SOLILOQUY: A CAUTIONARY TALE

Peter Campbell, Michael Groves and Dan Shepherd
CESG, Cheltenham, UK

1. Introduction

The Soliloquy primitive, first proposed by the third author in 2007, is based on cyclic lattices. It has very good efficiency properties, both in terms of public key size and the speed of encryption and decryption. There are straightforward techniques for turning Soliloquy into a key exchange or other public-key protocols. Despite these properties, we abandoned research on Soliloquy after developing (2010 to 2013) a reasonably efficient quantum attack on the primitive. A similar quantum algorithm has been recently published in some highly insightful independent work by Eisenträger, Hallgren, Kitaev, and Song [2]. However, their paper concentrates on computing unit groups of arbitrary degree number fields whereas we will show how to apply the approach to the special case of Soliloquy.

We begin with a complete description of the Soliloquy algorithm, omitting arguments for it resisting classical cryptanalysis. Then we review the difficulties of attacking it with a quantum computer, and describe the main concept of a "lattice fingerprint" that leads to our polynomial-time quantum attack.

The public key is a large prime, p, having perhaps between 3000 and 10000 bits, depending on the desired level of security. The private key is a factor of p over a certain cyclotomic ring; that is, some algebraic integer α whose norm is p. The private key has certain properties that enable the decryption of correctly formatted vectors, but which make its recovery from p an apparently hard (classical) lattice basis reduction problem. We associate a cyclic lattice with the principal prime ideal generated by α and the cryptography can be understood as a variant of the GGH cryptosystem [4] using that lattice. In particular, decryption will be achieved using Babai's round-off algorithm with the reduced basis given by the factor α. One advantage of Soliloquy over similar proposals such as [3, 9] is that the public basis of the lattice has a very compact representation as it is defined solely by p.

The quantum attack solves the problem of finding a small generator of a principal ideal in the ring of algebraic integers. The situation is helped by the fact that a basis for the group of units is (more-or-less) already known and so does not need to be first computed by another (possibly quantum) algorithm. We believe that there are aspects of the design and attack that would benefit from further study. We are not aware of any classical subexponential time attacks on the system, but this may also be worth thinking about.

Finally, we would like to state clearly that, following our work on the quantum algorithm, we have stopped the development of Soliloquy as a potential quantum-resistant primitive and we do not recommend its use for real-world deployment. For example, although the quantum algorithm required to break Soliloquy would appear to be somewhat more complex than a quantum algorithm for breaking RSA, the difference in their 'effective lifetimes' for corresponding key sizes would be predicted to be a couple of years at best, so inhibiting any cost-effective transition between these cryptographic solutions.

2. Soliloquy

2.1. **Background.** Let n be a prime, about 10 bits in size, $K = \mathbb{Q}(\zeta)$ be the cyclotomic field obtained by extending the rational field by a primitive n-th root of unity ζ, and $\mathcal{O} = \mathbb{Z}[\zeta]$ be the ring of integers in K. For a prime $p \equiv 1 \bmod n$, the principal ideal $p\mathcal{O}$ decomposes into a product of prime ideals

$$p\mathcal{O} = \prod_{i=1}^{n-1} \mathcal{P}_i$$

of norm $\mathrm{N}(\mathcal{P}_i) = p$, which are permuted by the action of the Galois group $\mathrm{Gal}(K/\mathbb{Q}) \simeq (\mathbb{Z}/n\mathbb{Z})^\times$. Each prime ideal has a two-element representation of the form $\mathcal{P}_i = p\mathcal{O} + (\zeta - c_i)\mathcal{O}$, where the c_i are the non-trivial n-th roots of unity mod p. Further, $c = 2^{(p-1)/n} \bmod p$ is non-trivial with probability $(1 - 1/n)$, in which case one of the prime ideals has the two-element representation $\mathcal{P} = p\mathcal{O} + (\zeta - c)\mathcal{O}$.

This gives a method of identifying certain rational primes with prime ideals of $\mathcal{O}$ which have specific two-element representations. Moreover, there is a natural homomorphism $\psi : \mathcal{O} \to \mathcal{O}/\mathcal{P} \simeq \mathbb{F}_p$ given by

$$\psi\left(\sum_{i=0}^{n-1} e_i \zeta^i\right) = \sum_{i=0}^{n-1} e_i c^i \bmod p$$

which we will use for our encryption function. To decrypt, we need a mechanism for recovering a distinguished coset representative. This is achieved by constructing our prime p so that $\mathcal{P} = \alpha\mathcal{O}$ for a small element $\alpha \in \mathcal{O}$ and viewing decryption as an instance of the close vector problem in the ideal lattice corresponding to $\mathcal{P}$.

A similar approach was used as the basis of the Smart-Vercauteren FHE scheme [9] and its variants such as [3]. However, here we are interested in the potential quantum-resistance of the system, rather than its homomorphic properties.

2.2. **Key generation.** A candidate SOLILOQUY private key will be an element

$$\alpha = \sum_{i=1}^{n} a_i \zeta^i \in \mathcal{O}$$

where the coefficients a_i are sampled from a discrete Gaussian of mean 0 and width σ. These are then tested to ensure that $p = \mathrm{N}(\alpha)$ is a valid SOLILOQUY public key:

(i) p is prime;
(ii) $c = 2^{(p-1)/n} \not\equiv 1 \bmod p$.

Note that these conditions guarantee only that $\alpha\mathcal{O}$ is *one* of the prime divisors of $p\mathcal{O}$. We can ensure that $\alpha\mathcal{O} = p\mathcal{O} + (\zeta - c)\mathcal{O}$ by taking the appropriate Galois conjugate of α, that is, by reordering the coefficients a_i.

2.3. **Primitive.** We will now outline the core SOLILOQUY algorithm which can be viewed as a key encapsulation mechanism.

Generate an ephemeral element

$$\epsilon = \sum_{i=1}^{n} e_i \zeta^i \in \mathcal{O}$$

where the coefficients are again sampled from a discrete Gaussian of mean 0 and width σ', or some other similar distribution. This is then encapsulated as

$$z = \sum_{i=1}^{n} e_i c^i \bmod p$$

considered as an integer $0 \leq z < p$.

To recover ϵ from z we note that $\epsilon \in z + \alpha \cdot \mathcal{O}$. Thus, we should see that

$$\epsilon = z - \lceil z\alpha^{-1} \rfloor \cdot \alpha,$$

provided that ϵ was chosen small enough so that $\lceil \epsilon \cdot \alpha^{-1} \rfloor = 0$, where $\lceil \omega \rfloor$ corresponds to co-ordinate-wise rounding. In fact, it is more efficient to store and use the integer value

$$d \equiv n^{-1} \cdot \mathrm{Tr}[p/\alpha] \pmod{p}$$

as private key, instead of storing α itself. For the sake of brevity, we will avoid any further discussion of the details and correctness of decryption, and instead direct the interested reader to [9], where a very similar procedure is used.

2.4. **Lattices.** It possible to view SOLILOQUY as a version of the GGH lattice-based encryption scheme [4]. Consider the cyclic lattice generated by the rows of the following matrix

$$C = \begin{bmatrix} a_0 & a_1 & \dots & a_{n-2} & a_{n-1} \\ a_{n-1} & a_0 & & a_{n-3} & a_{n-2} \\ \vdots & & \ddots & & \\ a_2 & a_3 & & a_0 & a_1 \\ a_1 & a_2 & & a_{n-1} & a_0 \end{bmatrix}.$$

This has Hermite Normal Form

$$H = \begin{bmatrix} 1 & 0 & \dots & 0 & -c^{n-1} \\ 0 & 1 & & 0 & -c^{n-2} \\ \dots & & \ddots & & \\ 0 & 0 & & 1 & -c \\ 0 & 0 & & 0 & p \end{bmatrix}$$

and we see that

$$(0, 0, \dots, 0, z) = (e_{n-1}, e_{n-2}, \dots, e_1, e_0) - (e_{n-1}, e_{n-2}, \dots, e_1, k) \cdot H$$

where $z = \sum_{i=0}^{n-1} e_i c^i - kp$.

In particular, since α is small, C is a reduced basis for the lattice and the decryption routine is essentially Babai's round-off algorithm. Indeed, the security of message recovery is given by the difficulty of the close vector problem in the cyclic lattice.

2.5. **Principal ideal problem.** Key recovery is an example of the small principal ideal problem: given a representation of a principal ideal $\mathcal{I}$ of $\mathcal{O}$, compute a small generator α of the ideal.

Until quite recently, the general case had been widely believed to be a hard problem, both classically and for a quantum computer. There are classical algorithms for constructing generators of principal ideals which are subexponential in the discriminant [1], and Hallgren [6] has an 'old' polynomial-time quantum algorithm that applies to algebraic fields of fixed (small) degree. We therefore set out with the belief that SOLILOQUY had the potential to be a practical quantum-resistant primitive, since it uses fields of relatively large degree.

3. Quantum Attack

3.1. **Simplification.** We begin by making some simplifications.

Firstly, we assume that the cyclotomic units have index 1 in the entire group of units $\mathcal{O}^\times$, which is almost certainly true for the specific instance of SOLILOQUY that had been proposed. A simple generating set for the cyclotomic units is of course known. The image of $\mathcal{O}^\times$ under the logarithm map forms a lattice. The determinant of this lattice turns out to be much bigger than the typical log-length of a private key α, so it is easy to recover the causally short private key given *any* generator of $\alpha\mathcal{O}$, *e.g.* via the LLL lattice reduction algorithm. We therefore reduce the problem to that of finding any generator of the input ideal, not necessarily a small one.

Secondly, we note that by [7] to find a generator of $\alpha \in \mathcal{O}$ it is enough to recover $\alpha' = \alpha \cdot \alpha^*$ in the ring of integers $\mathcal{O}' = \mathbb{Z}[\zeta + \zeta^{-1}]$ of the maximal real subfield $K' = \mathbb{Q}(\zeta + \zeta^{-1})$ of K. Thus in some sense we are working in a higher-dimension case of [5] rather than the more general situation considered in [6].

Consequently, for the remainder of this paper we will assume that we are attempting to recover *any totally positive generator* of the principal ideal $\alpha'\mathcal{O}'$ in the ring of integers $\mathcal{O}'$ of a totally real field K' of degree r, where we know a generating set $u_1, \ldots, u_{r-1}$ of the unit group $\mathcal{O}'^{\times}$ consisting of small totally positive units.

3.2. **Lattices and hidden subgroups.** We will solve this special case of the principal ideal problem by setting it up as a hidden lattice problem.

There is a natural logarithmic embedding $\log : K' \to \mathbb{R}^r$, given by

$$\log(\omega) = (\log(|\sigma_0(\omega)|), \ldots, \log(|\sigma_{r-1}(\omega)|))$$

where the σ_i are the Minkowski embeddings of K' into $\mathbb{R}$. Under this embedding the unit group maps to a lattice $\Lambda = \log(\mathcal{O}'^{\times})$ in $\mathbb{R}^r$ of rank $r-1$.

Consider the following rank-r lattice encoding α':

$$\Lambda_{\alpha'} = \begin{bmatrix} -1 & \log(\alpha') \\ 0 & \log(u_1) \\ \vdots & \\ 0 & \log(u_{r-1}) \end{bmatrix}.$$

Let X denote the set of all lattices in $\mathbb{R}^r$. We want to hide the lattice $\Lambda_{\alpha'}$ by defining a function $F : \mathbb{Z} \times \mathbb{R}^r \to X$, such that $F(k,v) = F(k',v')$ if and only if $(k,v) \equiv (k',v') \bmod \Lambda_{\alpha'}$. Note that the basis for the log-unit lattice Λ is explicitly known, so, although $\Lambda_{\alpha'}$ is a hidden high-dimensional lattice, it has a known sublattice Λ of co-dimension 1.

The control space to use is

$$G = \left\{ (k,v) \in \mathbb{Z} \times \mathbb{R}^r : \sum_{i=0}^{r-1} v_i = -k \log(\mathrm{N}(\alpha'\mathcal{O}')) \right\}$$

and set $F : G \to X$ to be

$$F(k,v) = \exp(v) \cdot (\alpha'\mathcal{O}')^k$$

where $\exp(v) = (\exp(v_0), \ldots, \exp(v_{r-1}))$. Although k and v will be exponentially large there are techniques which allow $F(k,v)$ to be computed efficiently, but we will not describe these here.

3.3. **Algorithm outline.** The basic quantum algorithm operates on three registers as follows:

1. Design efficient circuitry for rendering a "quantum fingerprint" $\psi(k,v)$ of the (LLL-reduced basis of the) lattice $F(k,v)$.

2. Suitably discretise and bound G and form the superposition
$$\sum_{(k,v)\in G} |k,v,0\rangle \mapsto \sum_{(k,v)\in G} |k,v,\psi(k,v)\rangle \, .$$
Take a quantum Fourier transform over G and measure the third register to obtain an approximation to the dual lattice $\Lambda^*_{\alpha'}$.

3. By iterating the previous step produce many samples close to $\Lambda^*_{\alpha'}$.

4. Feed these into a classical lattice-based algorithm to obtain an approximate basis for $\Lambda_{\alpha'}$ and use this to recover α' exactly.

The final step is relatively straightforward to solve using classical lattice-based algorithms, provided that we are working with sufficient precision. The main novel challenge was to define a suitable *quantum fingerprinter* (one that can handle lattice the $F(k,v)$ even when the parameters are exponentially large) that works with suitable discretisation of G.

3.4. **Lattice binning.** Unlike sublattices of the integers, *real* lattices do not possess a Hermite Normal Form so we cannot use this as an invariant to represent the lattice in quantum memory. We should instead use a "quantum fingerprint" to identify the lattice as closely as possible. Before we can define the fingerprint we need first to introduce the concept of "lattice binning".

Let B be an $r \times r$ matrix containing the basis for the lattice after Gram-Schmidt orthogonalisation and let $l \in \mathbb{R}$ be a fixed length. We will use an enumeration map $\phi_B : [0,l) \to \mathbb{Z}^r$ which depends on B and which can be inverted at integer points to facilitate reversible computation. For brevity we will not give an explicit description of ϕ_B. Instead we simply state the key property that if ζ_B denotes the inverse map then for any $x \in [0,l) \cap \mathbb{Z}$ we have
$$x = \lceil \zeta_B(\phi_B(x)) \rceil - 1.$$

In particular, if $C_r(B,l) = \{\ \phi_B(x)\ :\ x \in [0,l) \cap \mathbb{Z}\ \}$ is the set of evaluations at the integers then for any $\mathbf{v} \in C_r(B,l)$ we can use ζ to find an x for which $\mathbf{v} = \phi_B(x)$. Moreover, $C_r(B,l)$ is a discretised model for $E_r(\rho) = Ball_{r,\rho} \cdot B^{-1}$ in the sense that that it fits within an ellipsoid $E_r(\rho+\varepsilon)$ and covers all the integer points in $E_r(\rho-\varepsilon)$; that is,
$$E_r(\rho-\varepsilon) \cap \mathbb{Z}^r \ \subseteq\ C_r(B,l)\ \subseteq\ E_r(\rho+\varepsilon) \cap \mathbb{Z}^r.$$

Let O denote the isometry between the Gram-Schmidt and the "natural" bases for the lattice. Each $\mathbf{v} \in C_r(B,l)$ indexes a short vector $\mathbf{v} \cdot B$ in the

Gram-Schmidt basis and so a corresponding vector $\mathbf{v} \cdot B \cdot O$ in the natural basis. We will use another lattice to partition up natural space into cells or "bins" and replace the vector $\mathbf{v} \cdot B \cdot O$ by the label $\mathbf{u}$ of its bin. This reduces precision in a useful way, but is only efficient if size-reduction to the cells is straightforward. We will therefore restrict our attention to the case of binning via the regular cubic lattice. Further, we introduce a carefully chosen scaling factor q to make the computations more numerically stable.

The simple binning function is

$$\theta_B(\mathbf{v}) = \lceil q \cdot \mathbf{v} \cdot B \cdot O \rfloor.$$

This scales the vector indexed by $\mathbf{v}$ and rounds off each coordinate to the nearest integer. It is equivalent to the CVP algorithm for the regular cubic lattice, and is reversible. We will denote the inverse function by η_B.

Now let R be a random $r \times r$ orthogonal matrix, and let $\mathbf{w}$ be a random vector in $\mathbb{R}^r$. We can use R and $\mathbf{w}$ to rotate and translate the binning lattice and so obtain the randomised binning functions:

$$\theta_{R,\mathbf{w},B}(\mathbf{v}) = \lceil q \cdot \mathbf{v} \cdot B \cdot O \cdot R + \mathbf{w} \rfloor.$$

Randomised binning is again reversible and is preferable to simple binning since, over many random choice of parameters R and $\mathbf{w}$, the likelihood of two vectors going into the same bin depends *only* on their separation (relative to q).

3.5. **Fingerprint generator.** We are now in a position to describe the construction of the quantum fingerprint of a real lattice. Each of the steps in the process may be rendered as a fully precise unitary map.

1. Make the lattice Gram-Schmidt basis and isometry:
$$|k, v\rangle\,|0\rangle \mapsto |k, v\rangle\,|B, O\rangle$$

2. Make the superposition for the index x:
$$|0\rangle \mapsto \frac{1}{\sqrt{\lceil l \rceil}} \sum_{x=0}^{\lceil l \rceil - 1} |x\rangle$$

3. Use the enumeration function ϕ_B to compute $\mathbf{v} = \phi_B(x)$:
$$|B, O\rangle\,|x\rangle\,|0\rangle \mapsto |B, O\rangle\,|x\rangle\,|\phi_B(x)\rangle$$

4. Apply simple binning to compute the label $\mathbf{u} = \theta_B(\mathbf{v})$:
$$|B, O\rangle\,|\mathbf{v}\rangle\,|0\rangle \mapsto |B, O\rangle\,|\mathbf{v}\rangle\,|\theta_B(\mathbf{v})\rangle$$

5. Uncompute the index x exactly using ζ_B by reversing the following process:
$$|B, O\rangle\,|0\rangle\,|\mathbf{v}\rangle \mapsto |B, O\rangle\,|\lceil \zeta_B(\mathbf{v}) \rceil - 1\rangle\,|\mathbf{v}\rangle\,.$$

6. Uncompute the lattice coordinates $\mathbf{v}$ exactly using η_B by reversing the following process:
$$|B,O\rangle\,|0\rangle\,|\mathbf{u}\rangle \mapsto |B,O\rangle\,|\eta_B(\mathbf{u})\rangle\,|\mathbf{u}\rangle\,.$$
7. Finally, uncompute the basis B and isometry O by reversing the map used in the first step.

Putting all of these steps together we obtain a unitary map

$$|k,v\rangle\,|0\rangle \quad\mapsto\quad \frac{1}{\sqrt{\lceil l\rceil}}\sum_{x=0}^{\lceil l\rceil-1}|k,v\rangle\,\left|\theta_{B(k,v)}(\phi(x))\right\rangle$$

and the (simple) *quantum fingerprint* of (k,v) is the associated pure state

$$\left|\psi_{(k,v)}\right\rangle \quad=\quad \frac{1}{\sqrt{\lceil l\rceil}}\sum_{x=0}^{\lceil l\rceil-1}\left|\theta_{B(k,v)}(\phi(x))\right\rangle.$$

The coherent randomised version is:

$$\left|\psi'_{(k,v)}\right\rangle \quad=\quad \frac{\sum_R\sum_{\mathbf{w}}\sum_{x=0}^{\lceil l\rceil-1}|R\rangle\,|\mathbf{w}\rangle\,\left|\theta_{R,\mathbf{w},B(k,v)}(\phi(x))\right\rangle}{\sqrt{\#_R\cdot\#_{\mathbf{w}}\cdot\lceil l\rceil}}$$

3.6. **Fidelity.** The fingerprint structure allows us to define (for fixed l and q) a *fidelity* between two different descriptions of lattice bases

$$Fid(\;(k,v),(k,v)'\;) \quad:=\quad \left\langle\psi'_{(k,v)}\mid\psi'_{(k,v)'}\right\rangle.$$

A fidelity of 1 would indicate that $C_r(B,l)\cdot B\cdot O$ and $C_r(B',l)\cdot B'\cdot O'$, activate exactly the same set of bins for every $(R,\mathbf{w})$-binning strategy and so lattices must be very similar, or identical. A fidelity slightly less than q would be expected if the two lattices were essentially identical, but with different bases causing $C_r(B,l)\cdot B\cdot O$ and $C_r(B',l)\cdot B'\cdot O'$ to vary at ranges between $\rho-\varepsilon$ and $\rho+\varepsilon$ from the origin. When the two lattices are 'essentially different', there is no reason to expect significant overlap in any region, and so the fidelity should be small.

The idea is that, for correctly chosen (l,q), the numerical instability arising from computing $F(k,v)$ is removed by the binning strategy, as (real, infinite) $F(k,v)$ is replaced with (discrete, bounded) $\psi_{(k,v)}$.

4. Conclusion

One conclusion of this work is that designing quantum-resistant cryptography is a difficult task. It took us several years to develop Soliloquy and assess its security against classical attacks, and we had high hopes for quantum resistance at the start of the project. It took us several more years to investigate its potential quantum resistance, and we gather that other investigators in this field have also found making progress to be time-consuming.

As of late 2014, when novel types of quantum-resistant cryptography are being developed for real-world deployment, we caution that much care and patience will be required to ensure that each design receives a thorough security assessment.

Since issues to do with computational precision are overcome by careful algorithm construction, it would seem that quantum algorithms for resolving Abelian Hidden Subgroup Problems have broader applicability to cryptography than 'traditionally' documented. We are now in a position to be able to 'see' why it is unwise to base cryptographic security on problems such as hidden units in (the maximal order of) a number-field or hidden generators of a principal ideal. Fortunately, much of the mathematics relevant to understanding cyclotomic number fields is also of cryptographic relevance for schemes such as Ring-LWE [8], where the underlying hard problem is of a substantially different nature.

References

[1] J. Buchmann, *A subexponential algorithm for the determination of class groups and regulators of algebraic number fields*, Seminaire de theorie des nombres. Paris, 1989, pp. 28-41.

[2] K. Eisenträger, S. Hallgren, A. Kitaev & F. Song, *A quantum algorithm for computing the unit group of an arbitrary degree number field*, Proceedings of the 46th annual ACM Symposium on the Theory of Computing. ACM, 2014, pp. 293-302.

[3] C. Gentry & S. Halevi, *Implementing Gentry's Fully-Homomorphic Encryption scheme*, Advances in Cryptology - EUROCRYPT 2011. Springer Berlin, 2011, pp. 129-148.

[4] O. Goldreich, S. Goldwasser & S. Halevi, *Public-key cryptosystems from lattice reduction problems*, Advances in Cryptology - CRYPTO 1997. Springer Berlin, 1997, pp. 112-131.

[5] S. Hallgren, *Polynomial-time quantum algorithms for Pell's equation and the principal ideal problem*, Proceedings of the 34th annual ACM Symposium on Theory of Computing. ACM, 2002, pp. 653-458.

[6] S. Hallgren, *Fast quantum algorithms for computing the unit group and class group of a number field*, Proceedings of the 37th annual ACM Symposium on Theory of Computing. ACM, 2005, pp. 468-474.

[7] N. Howgrave-Graham & M. Syzdlo, *A method to solve cyclotomic norm equations* $f * \bar{f}$, Algorithmic Number Theory. Springer Berlin, 2004, 272-279.

[8] V. Lyubashevsky, C. Peikert & O. Regev, *A toolkit for Ring-LWE cryptography*, Advanced in Cryptology - EUROCRYPT 2013. Springer Berlin, 2013, pp. 35-54.

[9] N.P. Smart & F. Vercauteren, *Fully Homomorphic Encryption with relatively small key and ciphertext sizes*, Public Key Cryptography - PKC 2010. Springer Berlin, 2010, pp. 420-443.

[10] C. Thiel, *On the complexity of some problems in algorithmic algebraic number theory*, Doctoral Thesis, Saarbrüken, 1995.