QDay Prize

Quantum Circuit Overview

The following quantum circuit was constructed based on the Proos–Zalka algorithm to break the cryptography.
Given the ECC parameter G and the public key $Q = dG$, Shor's algorithm is used with $aG + bQ$ as the Oracle. For details of the Oracle, see this report.

```
        ┌───┐                         ░ ┌──────┐┌─┐    
   qa: ─┤ H ├───────■─────────────────░─┤ QFT† ├┤M├────
        ├───┤       │                 ░ ├──────┤└╥┘┌─┐ 
   qb: ─┤ H ├───────■─────────────────░─┤ QFT† ├─╫─┤M├─
        └───┘┌──────┴─────────┐┌─┐    ░ └──────┘ ║ └╥┘ 
   qx: ──────┤0               ├┤M├────░──────────╫──╫──
             │                │└╥┘┌─┐ ░          ║  ║  
   qy: ──────┤1               ├─╫─┤M├─░──────────╫──╫──
             │  oracle: aG+bQ │ ║ └╥┘ ░          ║  ║  
anc_0: ──────┤2               ├─╫──╫──░──────────╫──╫──
             │                │ ║  ║  ░          ║  ║  
anc_1: ──────┤3               ├─╫──╫──░──────────╫──╫──
             └────────────────┘ ║  ║  ░          ║  ║  
  a: 1/═════════════════════════╬══╬═════════════╩══╬══
                                ║  ║             0  ║  
  b: 1/═════════════════════════╬══╬════════════════╩══
                                ║  ║                0  
 xy: 2/═════════════════════════╩══╩═══════════════════
                                0  1                   
```

The quantum circuit consists of the following steps:

1. Prepare qa and qb with the number of bits of the order n.
2. Apply H gates to qa and qb to create a superposition.
3. Execute the quantum circuit so that $(qx, qy) = aG + bQ$.
4. Measure (qx, qy) to collapse to a single coordinate.
5. Apply the inverse quantum Fourier transform ($QFT^{\dagger}$) to qa and qb.
6. Finally, measure qa and qb.

The private key is obtained from the measured values a and b as follows:

$$\mathrm{len} = \lfloor \log_2 n \rfloor \tag{1}$$

$$x = \mathrm{floor}\left(\frac{a \cdot n}{2^{\mathrm{len}}}\right), \mathrm{ceil}\left(\frac{a \cdot n}{2^{\mathrm{len}}}\right) \tag{2}$$

$$y = \mathrm{floor}\left(\frac{b \cdot n}{2^{\mathrm{len}}}\right), \mathrm{ceil}\left(\frac{b \cdot n}{2^{\mathrm{len}}}\right) \tag{3}$$

$$d = x^{-1} \cdot y \pmod n \tag{4}$$

1. Let len be the number of bits of the order n.
2. Scale a to fit the range from 2^{len} to n.
3. Similarly, scale b.
4. Find the private key d by calculating the inverse of x, x^{-1}, and multiplying it by y.

To account for rounding errors during scaling, all four combinations of floor and ceiling for x and y are calculated, and it is checked whether any of them satisfy the private key condition.

From simulation, the probability of obtaining a measurement result that leads to the correct answer is about 60% to 70% when the number of bits of the order is small, but when the number of bits increases, the probability exceeds 80%.
On a real quantum computer (ibm_torino), the result was random due to noise.

Required Resources

In this work, a general-purpose quantum circuit is constructed to work correctly for any ECC parameter, so a relatively large number of quantum gates are required.
For the ECC Curves and Keys presented this time, the required number of quantum bits and quantum gates[1] are as follows.
The circuit creation time and simulation time on an iMac (Apple M1, Memory 16GB) are also shown for reference.

ecc bits	prime	order	quantum bits	gate count[1]	circuit creation	simulation
4	13	7	78	48261	1.6s	0.6s
6	43	31	152	452274	19.9s	10.7s
7	67	79	221	976577	48.8s	51.7s

ecc bits	prime	order	quantum bits	gate count[1]	circuit creation	simulation
8	163	139	276	2240599	126s	5m 27s
9	349	313	337	6347702	418s	1h 23m 54s
10	547	547	404	7000037	562s	9h 15m 51s
11	1051	1093	477	14783371	1355s	76h 44m 39s

It is estimated that about 200,000 quantum bits are required to break secp256k1 used in Bitcoin.

[1] Since multi-controlled gates are counted as one quantum gate, the actual number of quantum gates will be larger when transpiled for a real quantum computer.