MODULUS SWITCHING

HAO CHEN

1. The idea

References: [LaSt]

The setup of modulus switching is like this: you have a modulus q, a ring R (some ring of integers, or orders), a secret s, and samples

$$(a, b = as + e + \lambda q)$$

where $\lambda \in R$. You have a new modulus p you want to switch to, the idea is to take

$$\alpha = p/q$$

and multiply both sides by α. Now we get

$$(\alpha a, \alpha b = \alpha a s + e + \lambda p).$$

Note this is not a valid RLWE sample in R/pR: the problem is that αa and αb are not in R. To solve this problem, we introduce a rounding function:

$$[\cdot] : 1/qR \to R$$

The essential freedom we have in this modulus switching game is to choose the rounding function. For example, [BV] uses "Gaussian rounding"; there's also the "naive rounding", which represents αa as some polynomial with rational coefficients and then round to the nearest integer (this approach of course is basis-dependent and might not be what you want to do). Then we apply the rounding function and get samples

$$([\alpha a], [\alpha b]).$$

The first question is: how come is this still an RLWE sample? We must look at the "new error"

$$e' := [\alpha b] - [\alpha a]s$$

If modulus switching works we would want e' to behave like an "error", i.e., not too large or too uniform-looking. To see this, we have

$$e' = -b' - a's + \alpha e$$

where a', b' are "rounding errors". If s is sufficiently small, then e' can be bounded as well. Thus the essence of modulus switching is about how to make rounding errors small. In Gaussian rounding, one samples a' from a discrete Gaussian on a lattice coset:

$$a' \leftarrow D_{R+\alpha a, r}.$$

In other words, one samples a short vector a' so that $a' - \alpha a$ is in R. Then one defines $[\alpha a] := \alpha a - a'$. This is how the rounding is done. Same for b'. What's the problem of this approach?

The problem: the only known way to efficiently sample from discrete Gaussians is [GPV], which requires r to be $\omega(\log(n))$ times the Gram-Schmidt norm of the input lattice basis $\mathbf{B}$. So during modulus switching the error will have to grow by at least this amount. In l_2-norm, we can put a bound to be

$$l_2(e') \leq C(1 + ||s||_\infty)\sqrt{n}\log(n)||\tilde{B}|| + \alpha\, l_2(e).$$

2. Modulus switching attack

Modulus switching attack is based on the above modulus switching scheme, but deliberately switch to some modulus which is weak to our attack. For a non-appropriate example, think of $R = \mathbb{Z}[\zeta_p]$ and switching to the modulus p.

Now we are ready to explain why modulus switching attack does not usually work well in practice: the problem is in the $||\tilde{B}||$ term, which usually grows with the size of a weak modulus p. This may sound weird but one should think of $||\tilde{B}||$ as taking the place of the "expansion factors". Now if $f(1) \equiv 0 \pmod{p}$, then the expansion factor will have to be large compared to p as well.

We may still, via searching, find some instances vulnerable to modulus switching attack, relaxing the fact that R being the ring of integers. Examples like $x^n + ax + b$, or I have the following Galois example

$$f = x^8 + 12x^7 + 78x^6 + 312x^5 + 854x^4 + 1608x^3 + 2052x^2 + 1704x + 772,$$

and $q = f(1) = 7393$. This example is interesting because it has a relatively small $||\tilde{\mathbf{B}}||$.