# You could have invented Supersingular Isogeny Diffie-Hellman

Lorenz Panny

Technische Universiteit Eindhoven

Πλατανιάς, Κρήτη, 11 October 2017

# Shor's algorithm '94

Shor's algorithm quantumly breaks Diffie-Hellman
in any group in polynomial time.

# Shor's algorithm '94

Shor's algorithm quantumly breaks Diffie-Hellman
in any group in polynomial time.

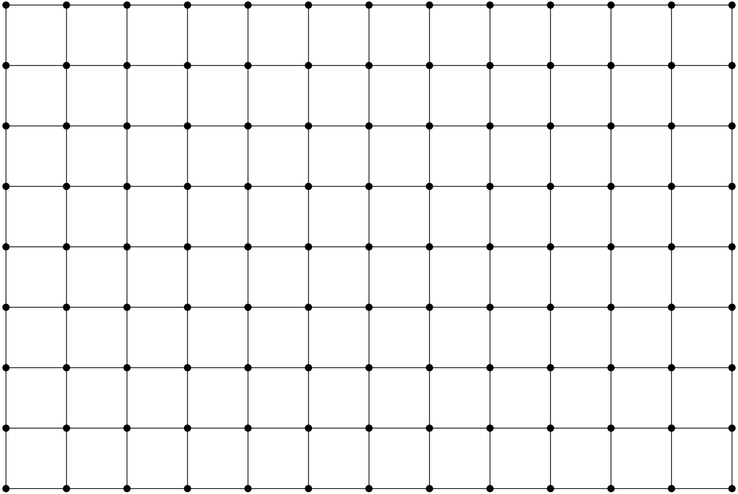
But mathematicians fancy elliptic curves... What do?

# Graph walking Diffie-Hellman?

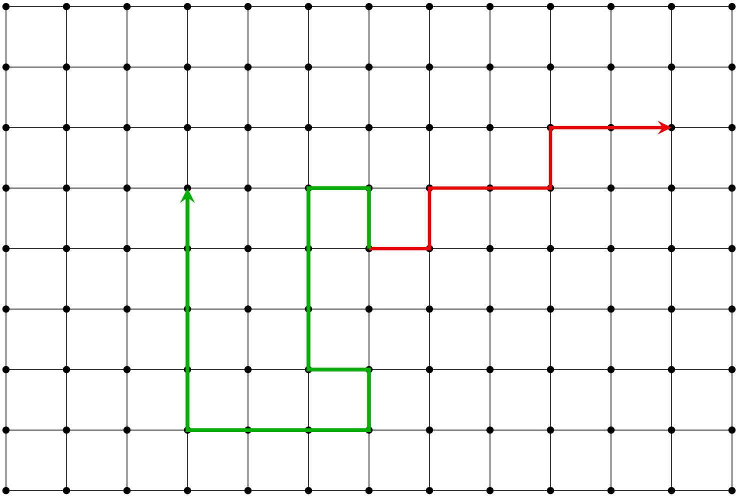
Imagine...

- We have a finite graph and some starting node
- There is a set of 'directions' for navigating the graph
- Alice and Bob do Diffie-Hellman using secret paths

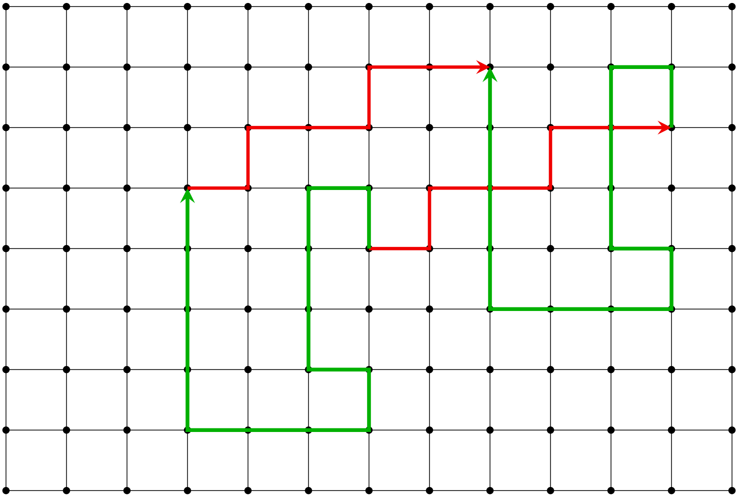
# Graph walking Diffie-Hellman?



# Graph walking Diffie-Hellman?



# Graph walking Diffie-Hellman?



# Graph walking Diffie-Hellman?

Imagine...

- We have a finite graph and some starting node
- There is a set of 'directions' for navigating the graph
- Alice and Bob do Diffie-Hellman using secret paths

# Graph walking Diffie-Hellman?

Imagine...

- We have a finite graph and some starting node
- There is a set of 'directions' for navigating the graph
- Alice and Bob do Diffie-Hellman using secret paths

It should be hard to recover the path given the end points.
$\Longrightarrow$ The graph must be 'random' and exponentially large.

# Graph walking Diffie-Hellman?

Imagine...

- We have a finite graph and some starting node
- There is a set of 'directions' for navigating the graph
- Alice and Bob do Diffie-Hellman using secret paths

It should be hard to recover the path given the end points.
$\Longrightarrow$ The graph must be 'random' and exponentially large.

How to make sure Alice and Bob arrive at the same end point?

# Graph walking?

Stand back!

We're going to do math.

# Elliptic curves

An elliptic curve (modulo details) is given by an equation

$$E\colon\ y^2 = x^3 + ax + b.$$

A point on $E$ is a solution to this equation or $\infty$.
Isomorphism classes are identified by their $j$-invariant.

# Elliptic curves

An elliptic curve (modulo details) is given by an equation

$$E\colon\; y^2 = x^3 + ax + b.$$

A point on $E$ is a solution to this equation or $\infty$.
Isomorphism classes are identified by their $j$-invariant.

$E$ is an abelian group: we can 'add' points.

- The neutral element is $\infty$.
- The inverse of $(x, y)$ is $(x, -y)$.
- The sum of $(x_1, y_1)$ and $(x_2, y_2)$ is

$$\left(\lambda^2 - x_1 - x_2, \lambda(2x_1 + x_2 - \lambda^2) - y_1\right)$$

where $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$ if $x_1 \neq x_2$ and $\lambda = \frac{3x_1^2 + a}{2y_1}$ otherwise.

# Isogenies

An isogeny of elliptic curves is a non-constant map $E \to E'$

- given by rational functions
- that is a group homomorphism

The degree of a separable[1] isogeny is the size of its kernel.

[1] Over $\mathbb{F}_q$, this means it does not factor through Frobenius $(x, y) \mapsto (x^q, y^q)$.

# Isogenies

An isogeny of elliptic curves is a non-constant map $E \to E'$

- given by rational functions
- that is a group homomorphism

The degree of a separable[1] isogeny is the size of its kernel.

Example: For each $m \neq 0$, the multiplication-by-$m$ map

$$[m]\colon E \to E$$

is a degree-$m^2$ isogeny. If $m \neq 0$ in the base field, its kernel is

$$E[m] \cong \mathbb{Z}/m \times \mathbb{Z}/m.$$

[1] Over $\mathbb{F}_q$, this means it does not factor through Frobenius $(x, y) \mapsto (x^q, y^q)$.

# Isogenies

An isogeny of elliptic curves is a non-constant map $E \to E'$

- given by rational functions
- that is a group homomorphism

The degree of a separable[1] isogeny is the size of its kernel.

Example: $(x, y) \mapsto \left( \frac{x^3-4x^2+30x-12}{(x-2)^2}, \frac{x^3-6x^2-14x+35}{(x-2)^3} \cdot y \right)$
defines a degree-3 isogeny of the elliptic curves

$$\{y^2 = x^3 + x\} \longrightarrow \{y^2 = x^3 - 3x + 3\}$$

over $\mathbb{F}_{71}$. Its kernel is $\{(2, 9), (2, -9), \infty\}$.

[1] Over $\mathbb{F}_q$, this means it does not factor through Frobenius $(x, y) \mapsto (x^q, y^q)$.

# Isogeny graphs

Fix a prime power $q$ and an integer $\ell \geq 2$.

The $\ell$-isogeny graph over $\mathbb{F}_q$ consists of the following data:

- Nodes: isomorphism classes of elliptic curves $/\mathbb{F}_q$.
- Edges: equivalence classes[1] of degree-$\ell$ isogenies.

[1] Two isogenies $\varphi\colon E \to E'$ and $\psi\colon E \to E''$ are identified if $\psi = \iota \circ \varphi$ for some isomorphism $\iota\colon E' \to E''$.

# Isogeny graphs

Fix a prime power $q$ and an integer $\ell \geq 2$.

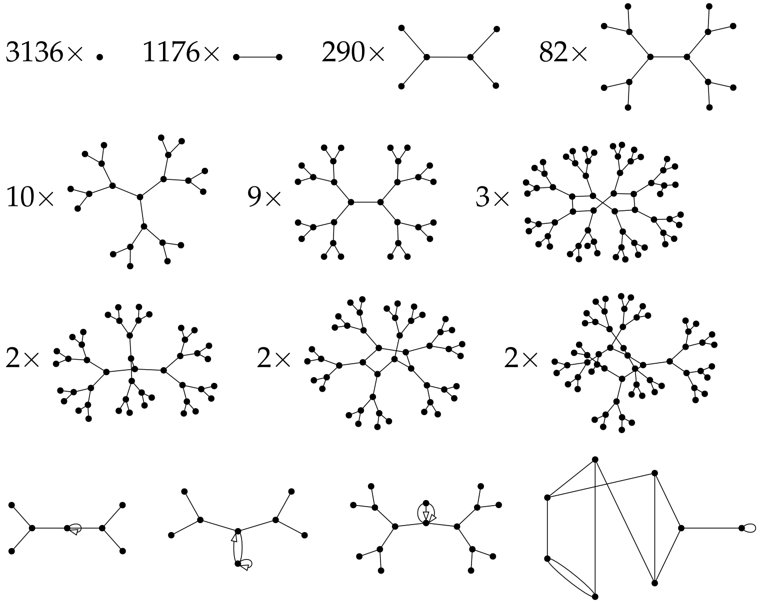
The $\ell$-isogeny graph over $\mathbb{F}_q$ consists of the following data:

- Nodes: isomorphism classes of elliptic curves $/\mathbb{F}_q$.
- Edges: equivalence classes[1] of degree-$\ell$ isogenies.

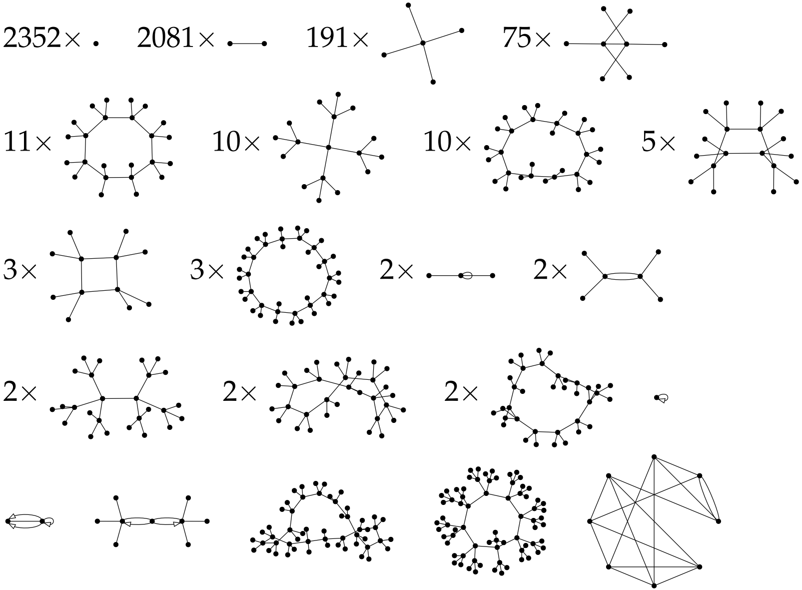
The $\ell$-isogeny graph is an undirected multigraph
except for edges touching the $j$-invariants 0 or 1728.

[1] Two isogenies $\varphi\colon E \to E'$ and $\psi\colon E \to E''$ are identified if $\psi = \iota \circ \varphi$ for some isomorphism $\iota\colon E' \to E''$.

# 2-isogeny graph over $\mathbb{F}_{97^2}$



# 3-isogeny graph over $\mathbb{F}_{97^2}$



# Supersingular elliptic curves

An elliptic curve $E/\bar{\mathbb{F}}_p$ is supersingular if $E[p] = \{\infty\}$.

# Supersingular elliptic curves

An elliptic curve $E/\bar{\mathbb{F}}_p$ is supersingular if $E[p] = \{\infty\}$.

If $p \geq 5$, then $E/\mathbb{F}_p$ is supersingular iff $\#E(\mathbb{F}_p) = p + 1$.

# Supersingular elliptic curves

An elliptic curve $E/\bar{\mathbb{F}}_p$ is supersingular if $E[p] = \{\infty\}$.

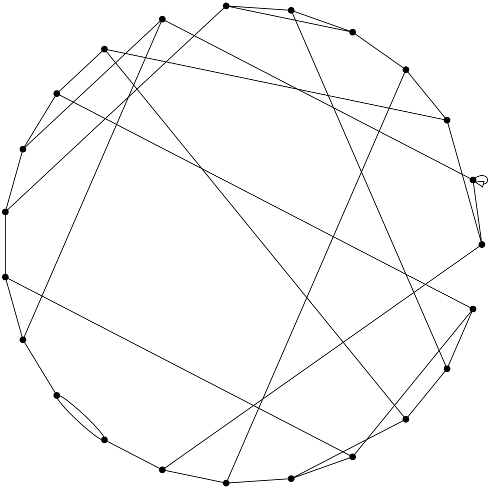
If $p \geq 5$, then $E/\mathbb{F}_p$ is supersingular iff $\#E(\mathbb{F}_p) = p + 1$.

Every supersingular elliptic curve is defined over $\mathbb{F}_{p^2}$.

# Supersingular isogeny graphs

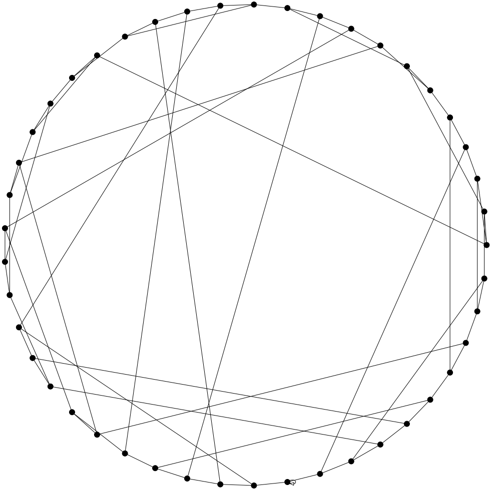
The supersingular elliptic curves form a component of the $\ell$-isogeny graph over $\mathbb{F}_{p^2}$, the supersingular $\ell$-isogeny graph.

# Supersingular isogeny graphs



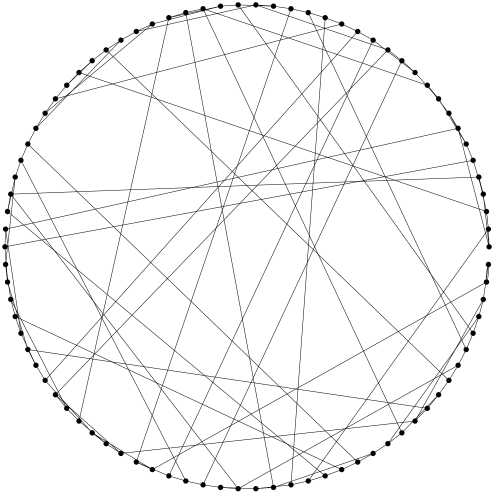
$p = 277, \ell = 2$

# Supersingular isogeny graphs



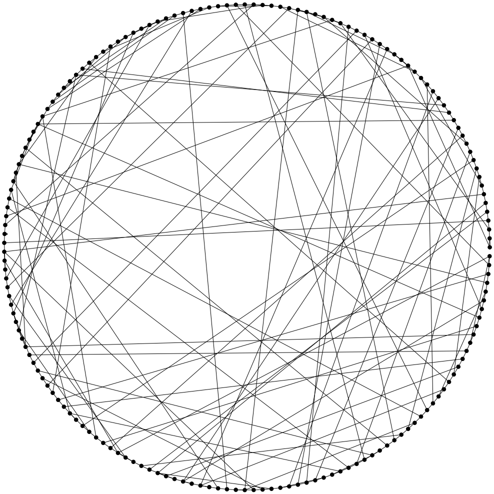
$p = 541, \ell = 2$

# Supersingular isogeny graphs



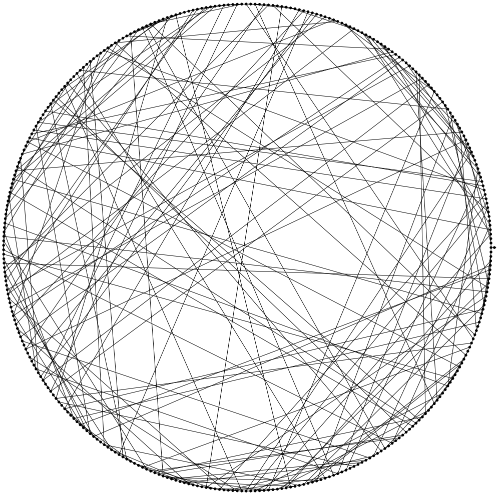
$p = 1033, \ell = 2$

# Supersingular isogeny graphs



$p = 2053, \ell = 2$

# Supersingular isogeny graphs



$p = 4129, \ell = 2$

# Supersingular isogeny graphs

The supersingular elliptic curves form a component of the $\ell$-isogeny graph over $\mathbb{F}_{p^2}$, the supersingular $\ell$-isogeny graph.

# Supersingular isogeny graphs

The supersingular elliptic curves form a component of the $\ell$-isogeny graph over $\mathbb{F}_{p^2}$, the supersingular $\ell$-isogeny graph.

There are $\lfloor p/12 \rfloor + \varepsilon$ supersingular elliptic curves over $\bar{\mathbb{F}}_p$.

# Supersingular isogeny graphs

The supersingular elliptic curves form a component of the $\ell$-isogeny graph over $\mathbb{F}_{p^2}$, the supersingular $\ell$-isogeny graph.

There are $\lfloor p/12 \rfloor + \varepsilon$ supersingular elliptic curves over $\bar{\mathbb{F}}_p$.

- $y^2 = x^3 + 1$ is supersingular iff $p \equiv -1 \pmod 3$.
- $y^2 = x^3 + x$ is supersingular iff $p \equiv -1 \pmod 4$.

# Supersingular isogeny graphs

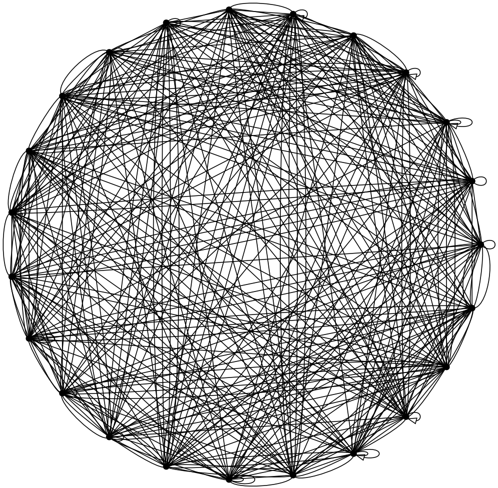
The supersingular elliptic curves form a component of the $\ell$-isogeny graph over $\mathbb{F}_{p^2}$, the supersingular $\ell$-isogeny graph.

There are $\lfloor p/12 \rfloor + \varepsilon$ supersingular elliptic curves over $\bar{\mathbb{F}}_p$.

- $y^2 = x^3 + 1$ is supersingular iff $p \equiv -1 \pmod 3$.
- $y^2 = x^3 + x$ is supersingular iff $p \equiv -1 \pmod 4$.

The supersingular $\ell$-isogeny graph is (almost) Ramanujan. (Almost) all nodes have out-degree $\ell + 1$.

# Supersingular isogeny graphs



$p = 277, \ell = 31$

# Algorithms?

State of this talk:

- Exponentially large ‘random’ graph. ✓
- How to compute on this graph?

# Isogenies and kernels

For any finite subgroup $G$ of $E$, there exists a unique[1] separable isogeny $\varphi_G \colon E \to E'$ with kernel $G$.

The curve $E'$ is called $E/G$.

[1](up to isomorphism of $E'$)

# Vélu's formulas '71

Let $G$ be a finite subgroup of an elliptic curve $E$. Then

$$P \mapsto \left( x(P) + \sum_{\substack{Q \in G \\ Q \neq \infty}} \big( x(P+Q) - x(Q) \big),\ y(P) + \sum_{\substack{Q \in G \\ Q \neq \infty}} \big( y(P+Q) - y(Q) \big) \right)$$

defines an isogeny of elliptic curves whose kernel is $G$.

# Vélu's formulas '71

Let $G$ be a finite subgroup of an elliptic curve $E$. Then

$$P \mapsto \left( x(P) + \sum_{\substack{Q \in G \\ Q \neq \infty}} \big( x(P+Q) - x(Q) \big),\ y(P) + \sum_{\substack{Q \in G \\ Q \neq \infty}} \big( y(P+Q) - y(Q) \big) \right)$$

defines an isogeny of elliptic curves whose kernel is $G$.

For small $G$, this leads to efficient formulas for

- computing the defining equation of $E/G$
- evaluating the isogeny $E \to E/G$ at a point

# Representing isogeny paths

- Storing each curve and kernel on the way is expensive.

$$E \xrightarrow{\psi_1} E_1 \xrightarrow{\psi_2} \dots \xrightarrow{\psi_{n-1}} E_{n-1} \xrightarrow{\psi_n} E/G$$

(It would also make the DH system we're building impossible...)

# Representing isogeny paths

- Storing each curve and kernel on the way is expensive.

$$E \xrightarrow{\psi_1} E_1 \xrightarrow{\psi_2} \dots \xrightarrow{\psi_{n-1}} E_{n-1} \xrightarrow{\psi_n} E/G$$

(It would also make the DH system we're building impossible...)

- Use the kernel of the composition!

$$\begin{tikzcd} E \arrow[r, "\psi_1"] \arrow[rrrr, bend right, "\varphi_G"'] & E_1 \arrow[r, "\psi_2"] & \dots \arrow[r, "\psi_{n-1}"] & E_{n-1} \arrow[r, "\psi_n"] & E/G \end{tikzcd}$$

# Representing isogeny paths

- Storing each curve and kernel on the way is expensive.

$$E \xrightarrow{\psi_1} E_1 \xrightarrow{\psi_2} \ldots \xrightarrow{\psi_{n-1}} E_{n-1} \xrightarrow{\psi_n} E/G$$

(It would also make the DH system we're building impossible...)

- Use the kernel of the composition!

$$\begin{tikzcd} E \arrow[r, "\psi_1"] \arrow[rrrr, bend right, "\varphi_G"'] & E_1 \arrow[r, "\psi_2"] & \ldots \arrow[r, "\psi_{n-1}"] & E_{n-1} \arrow[r, "\psi_n"] & E/G \end{tikzcd}$$

- Evaluate $\varphi_G$ via a chain of small-degree isogenies:
If $G \cong \mathbb{Z}/\ell^n$, set $\ker \psi_i := [\ell^{n-i}](\psi_{i-1} \circ \cdots \circ \psi_1)(G)$.
(This is usually not the optimal strategy.)

# Commutativity?

State of this talk:

- Exponentially large ‘random’ graph. ✓
- Efficient formulas to traverse it. ✓
- How to make Alice and Bob’s walks commute?

# Commutativity?

We want:

$$\begin{tikzcd} & E_A \arrow[dr, "\psi_B"] & \\ E_0 \arrow[ur, "\varphi_A"] \arrow[dr, "\varphi_B"'] & & E \\ & E_B \arrow[ur, "\psi_A"'] & \end{tikzcd}$$

# Commutativity?

We want:

$$\begin{tikzcd} & E_A \arrow[dr, "\psi_B"] & \\ E_0 \arrow[ur, "\varphi_A"] \arrow[dr, "\varphi_B"'] & & E \\ & E_B \arrow[ur, "\psi_A"'] & \end{tikzcd}$$

If only Bob could help Alice by 'shifting' her $\ker \varphi_A$ to $E_B$...
but Alice must keep $\varphi_A$ secret... $\ddot\frown$

# Commutativity!

We want:

$$\begin{tikzcd} & {\color{red}E_A} \arrow[dr, "{\color{green}\psi_B}"] & \\ E_0 \arrow[ur, "{\color{red}\varphi_A}"] \arrow[dr, "{\color{green}\varphi_B}"'] & & E \\ & {\color{green}E_B} \arrow[ur, "{\color{red}\psi_A}"'] & \end{tikzcd}$$

If only Bob could help Alice by 'shifting' her $\ker \varphi_A$ to $E_B$...
but Alice must keep $\varphi_A$ secret... ☹

Solution: Bob 'shifts' a public group that contains $\ker \varphi_A$.

# Commutativity!

We want:

$$\begin{tikzcd} & E_A \arrow[dr, "\psi_B"] & \\ E_0 \arrow[ur, "\varphi_A"] \arrow[dr, "\varphi_B"'] & & E \\ & E_B \arrow[ur, "\psi_A"'] & \end{tikzcd}$$

If only Bob could help Alice by 'shifting' her $\ker \varphi_A$ to $E_B$...
but Alice must keep $\varphi_A$ secret... $\ddot\frown$

Solution: Bob 'shifts' a public group that contains $\ker \varphi_A$.

- Fix some public generator points $P, Q \in E_0[\deg \varphi_A]$.
- Alice computes $\varphi_A \colon E_0 \to E_A$ with kernel $\langle P + [a]Q \rangle$.
- Bob uses $\varphi_B$ to 'shift' $P, Q$ to $E_B$ and gives them to Alice.
- Alice computes $\psi_A$ with kernel $\langle \varphi_B(P) + [a]\varphi_B(Q) \rangle$.
- By ~~magic~~ math, Bob will arrive at an isomorphic $E$.

# The SIDH protocol (De Feo–Jao–Plût 2011)

Public parameters:

- a large prime $p = 2^{n_A}3^{n_B} - 1$ and a supersingular $E_0/\mathbb{F}_p$.
- bases $(P_A, Q_A)$ and $(P_B, Q_B)$ of $E_0[2^{n_A}]$ and $E_0[3^{n_B}]$.

# The SIDH protocol (De Feo–Jao–Plût 2011)

Public parameters:

- a large prime $p = 2^{n_A} 3^{n_B} - 1$ and a supersingular $E_0/\mathbb{F}_p$.
- bases $(P_A, Q_A)$ and $(P_B, Q_B)$ of $E_0[2^{n_A}]$ and $E_0[3^{n_B}]$.

| Alice | public | Bob |
|---|---|---|
| $a \xleftarrow{\text{random}} \{0 \ldots 2^{n_A - 1}\}$ | | $b \xleftarrow{\text{random}} \{0 \ldots 3^{n_B - 1}\}$ |
| $G_A := \langle P_A + [2a] Q_A \rangle$ <br> compute $\varphi_A : E_0 \to E_0/G_A$ | | $G_B := \langle P_B + [3b] Q_B \rangle$ <br> compute $\varphi_B : E_0 \to E_0/G_B$ |
| $\varphi_A(P_B),\ \varphi_A(Q_B)$ → | | ← $\varphi_B(P_A),\ \varphi_B(Q_A)$ |
| recover $E_B = E_0/G_B$ <br> $H_A := \langle \varphi_B(P_A) + [2a] \varphi_B(Q_A) \rangle$ <br> $s := j(E_B/H_A)$ | | recover $E_A = E_0/G_A$ <br> $H_B := \langle \varphi_A(P_B) + [3b] \varphi_A(Q_B) \rangle$ <br> $s := j(E_A/H_B)$ |

# Optimizations

- Projective representation of curve coefficients.[1]
- Distortion map on $E_0$ speeds up public key generation.[1]
- Use of Montgomery model and $x$-only arithmetic.[1]
- Compression reduces public key size to $\frac{7}{2}\log_2 p$ bits.[2]

[1] Costello–Longa–Naehrig 2016, `https://ia.cr/2016/413`

[2] Costello–Jao–Longa–Naehrig–Renes–Urbanik 2016, `https://ia.cr/2016/963`

# Optimizations

- Projective representation of curve coefficients.[1]
- Distortion map on $E_0$ speeds up public key generation.[1]
- Use of Montgomery model and $x$-only arithmetic.[1]
- Compression reduces public key size to $\frac{7}{2}\log_2 p$ bits.[2]

Current performance records:[2]

| | Public keys | Cycles | Wall-clock time |
|---|---|---|---|
| uncompressed | 564 bytes | $192 \cdot 10^6$ | $\approx 50$ ms |
| compressed | 330 bytes | $469 \cdot 10^6$ | $\approx 150$ ms |

(Parameters aimed at 192 bits of classical and 128 bits of quantum security.)

[1] Costello–Longa–Naehrig 2016, `https://ia.cr/2016/413`
[2] Costello–Jao–Longa–Naehrig–Renes–Urbanik 2016, `https://ia.cr/2016/963`

# Security

The security of SIDH depends on the hardness of..:

- Computing *an* isogeny between two given curves.[1]
- ...when the images of some points are known.[2]
- Computing the endomorphism ring of a given curve.[3]

[1] Galbraith–Petit–Shani–Ti 2016, https://ia.cr/2016/859

[2] Petit 2017, https://ia.cr/2017/571

[3] Kohel–Lauter–Petit–Tignol 2014, https://arxiv.org/abs/1406.0981

# Security

The security of SIDH depends on the hardness of..:

- Computing *an* isogeny between two given curves.[1]
- ...when the images of some points are known.[2]
- Computing the endomorphism ring of a given curve.[3]

Best known attacks: $\mathcal{O}(p^{1/4})$ classically and $\mathcal{O}(p^{1/6})$ quantumly.

[1] Galbraith–Petit–Shani–Ti 2016, `https://ia.cr/2016/859`

[2] Petit 2017, `https://ia.cr/2017/571`

[3] Kohel–Lauter–Petit–Tignol 2014, `https://arxiv.org/abs/1406.0981`

# Security

The security of SIDH depends on the hardness of..:

- Computing *an* isogeny between two given curves.[1]
- ...when the images of some points are known.[2]
- Computing the endomorphism ring of a given curve.[3]

Best known attacks: $\mathcal{O}(p^{1/4})$ classically and $\mathcal{O}(p^{1/6})$ quantumly.

Caution! If Bob reuses his key pair, Alice can recover his private key in $\mathcal{O}(\log p)$ queries.[1]

[1] Galbraith–Petit–Shani–Ti 2016, `https://ia.cr/2016/859`

[2] Petit 2017, `https://ia.cr/2017/571`

[3] Kohel–Lauter–Petit–Tignol 2014, `https://arxiv.org/abs/1406.0981`

# Open problems

- How can we *cheaply* reuse key pairs?
- Will this ever be really fast?

# Open problems

- How can we *cheaply* reuse key pairs?
- Will this ever be really fast?

- Is this scheme actually secure?
  Are there weak parameters, side channels, fault attacks, ..?

Thank you!