Post-Quantum Elliptic Curve Cryptography

Vladimir Soukharev

InfoSec Global

1 Introduction

Practical quantum technologies, that would allow to build a large-scale quantum computer, have been actively emerging. According to some experts in the area, it might take another 15-20 years to be able to build one. Quantum computers will open new capabilities for the world. The list of benefits is impressive. However, in the hands of malicious adversaries, quantum computer could become a real threat. All of today's standardised public-key cryptography could be efficiently broken by large-scale quantum computers. It is vitally important to develop protection against this threat now or in the near future. Quantum-resistant cryptographic algorithms should be developed and implemented well before the arrival of quantum computer, otherwise it will be too late for many for many areas in secure data protection and communication. As it is not yet feasible to use quantum-based techniques, the solution is Post-Quantum Cryptography, classical cryptographic schemes that would be quantum-resistant.

There are five major candidates for Post-Quantum Cryptography, namely, those are: Elliptic Curve Isogeny-Based Cryptography, Hash-Based Signatures, Lattice-Based Cryptography, Code-Based Systems, and Multivariate Polynomials-Based Systems. The schemes based on isogenies can be viewed as a continuation of Elliptic Curve Cryptography, but as a Post-Quantum continuation. The underlying hard problem for isogeny-based cryptography is *given two isogenous supersingular elliptic curves, find an isogeny between them.* Currently no quantum algorithm is known for solving this problem in general in less than exponential time. One of the main reasons why this problem seems intractable for quantum computers is that the endomorphism ring for the elliptic curve is non-commutative, which shields the problem against attacks like Shor's algorithm. Compared to other post-quantum proposals, this approach would be one of the easiest drop-in replacements for the current cryptographic infrastructure. It also has the shortest key size. Besides that, it is based on elliptic curves, hence it is something the cryptographic industrial world has already partially seen and a lot of code can be reused.

The ciphersuite ideally has the following three major components: key agreement, public-key encryption, and a digital signature. The first two components were the first isogeny-based cryptographic schemes developed a few years ago. Several authentication-related protocols have been developed. Recently, the digital signature has been derived and shown to be secure in the quantum random oracle model.

2 Background

In this section, we will examine some background needed to understand the schemes.

We define $E[m]$ to be the set of points in E, such that their order divides m. That is, if $P \in E[m]$, then $mP = \infty$ (identity point). We form our prime to be of the form $p = \ell_A^{e_A} \cdot \ell_B^{e_B} \cdot f \pm 1$, where ℓ_A and ℓ_B are small primes and f is a cofactor. Supersingular elliptic curves are defined over $\mathbb{F}_{p^2}$. When m divides the order of the curve, the group structure of $E[m]$ is isomorphic to $(\mathbb{Z}_m)^2$, hence it needs two elements which are elliptic curve points to generate the entire $E[m]$.

For our purposes, we will need $E[\ell_A^{e_A}]$, which will have generators P_A, Q_A and $E[\ell_B^{e_B}]$, which will have generators P_B, Q_B. This can be extended beyond two-prime construction. In practice, we use two and three prime constructions.

In general, a private key for user A is two scalar values $m_A, n_A \in \mathbb{Z}_{\ell_A^{e_A}}$ (modulo $\ell_A^{e_A}$). These values are used to compute an elliptic curve point $K_A = m_A P_A + n_A Q_A$. The point K_A is used as the generator of the kernel, denoted $\langle K_A \rangle$, of the isogeny to compute the corresponding isogeny ϕ_A itself. In practice, we do not explicitly state isogenies, but use their kernels to compute them. Let E_A be the resulting curve to which isogeny ϕ_A maps, that is $\phi_A \colon E \to E_A$. E_A is exactly the public key corresponding to m_A, n_A.

Given an isogeny $\phi \colon E \to E'$, and a point $aP_1 + bP_2$, where $P_1, P_2 \in E$, we know that $\phi(a \cdot P_1 + b \cdot P_2) = \phi(a \cdot P_1) + \phi(b \cdot P_2) = a \cdot \phi(P_1) + b \cdot \phi(P_2) \in E'$.

3 Key Agreement

For the key exchange scheme, provided in [1], we have two users A and B. The starting elliptic curve E is public, as well as, the bases points $\{P_A, Q_A\}$ and $\{P_B, Q_B\}$.

Key Generation. Alice does the following:

1. Randomly selects $m_A, n_A \in \mathbb{Z}_{\ell_A^{e_A}}$.
2. Computes $K_A = m_A P_A + n_A Q_A$.
3. Obtains E_A using the kernel $\langle K_A \rangle$ for the corresponding isogeny $\phi_A \colon E \to E_A$.
4. Computes the values of P_B and Q_B under her isogeny ϕ_A, namely $\phi_A(P_B)$ and $\phi_A(Q_B)$. (These are referred to as *auxiliary points.*)
5. Publishes E_A and auxiliary points $\phi_A(P_B)$ and $\phi_A(Q_B)$.

Bob does the same symmetrically generating private key $m_B, n_B \in \mathbb{Z}_{\ell_B^{e_B}}$, computing and publishing the corresponding public key: elliptic curve E_B and auxiliary points $\phi_B(P_A)$ and $\phi_B(Q_A)$.

Obtaining The Shared Key. Alice does the following:

1. Using her private key values m_A, n_A and Bob's auxiliary points $\phi_B(P_A)$, $\phi_B(Q_A)$, computes $m_A \cdot \phi_B(P_A) + n_A \cdot \phi_B(Q_A)$. Note that $m_A \cdot \phi_B(P_A) + n_A \cdot \phi_B(Q_A) = \phi_B(m_A \cdot P_A) + \phi_B(n_A \cdot Q_A) = \phi_B(m_A \cdot P_A + n_A \cdot Q_A) = \phi_B(K_A)$. Hence it is the image of Alice's kernel generator point in Bob's curve E_B.
2. Using that value as the generator point for the kernel $\langle \phi_B(K_A) \rangle$, Alice maps $E_B \to E_{AB}$.
3. Computes the j-invariant of E_{AB} and uses that as a value of the common key.

Bob does the same symmetrically computing $m_B \cdot \phi_A(P_B) + n_B \cdot \phi_A(Q_B) = \langle \phi_A(K_B) \rangle$ and using it to obtain maps $E_A \to E_{BA}$. Finally, he computes the j-invariant of E_{BA} and uses that as a value of the common key.

Note that the curves E_{AB} and E_{BA} are isomorphic, which means that they have the same j-invariants. If two elliptic curves are isomorphic, they are considered to be the same curve.

Figure 1 summarises the above description.

Notice that this approach in the *á la Diffie-Hellman* style.

4 Public Key Encryption Scheme

The public key encryption scheme, based on supersingular isogenies, uses the same primitives as the key exchange scheme and then the value of j-invariant of E_{AB} is hashed and XOR'ed with the message.

5 Digital Signature

We present the first general-purpose digital signature scheme based on supersingular elliptic curve isogenies. The scheme is secure against quantum adversaries in the quantum random oracle model, has small key sizes, and can take advantage of offline computation to sign efficiently. This scheme is

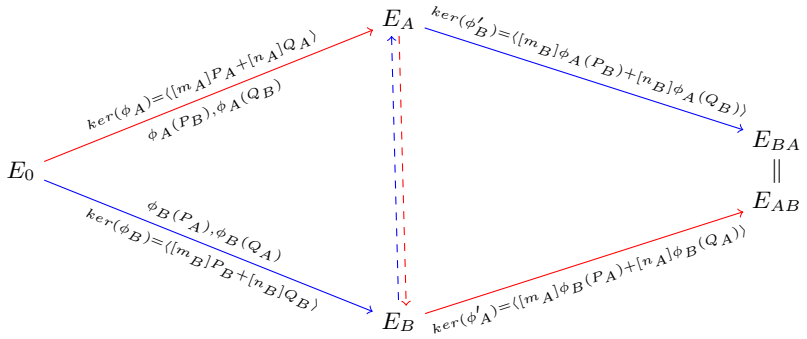

Fig. 1: Key-exchange protocol using isogenies on supersingular curves.

an application of Unruh's [2] construction of non-interactive zero-knowledge proofs to an interactive zero-knowledge proof proposed in [1].

Zero-Knowledge Proof of Identity is shown in the following diagram.

$$\begin{array}{ccc} E & \xrightarrow{\phi} & E/\langle S \rangle \\ \psi \downarrow & & \downarrow \psi' \\ E/\langle R \rangle & \xrightarrow{\phi'} & E/\langle S, R \rangle \end{array}$$

The prover needs to prove that he possesses a secret ϕ, by computing ψ and providing $E/\langle R \rangle$ and $E/\langle S, R \rangle$. When asked, provides either ψ, ψ' or ϕ', depending on the request by verifier. Hence, in one round there is a $1/2$ chance of cheating.

To obtain the digital signature, the signer must combine protocol and Unruh's construction, namely precompute the diagram for sufficient number of rounds. (usually twice the number of quantum bits of security required). Then, the idea is as follows: the signer hashes the public knowledge parts with the message and based on the output of the hash (as a bitstring of length that corresponds to the number of rounds required), provides corresponding responses. All of that forms the signature. More details can be found in [3].

6 Conclusion

The now complete set of cryptographic schemes shows that elliptic curves can be used as a protection against quantum computers. The emergence of quantum computers will bring many benefits to the society. However, in the hands of adversary they will become a threat to security. Thus, in order to prevent that threat, we must start the transition to quantum-resistant cryptographic protocols as soon as possible. In reality, the transition is a long and complicated process. Elliptic curve isogeny-based schemes have the properties, which will allow the smoothest transition, compared to all the other post-quantum candidates.

References

1. Luca De Feo, David Jao, and Jérôme Plût. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *J. Mathematical Cryptology*, 8(3):209–247, 2014.
2. Dominique Unruh. *Non-Interactive Zero-Knowledge Proofs in the Quantum Random Oracle Model*, pages 755–784. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.
3. Youngho Yoo, Reza Azarderakhsh, Amir Jalali, David Jao, and Vladimir Soukharev. A post-quantum digital signature scheme based on supersingular isogenies. Cryptology ePrint Archive, Report 2017/186, 2017. `http://eprint.iacr.org/2017/186`.