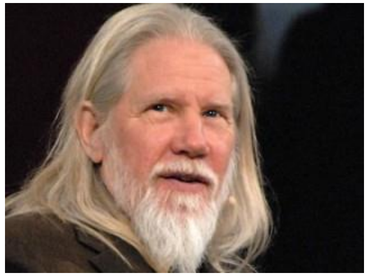
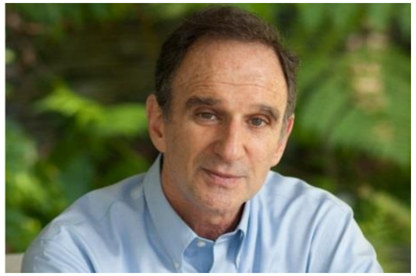
# Curves and Fields for Efficient Cryptographic Pairings

Michael Naehrig
XCG Cryptography Research Group
Microsoft Research

Pacific Northwest Number Theory Conference 2013
Seattle, 1 June 2013

# Public-Key Cryptography

- Diffie-Hellman key agreement (1976)
- Cyclic group $(G, +)$, $G = \langle P \rangle$, prime order $\mathrm{r} = |G|$
- $[m]P = \underbrace{P + P + \cdots + P}_{m \text{ times}}$

$P_A$

**Alice**

Secret $a \in \mathbf{Z}/r\mathbf{Z}$

$P_A = [a]P$

$P_B = [b]P$

$\boxed{\mathrm{K} = [a]P_B = [b]P_A = [ab]P}$

$P_B$

**Bob**

Secret $b \in \mathbf{Z}/r\mathbf{Z}$

$P_B = [b]P$

$P_A = [a]P$

# Public-Key Cryptography

- Cyclic group $(G, +)$, $G = \langle P \rangle$, prime order $\mathrm{r} = |G|$,
- Diffie-Hellman Problem (DHP) in $G$ :
  given $\mathrm{P_A} = [a]P$ and $\mathrm{P_B} = [b]P$, find $[ab]P$.
- Discrete Logarithm Problem (DLP) in $G$ : given $\mathrm{P_A} = [a]P$, find $a$.
- For security, DHP/DLP in $G$ must be computationally infeasible.

$P_A$

$\boxed{\mathrm{K} = [a]P_B = [b]P_A = [ab]P}$

$P_B$

**Alice**

Secret $a \in \mathbf{Z}/r\mathbf{Z}$

$P_A = [a]P$

$P_B = [b]P$

**Bob**

Secret $b \in \mathbf{Z}/r\mathbf{Z}$

$P_B = [b]P$

$P_A = [a]P$

# Three-Party Key Agreement

- Extension to three participants needs two communication rounds

1st round

$[c]P_B = [bc]P$

$[c]P_A = [ac]P$

**Charlie**

Secret $c \in \mathbf{Z}/r\mathbf{Z}$

$P_C = [c]P$

$P_A = [a]P$ $\quad P_B = [b]P$

$P_A$

$P_B$

$P_B$

**Alice**

Secret $a \in \mathbf{Z}/r\mathbf{Z}$

$P_A = [a]P$

$P_B = [b]P$

$[a]P_B = [ab]P$

**Bob**

Secret $b \in \mathbf{Z}/r\mathbf{Z}$

$P_B = [b]P$

# Three-Party Key Agreement

- Extension to three participants needs two communication rounds

2nd round

$[c]P_B = [bc]P$

$[c]P_A = [ac]P$

**Charlie**

Secret $c \in \mathbf{Z}/r\mathbf{Z}$

$P_C = [c]P$

$P_A = [a]P$ $\quad P_B = [b]P$

$[ab]P$

$[bc]P$

$[ac]P$

**Alice**

Secret $a \in \mathbf{Z}/r\mathbf{Z}$

$P_A = [a]P$

$P_B = [b]P$

$[a]P_B = [ab]P$

$$\boxed{\begin{aligned} K = [abc]P &= [a]([bc]P) \\ &= [c]([ab]P) = [b]([ac]P) \end{aligned}}$$

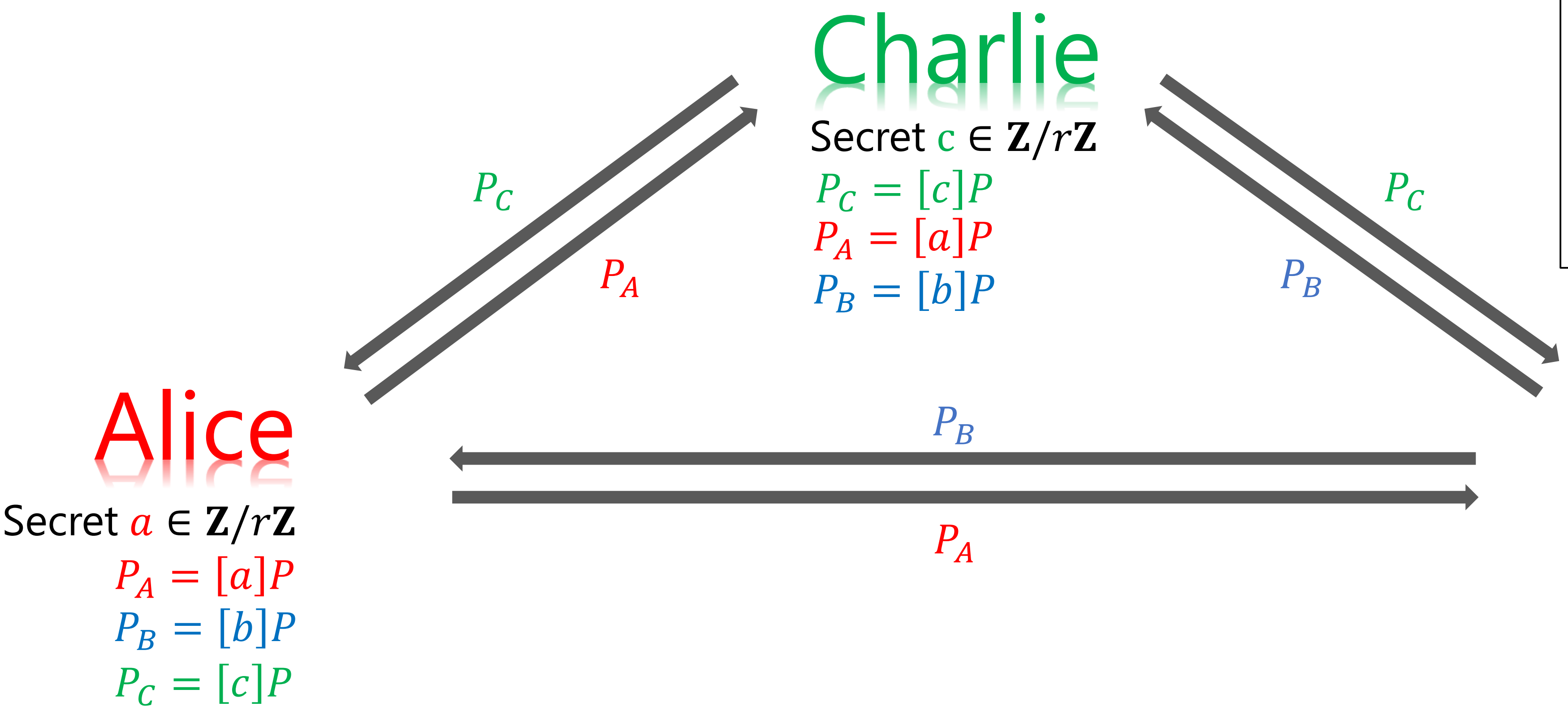
**Bob**

Secret $b \in \mathbf{Z}/r\mathbf{Z}$

$P_B = [b]P$

# Three-Party Key Agreement (Joux, 2000)

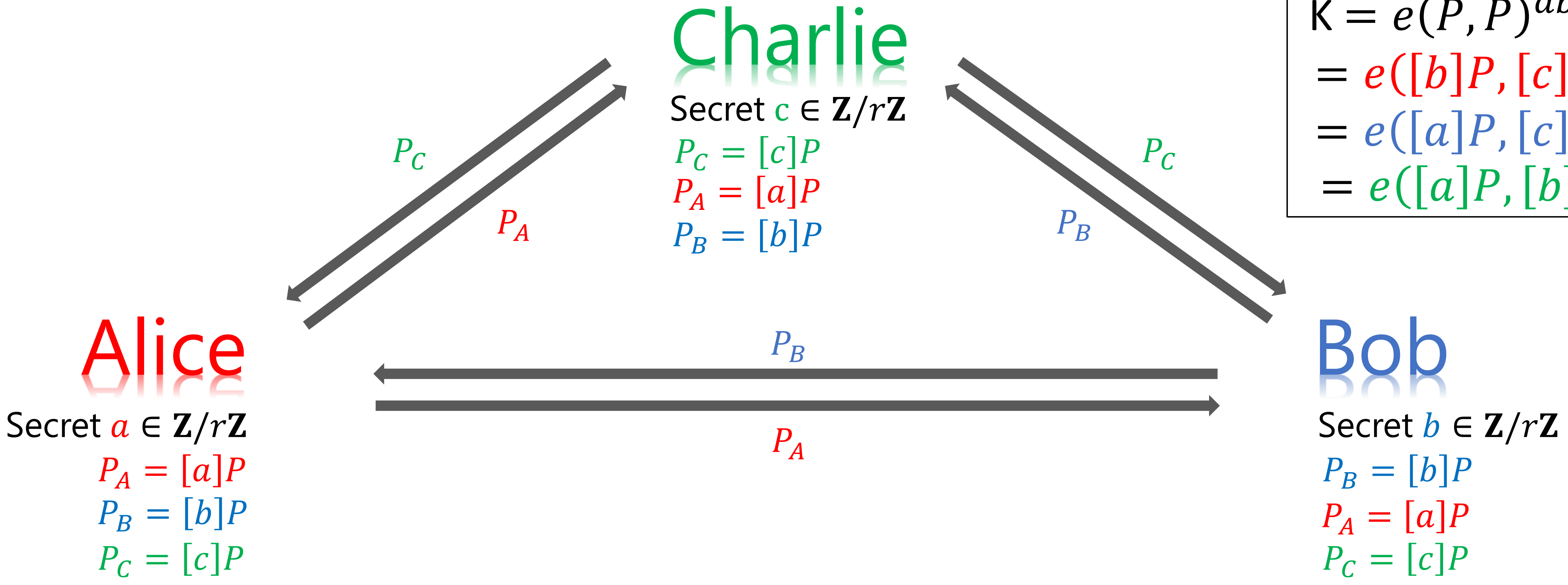
- If we have a bilinear map $e: G \times G \to G_3$ ,
  where $(G_3, \cdot)$ is a cyclic group of prime order, and $e(P, P) \neq 1$:




$$
\begin{aligned}
\mathrm{K} &= e(P, P)^{abc} \\
&= e([b]P, [c]P)^{a} \\
&= e([a]P, [c]P)^{b} \\
&= e([a]P, [b]P)^{c}
\end{aligned}
$$

# Three-Party Key Agreement (Joux, 2000)

- Bilinear Diffie-Hellman (BDH) problem:
  Given P, $[a]P, [b]P, [c]P \in G$, find $e(P,P)^{abc}$.
- BDHP must be computationally infeasible.




$$\boxed{\begin{aligned} \mathrm{K} &= e(P,P)^{abc} \\ &= e([b]P,[c]P)^{a} \\ &= e([a]P,[c]P)^{b} \\ &= e([a]P,[b]P)^{c} \end{aligned}}$$

# Three-Party Key Agreement (Joux, 2000)

- If we have a bilinear map $e: G_1 \times G_2 \to G_3$ ,
  where $(G_3,\cdot)$ is a cyclic group of prime order, and $e(P,Q) \neq 1$:

$$\begin{aligned} \mathrm{K} &= e(P,Q)^{abc} \\ &= e([b]P,[c]Q)^{a} \\ &= e([a]P,[c]Q)^{b} \\ &= e([a]P,[b]P)^{c} \end{aligned}$$

**Charlie**

Secret $c \in \mathbf{Z}/r\mathbf{Z}$

$Q_C = [c]Q$

$P_A = [a]P$

$Q_B = [b]Q$

$Q_C$ $P_A$ $Q_C$ $Q_B$

$P_B$

$P_A$

**Alice**

Secret $a \in \mathbf{Z}/r\mathbf{Z}$

$P_A = [a]P$

$P_B = [b]P$

$Q_C = [c]Q$

**Bob**

Secret $b \in \mathbf{Z}/r\mathbf{Z}$

$P_B = [b]P, Q_B = [b]Q$

$P_A = [a]P$

$Q_C = [c]Q$

# Short Digital Signatures (Boneh-Lynn-Shacham, 2000)

- System parameters: a pairing $e: G_1 \times G_2 \to G_3$ , $P \in G_1, Q \in G_2$, and a cryptographic hash function $H: \{0,1\}^* \to G_1$
- Alice's private key: $x_A \in \mathbf{Z}/r\mathbf{Z}$, public key: $Q_A = [x_A]Q \in G_2$
- Signature of message $M \in \{0,1\}^*$: $\sigma = [x_A]H(M) \in G_1$
- Verification: check whether $e(\sigma, Q) = e(H(M), Q_A)$
- Correctness:

$$e(\sigma, Q) = e([x_A]H(M), Q) = e(H(M), [x_A]Q) = e(H(M), Q_A)$$

- Only half the size of (EC)DSA signatures for same security

# Many More Interesting Applications…

- Non-interactive key agreement (Sakai-Ohgishi-Kasahara, 2000)
- Identity-based encryption (Boneh-Franklin, 2001)
- Attribute-based encryption (Sahai-Waters, 2004)
- Non-interactive zero-knowledge proofs (Groth-Sahai, 2008)
- Anonymous credentials (Belenkiy et al., 2009)
- Verifiable computation (Gentry-Howell-Parno-Raykova, 2013)

# Realizing Cryptographic Pairings

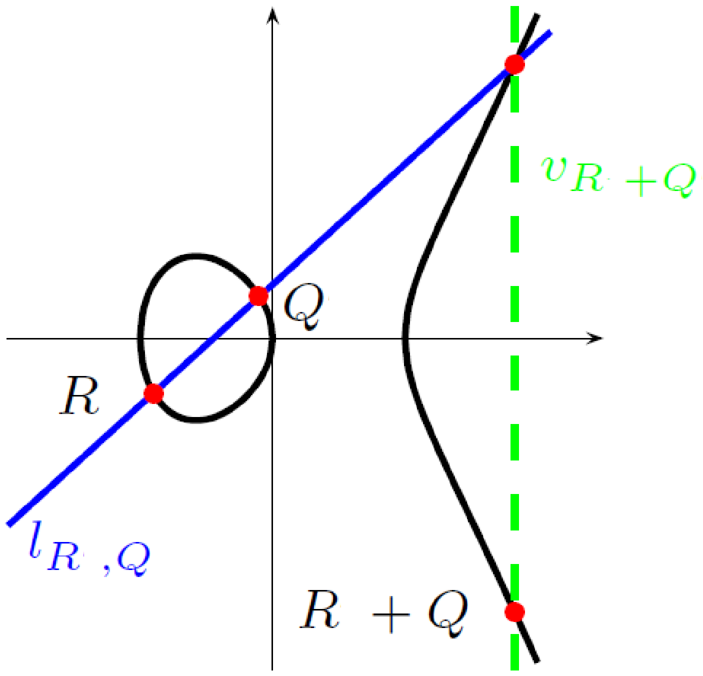
- Need quite large groups $G_1, G_2, G_3$
  s.t. solving DLP in all groups is computationally infeasible
- Need a pairing $e\colon\ G_1 \times G_2 \to G_3$
- Efficiency: need fast exponentiations in $G_1, G_2, G_3$
  and fast algorithm to compute the pairing
- There are different notions of practicality

**Need security and good performance!**
**Slow crypto will not be used!**

# Elliptic Curves over Finite Fields

- …have been used to provide groups for DL-based systems before (proposed by Miller and Koblitz in 1985, standardized for use in real-world applications)
- …have algorithms for efficient exponentiations in these groups
- …have undergone extensive cryptanalysis to build confidence in their security
- …have a pairing that maps two points to a finite field element

# Elliptic Curves over Finite Fields




- $\mathbf{F}_q$ finite field, $E$ an elliptic curve over $\mathbf{F}_q$
- If $\mathrm{char}(q) \notin \{2,3\}, E: y^2 = x^3 + ax + b$, $a, b \in \mathbf{F}_q$
- $E(\mathbf{F}_q) = \{(x,y) \in \mathbf{F}_q^2 : y^2 = x^3 + ax + b\} \cup \{\infty\}$ is an Abelian group with neutral element $\infty$
- $n = \#E(\mathbf{F}_q) = q + 1 - t, |t| \leq 2\sqrt{q}$
- Choose field and curve parameters s.t. $n = \#E(\mathbf{F}_q)$ has a large prime divisor $r$, use the group $G = \langle P \rangle$, where $\mathrm{ord}(P) = r$ and s.t. solving DLP is infeasible

# The Tate Pairing

$E/\mathbf{F}_q$ elliptic curve, $r$ a prime divisor of $n = \#E(\mathbf{F}_q)$

Embedding degree: smallest integer $k$ such that $r \mid q^k - 1$

For $k > 1$, $r$-torsion group $\mathrm{E[r]} \subset E\left(\mathbf{F}_{q^k}\right)$

- $G_1 = \langle P \rangle = E\left(\mathbf{F}_q\right)[r]$, $G_2 = \langle Q \rangle = E\left(\mathbf{F}_{q^k}\right)[r]$, $\infty \neq P, Q \notin E(\mathbf{F}_q)$
- $G_3 = \mu_r \subset \mathbf{F}_{q^k}^*$, group of $r$-th roots of unity

$$\boxed{t_r: G_1 \times G_2 \to G_3, (P, Q) \mapsto f_{r,P}(Q)^{(q^k-1)/r}}$$

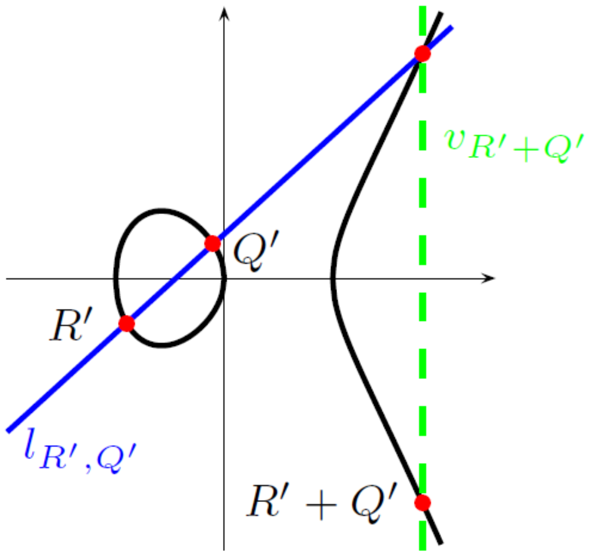
# Optimal Pairings

In practice, compute variants of the Tate pairing:

- $E/\mathbf{F}_q$ elliptic curve, $r$ a prime divisor of $n = \#E(\mathbf{F}_q)$, $k$ even
- Use a twist $E'$ of $E$: $\psi: E' \to E$ twisting isomorphism over $\mathbf{F}_{q^k}$
  $G_2' = \langle Q' \rangle = E'(\mathbf{F}_{q^e})[r]$, $\infty \neq Q'$, where $\psi(Q') = Q$, $e \in \{\frac{k}{2}, \frac{k}{4}, \frac{k}{6}\}$
  (depending on $j(E)$)
- Replace function $f_{r,P}(Q)$ by $g_{m,Q'}$(P) of smaller degree
  (for a suitable $m \in \mathbf{Z}$)

$$\boxed{a_{\text{opt}}: G_2' \times G_1 \to G_3, (Q', P) \mapsto g_{m,Q'}(P)^{(p^k-1)/r}}$$

# Components of Miller's Algorithm



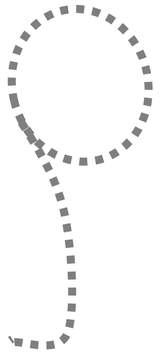

- Build function $g_{m,Q'}(P)$ iteratively in Miller loop from DBL/ADD steps (while computing $[m]Q'$)

| DBL | ADD | computation |
|---|---|---|
| $l_{R',R'}(P)$ | $l_{R',Q'}(P)$ | Coefficients in $\mathbf{F}_{q^e}$, evaluated at P ∈ $E(\mathbf{F}_q)$ |
| $R' \leftarrow [2]R'$ | $R' \leftarrow R' + Q'$ | Curve arithmetic in $E'(\mathbf{F}_{q^e})$ |
| $f \leftarrow f^2 \cdot l_{R',R'}(P)$ | $f \leftarrow f \cdot l_{R',Q'}(P)$ | General squaring, special mult. in $\mathbf{F}_{q^k}$ |

- Final exponentiation to the power $(q^k - 1)/r$ can use Frobenius automorphism and arithmetic in special subgroups of $\mathbf{F}_{q^k}^*$

# Minimal Requirements for Security

- Hardness of DLP measured by runtime of best known algorithms
- Security level of $\lambda$ bits: best algorithm needs $2^{\lambda}$ operations
- Elliptic Curve Groups: Pollard-$\rho$ (generic algorithm)
  random walk through group $G$ with $|G| = r$
  expected number of steps before collision occurs: $\approx \sqrt{r}$
  i.e. for 128 bits of security, group order must be around 256 bits



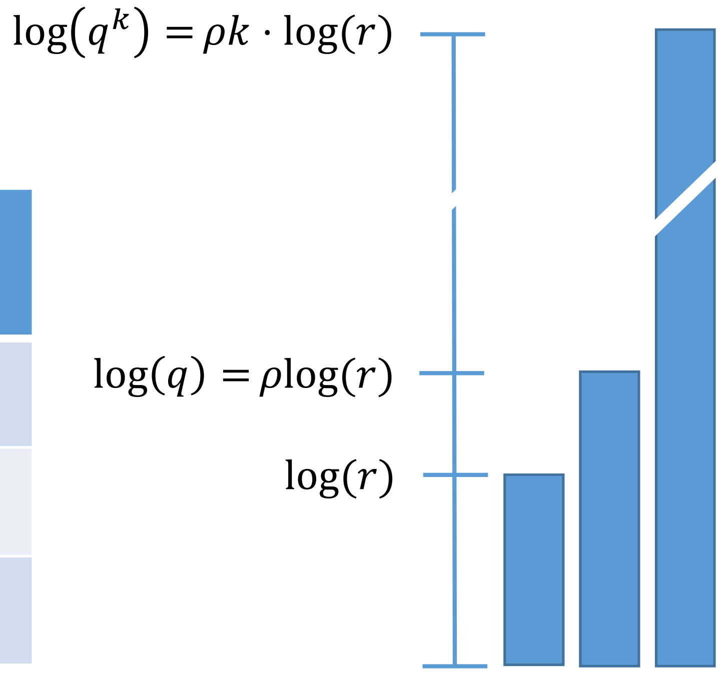
- Finite Field Group: Index Calculus algorithm (uses field structure)
  similar to factoring algorithms, uses a factor base of "small" elements,
  sub-exponential algorithm $\Rightarrow$ much larger field sizes required
- Recent work by Joux, significant improvement for binary field extensions
  lowering asymptotic complexity

# Minimal Requirements for Security

- Take $k$ as small as possible, but DLP must be infeasible in all groups
- $\rho = \log(q) / \log(r)$

| Security level (bits) | EC group order Size of $r$ (bits) | Extension field size Size of $q^k$ (bits) | Ratio $\rho \cdot k$ |
|---|---|---|---|
| 128 | 256 | 3072 | 12 |
| 192 | 384 | 7680 | 20 |
| 256 | 512 | 15360 | 30 |

NIST recommendations for key sizes (2012)

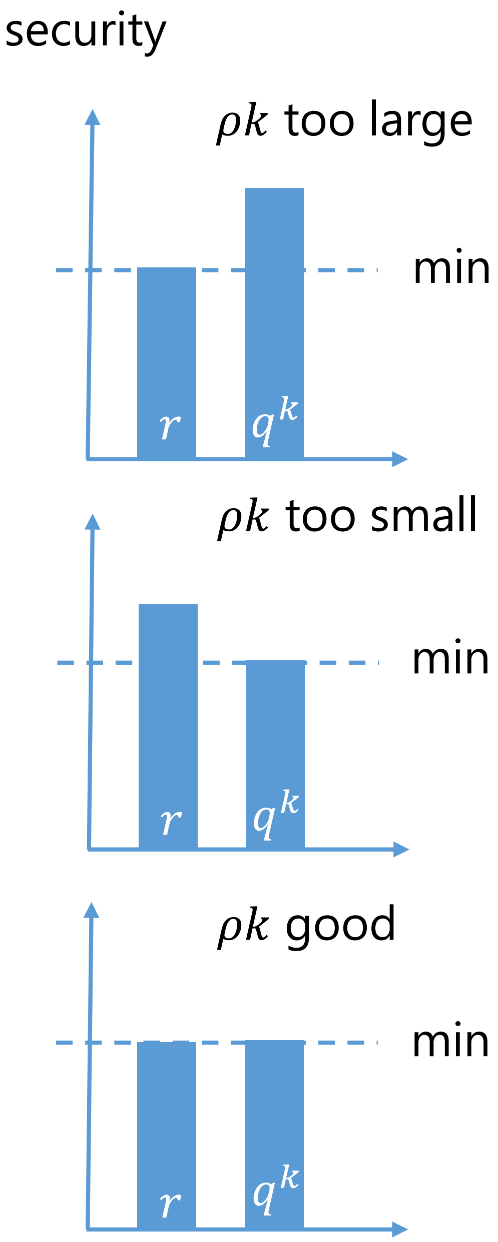


# Balanced Parameter Choice

- $\rho = \log(q) / \log(r)$, $\rho k \cdot \log(r) = \log q^k$
- If $\rho$ is too large, $q$ is larger than necessary.
- If $\rho k$ is too large, $q^k$ is larger than necessary.
- If $\rho k$ is too small, $r$ is larger than necessary.

| Security level (bits) | EC group order Size of $r$ (bits) | Extension field size Size of $q^k$ (bits) | Ratio $\rho \cdot k$ |
|---|---|---|---|
| 128 | 256 | 3072 | 12 |
| 192 | 384 | 7680 | 20 |
| 256 | 512 | 15360 | 30 |

NIST recommendations for key sizes (2012)



# Supersingular Elliptic Curves

Pairings on supersingular elliptic curves are efficient (Menezes-Okamoto-Vanstone, 1993 and Frey-Rueck, 1994)

- $k \leq 6$ (only suitable for low security)
- If $\mathrm{char}(\mathbf{F}_q) > 3$, then $k \leq 2$

Reducing discrete logarithms via pairings:
For $P \in G_1$ there exists $Q \in G_2$ with $e(P, Q) \neq 1$

- The map $G_1 \to G_3, P \mapsto e(P, Q)$ is a group isomorphism
- Solve DLP $P_A = [a]P$ in $G_1$ by solving DLP $g_A = e(P, Q)^a$ in $G_3$

# Pairing-Friendly Curves

The embedding degree of an ordinary elliptic curve is large in general. ($k$ is the order of $q \bmod r$)

- No chance of finding small $k$ by random search.

Find primes $p, r$ and an integer $n$ as follows

- $n = p + 1 - t$, $|t| \leq 2\sqrt{p}$, $t \neq 0$
- $r \mid n$
- $r \mid p^k - 1$ for small $k$ or $r \mid \Phi_k(p)$ ($k$-th cyclotomic polynomial)
- $t^2 - 4p = Dv^2 < 0$, $|D|$ small enough to compute the Hilbert class polynomial in $\mathbf{Q}(\sqrt{D})$

# Polynomial Parameterizations

Best pairing-friendly curves come from polynomial families

- Parameterize $p, r, t$ by polynomials $p(x), r(x), t(x) \in \mathbf{Q}[x]$ that satisfy the above conditions
- Define rho value for a family $\rho = \deg(p)\,/\deg(r)$
- Look at factorization of $\Phi_k(p(x))$ or $\Phi_k(t(x) - 1)$ for low-degree candidates for $p(x)$ or $t(x)$ of the right degree
- Take $r(x)$ to be one of the factors
- Hope for the CM equation to be nice

# Example

$k = 12$ $\longrightarrow$ $\Phi_{12}(x) = x^4 - x^2 + 1$

$t(x) = 6x^2 + 1$

$\Phi_{12}(t(x) - 1) = \Phi_{12}(6x^2) = n(x)n(-x),$
where $n(x) = 36x^4 + 36x^3 + 18x^2 + 6x + 1$

$p(x) = n(x) + t(x) - 1 = 36x^4 + 36x^3 + 24x^2 + 6x + 1$

Set $r(x) = n(x)$

$t(x)^2 - 4p(x) = -3(6x^2 + 4x + 1)^2$

$\rho = 1$

$D = -3$

$j(E) = 0$ $\longrightarrow$ $E: y^2 = x^3 + b$

# Families of Pairing-Friendly Curves

All examples below have $j(E) = 0$,

- $e = k/6$ (minimal fields for twist group $G_2'$)
- $E: y^2 = x^3 + b$

| $\lambda$ | Family | $k$ | $p(x)$ | $r(x)$ | $t(x)$ |
|---|---|---|---|---|---|
| 128 | BN<br>(Barreto-N., 2005) | 12 | $36x^4 + 36x^3 + 24x^2 + 6x + 1$ | $36x^4 + 36x^3 + 18x^2 + 6x + 1$ | $6x^2 + 1$ |
| 192 | BLS<br>(Barreto-Lynn-Scott, 2002) | 12 | $(x-1)^2(x^4 - x^2 + 1)/3 + x$ | $x^4 - x^2 + 1$ | $x + 1$ |
| 192 | KSS<br>(Kachisa-Schaefer-Scott, 2008) | 18 | $(x^8 + 5x^7 + 7x^6 + 37x^5 + 188x^4$<br>$+ 259x^3 + 343x^2 + 1763x + 2401)/21$ | $(x^4 + 16x^2 + 7)/7$ | $(x^6 + 37x^3$<br>$+ 343)/7^3$ |
| 256 | BLS<br>(Barreto-Lynn-Scott, 2002) | 24 | $(x-1)^2(x^8 - x^4 + 1)/3 + x$ | $x^8 - x^4 + 1$ | $x + 1$ |

# Families of Pairing-Friendly Curves

To find specific curves, search for an integer $u$ such that

- $p(u), r(u)$ are both prime
- Try different $b$ until $E: y^2 = x^3 + b$ has a point of order $r$

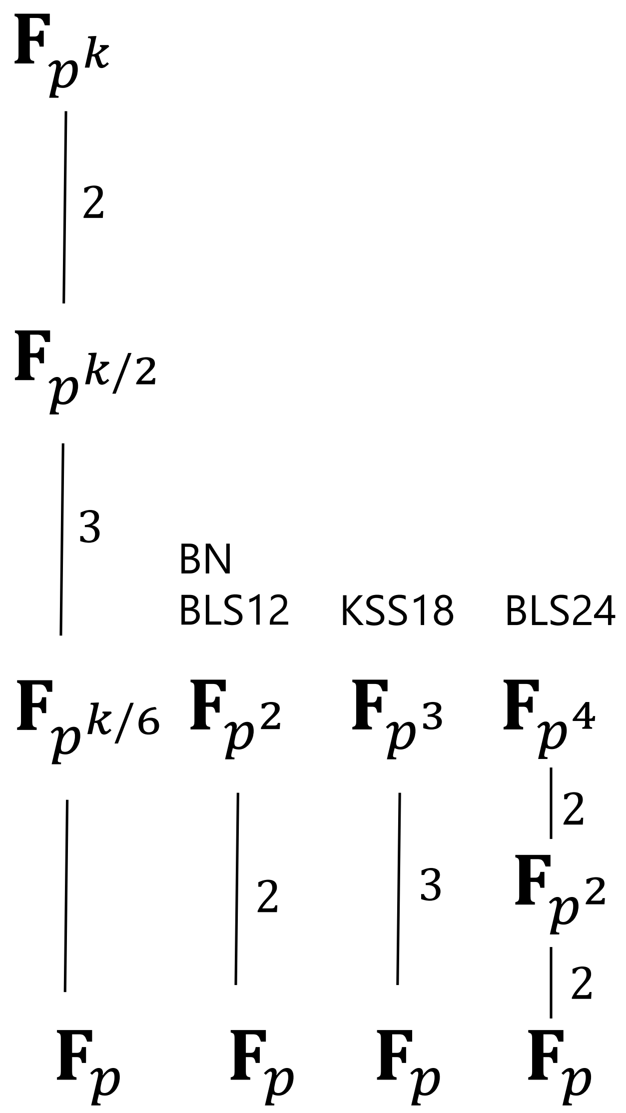
| $\lambda$ | Family | $k$ | $\rho$ | $\rho k$ | $\log(r)$ | $\log(p)$ | $u$ |
|---|---|---|---|---|---|---|---|
| 128 | BN (Barreto-N., 2005) | 12 | 1 | 12 | 254 | 254 | $-(2^{62} + 2^{55} + 1)$ |
| 192 | BLS (Barreto-Lynn-Scott, 2002) | 12 | 1.25 | 15 | 424 | 635 | $2^{106} - 2^{72} + 2^{69} - 1$ |
| 192 | KSS (Kachisa-Schaefer-Scott, 2008) | 18 | 1.33 | 24 | 376 | 508 | $2^{64} - 2^{51} + 2^{47} + 2^{28}$ |
| 256 | BLS (Barreto-Lynn-Scott, 2002) | 24 | 1.25 | 30 | 504 | 629 | $2^{63} - 2^{47} + 2^{38}$ |

# Field Extensions

- Construct degree-6 extension as
  $\mathbf{F}_{p^k} = \mathbf{F}_{p^{k/6}}(z),\ z^6 = \xi$
  $\mathbf{F}_{p^{k/2}} = \mathbf{F}_{p^{k/6}}(v),\ v^3 = \xi$
- Use monomials with small constants for all field extensions
- $p \equiv 3 \bmod 4$: $\mathbf{F}_{p^2} = \mathbf{F}_p(i),\ i^2 = -1$

$$(\alpha_0 + i\alpha_1) \cdot (\beta_0 + i\beta_1)$$
$$= (\alpha_0 \cdot \beta_0 - \alpha_1 \cdot \beta_1) + i(\alpha_0 \cdot \beta_1 + \alpha_1 \cdot \beta_0)$$

- Karatsuba multiplication (only 3 mults)
  $\alpha_0\beta_1 + \alpha_1\beta_0 = (\alpha_0 + \alpha_1)(\beta_0 + \beta_1) - \alpha_0\beta_0 - \alpha_1\beta_1$



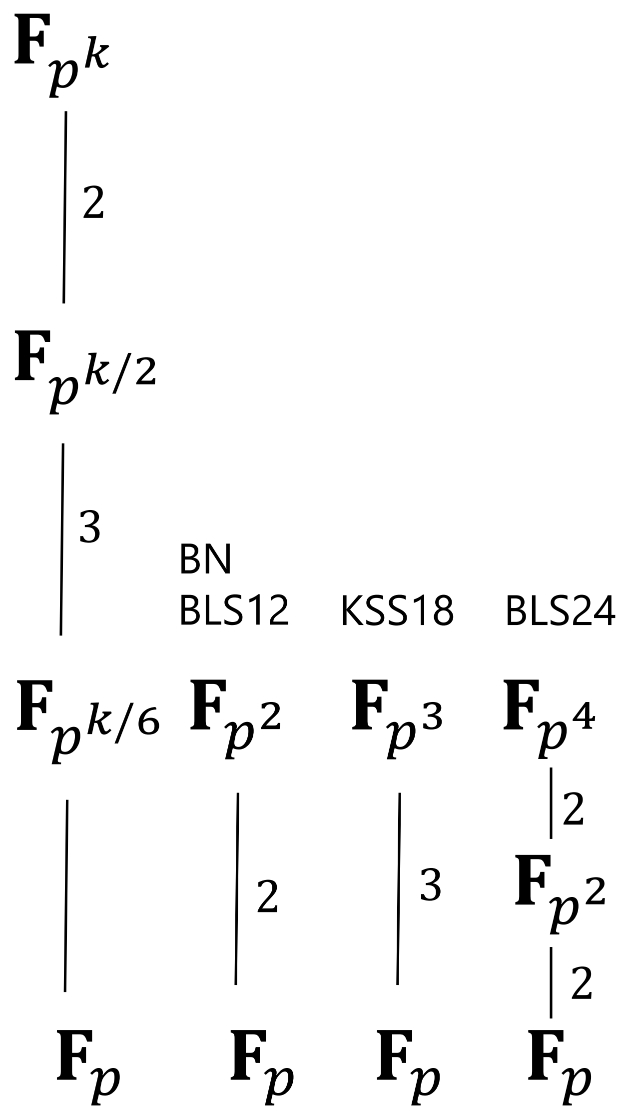
# Field Extensions

Lazy reduction:

- Choose $p$ of size a few bits smaller than multiple of machine word size (e.g. 64)
- Separate modular multiplication from modular reduction and postpone reduction until after following additions/subtractions
- Example: Do not reduce (3 reductions)

$$\alpha_0\beta_0, \alpha_1\beta_1, (\alpha_0 + \alpha_1)(\beta_0 + \beta_1)$$

- instead keep double precision for adds/subs and reduce (2 reductions)
  $(\alpha_0\beta_0 - \alpha_1\beta_1)$, $(\alpha_0 + \alpha_1)(\beta_0 + \beta_1) - \alpha_0\beta_0 - \alpha_1\beta_1$
- Carry up in the tower



# The Final Exponentiation

Exponent $\quad c = \frac{p^k-1}{r}, \log(c) \approx (k-1)\log(p)$

Assume $k$ even: $\quad c = (p^{k/2}-1)\,\frac{p^{k/2}+1}{r}$

$k = 12$: $\quad \frac{p^{12}-1}{r} = (p^6-1)(p^2+1)\,\frac{p^4-p^2+1}{r}$

- Use Frobenius: $f^c = [(f^{p^6}f^{-1})^{p^2}(f^{p^6}f^{-1})]^{\frac{p^4-p^2+1}{r}}$

- $\frac{p^4-p^2+1}{r} = \lambda_3 p^3 + \lambda_2 p^2 + \lambda_1 p + \lambda_0$, $|\lambda_i| < p$, $\lambda_i = \lambda_i(u), \deg\big(\lambda_i(x)\big) \leq 3$
  This part can be done with 3 exponentiations by $u$, some Frobenius applications and some multiplications and squarings
- Note: After exp by $(p^6-1)$, elts have norm 1, i.e. $f^{-1} = f^{p^6} = \bar{f}$

# The Final Exponentiation

- Actual exponentation work: 3 exponentiations by $u$, $\approx 3\log(p)$ instead of $\approx 11\log(p)$
- Usually, $u$ can be chosen very sparse, i.e. exponentiation is almost only squarings
- After exp by $(p^6 - 1)(p^2 + 1)$, result is in cyclotomic subgroup of $\mathbf{F}_{p^k}^*$, i.e. these squarings cost only $\approx 50\%$ of the original squarings
- Still, this exponentiation is more than half the cost of a pairing

# Exponentiations in Pairing Groups

Often protocols use only few pairings, but many exponentiations in $G_1$ and/or $G_2'$

- Important to speed up those as much as possible
- Use endomorphisms in curve groups (GLV/GLS methods and precomputations)
- Endomorphisms give certain multiples of curve points for free
  Example: $E/\mathbf{F}_p: y^2 = x^3 + b, p \equiv 1 \bmod 3$,
  has endomorph. $\phi: (x, y) \mapsto (\zeta x, y), \zeta^3 = 1, \zeta \neq 1$ and
  $\phi(P) = [\lambda]P$ for some $\lambda \in \mathbf{Z}/r\mathbf{Z}$, $\lambda^2 + \lambda + 1 \equiv 0 \bmod r$

# Efficiency of Pairings

- Ten years ago pairings were considered too slow for practical use
- At 128-bit security, efficiency gain of factor 50 (within last 6 years)
  Current speed record is <0.5ms per pairing on AMD Phenom II
  Within factor 10 of cost for exponentiations in curve groups
- Careful parameter choice is important

Pairings are efficient!

Thank you!

mnaehrig@microsoft.com