together when we divide the coefficient by n, thus no extra multiplications is needed.

Now suppose we want to reveal the coefficients of secret key $(\hat{s}_0, \hat{s}_1)$ in the NTT domain, notice that they are point-wisely multiplied by the ciphertext $(\hat{u}_0, \hat{u}_1)$, then our first chosen intermediate value is $\texttt{fqmul}(\hat{s}_1, \hat{u}_1)$, i.e. r_0 in the first line of equation (1). The overall process of this step is depicted in Fig. 1.

3 Lattice Attack

NTT and inverse NTT are linear transform, thus we can describe NTT with a matrix-vector multiplication. Let $\mathbf{M} = [\mathbf{m}_0, \mathbf{m}_2, ..., \mathbf{m}_{254}]$ be the inverse NTT matrix. $\mathbf{M}$ is a 128×128 matrix since there are 7 layers in the NTT of Kyber. Suppose we have recovered $2(128 - \ell)$ coefficients of $\hat{\mathbf{s}}$ from the polynomial multiplication $\hat{\mathbf{s}} \circ \hat{\mathbf{u}}$, i.e., we need to recover the rest 2ℓ coefficients. Notice that the coefficients of indices $2i$ and $2i+1$ are either recovered or rejected simultaneously. Now we focus on the coefficients of indices $2i$. Let $A = \{a_0, a_1, ..., a_{127-\ell}\}$ be the indices that are successfully recovered in the CPA step, and $B = \{b_0, b_1, ..., b_{\ell-1}\}$ be the indices that are still unknown, then the inverse NTT $\mathrm{NTT}^{-1}(\hat{\mathbf{s}}) = \mathbf{M}\hat{\mathbf{s}} = \mathbf{s} \mod q$ can be split into two halves as followed:

$$\mathbf{M}_A\hat{\mathbf{s}}_A + \mathbf{M}_B\hat{\mathbf{s}}_B = \mathbf{s} \mod q,$$

where $\mathbf{M}_A := [\mathbf{m}_{a_0}, ..., \mathbf{m}_{a_{127-\ell}}]$ is a matrix whose columns are those of $\mathbf{M}$ whose indices are in A, $\hat{\mathbf{s}}_A = [\hat{s}_{a_0}, ..., \hat{s}_{a_{127-\ell}}]^\top$, and the similar definition for $\mathbf{M}_B$ and $\hat{\mathbf{s}}_B$. Notice that $\mathbf{s}$ is an extremely short vector since it is the secret key sampled from β_η. By calling the known vector $\mathbf{t} = [\mathbf{m}_i]_{i\in A}[\hat{s}_i]_{i\in A}^\top$, the known basis $\bar{\mathbf{A}} = -\mathbf{M}_B$, and an unknown vector $\mathbf{s}' = [\hat{s}_j]_{j\in B}^\top$, we now have $\mathbf{t} = \bar{\mathbf{A}}\mathbf{s}' + \mathbf{s} \mod q$, which is exactly the definition of an LWE problem. Compared to the original mod-LWE problem in Kyber, this problem becomes simpler since the rank of $\bar{\mathbf{A}}$ is less than the original one.

We use the standard technique of Kannan's embedding to solve the LWE problem. We can treat the LWE problem as an uSVP by a technique called Kannan's embedding. Given the LWE instance above, we consider the following basis matrix

$$\mathbf{B}_{Kan} = \left[\begin{array}{c|c} \begin{matrix} \mathbf{I}_l & \mathbf{A}' \\ \mathbf{0} & q\mathbf{I}_{n-\ell} \end{matrix} & \mathbf{0} \\ \hline \mathbf{t}^\top & 1 \end{array}\right].$$

where $[\mathbf{I}_\ell \mid \mathbf{A}']$ denotes the reduced row echelon matrix of $\bar{\mathbf{A}}^\top$, which can be easily calculated by Gaussian elimination.

To determine the least number of coefficients we must recover in the CPA step, we conduct an experiment on solving the SVP randomly generated by script. The result is shown in Fig. 2, where the blue line is the success rate of finding $[\mathbf{s}^\top \mid 1]$ by the BKZ algorithm of block size 50 for 20 randomly generated $\mathbf{s}$, and the red line is the running time of the algorithm. From the result, the critical point of guaranteed success is on $\ell = 89, \ell = 90$ for Kyber512, Kyber768/1024, respectively. This means that in the CPA step, we need at least $128 - 89 = 39$ recovered coefficients for Kyber512, or 38 for Kyber768/1024, so that we can have a fully recovered secret key when using the BKZ algorithm of block size 50 to solve the reduced SVP problem. The reason that Kyber768/1024 is easier to solve is because η of Kyber768/1024 is smaller than that of Kyber512.

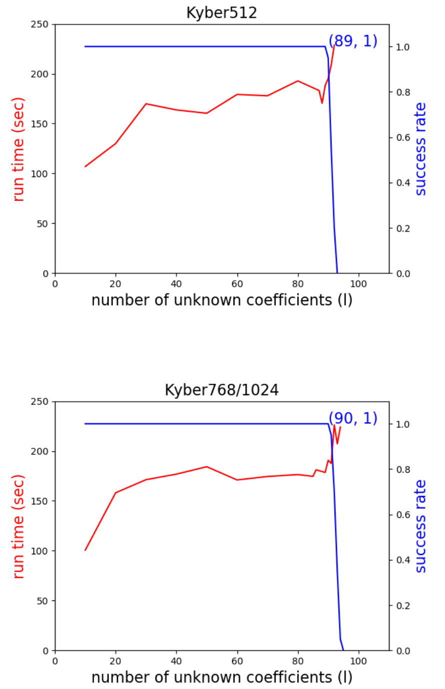

Fig. 2. Success rate and running time on randomly generated uSVP in the lattice $\mathbf{B}_{Kan}$ for Kyber512 and Kyber768/1024

4 Conclusion

In this thesis, we derived a practical methodology to combine correlation power analysis and lattice attack that exploited the Number Theoretic Transform inside some lattice-based cryptosystems. With 200 traces, our attack terminated within 20 minutes on a 16-core computer. Compared to other SCA targeting NTT in the cryptosystems, our attack achieves lower runtime in practice. Furthermore, there is potential for decreasing the number of traces by using lattice reduction if the same measurement is used.

Bibliography

[1] J. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehle. CRYSTALS - Kyber: A CCA-secure module-lattice-based KEM. In *2018 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 353–367, 2018.

[2] Y. Chen and P. Q. Nguyen. BKZ 2.0: Better lattice security estimates. In D. H. Lee and X. Wang, editors, *Advances in Cryptology – ASIACRYPT 2011*, pages 1–20, 2011.

[3] P. Kocher, J. Jaffe, and B. Jun. Differential power analysis. In M. Wiener, editor, *Advances in Cryptology — CRYPTO' 99*, pages 388–397, 1999.

[4] P. C. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems. In *Advances in Cryptology - CRYPTO '96, 16th Annual International Cryptology Conference, Proceedings*, pages 104–113, 1996.