Pairing-based Cryptography and Its Applications

Rong-Jaye Chen

Department of Computer Science,
National Chiao Tung University, Taiwan

Outline

[1] Elliptic Curve Cryptograph (ECC)

1. Elliptic Curve
2. Elliptic Curve DLP

[2] Pairing-based Cryptography (PBC)

1. Pairings
2. Cryptography from Pairings

[3] Applications of PBC

1. ID-based Encryption
2. Searchable Encryption
3. Broadcast Encryption

Elliptic Curve Cryptography (ECC)

1. Elliptic Curves

- **Over Fields of Characteristic p>3**
  - Curve form

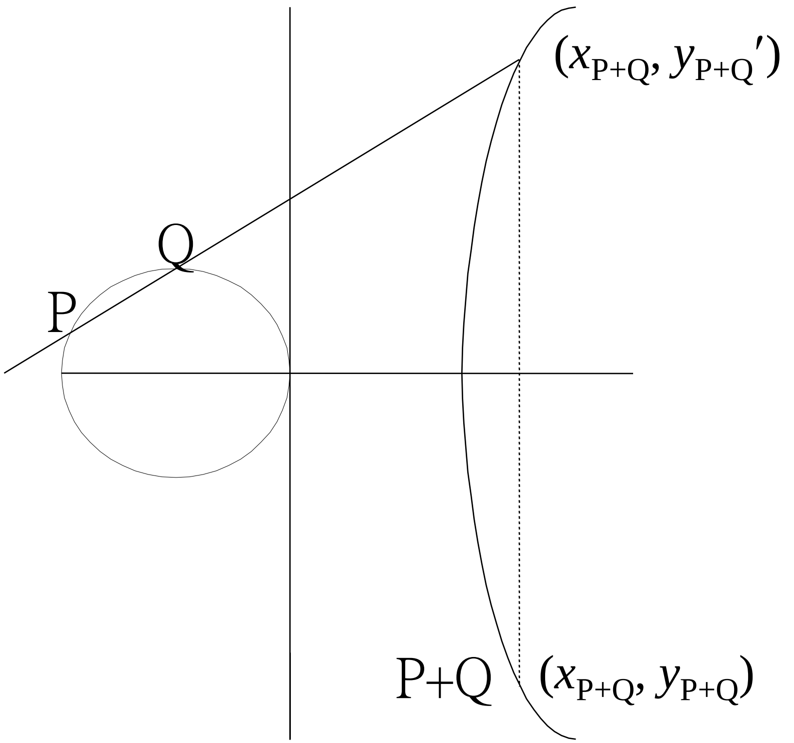
    $E: Y^2 = X^3 + aX + b$

    where $a, b \in F_q$, $q = p^n$

    $4a^3+27b^2 \neq 0$
  - Group operation

    given $P_1(x_1,y_1)$ and $P_2(x_2,y_2)$

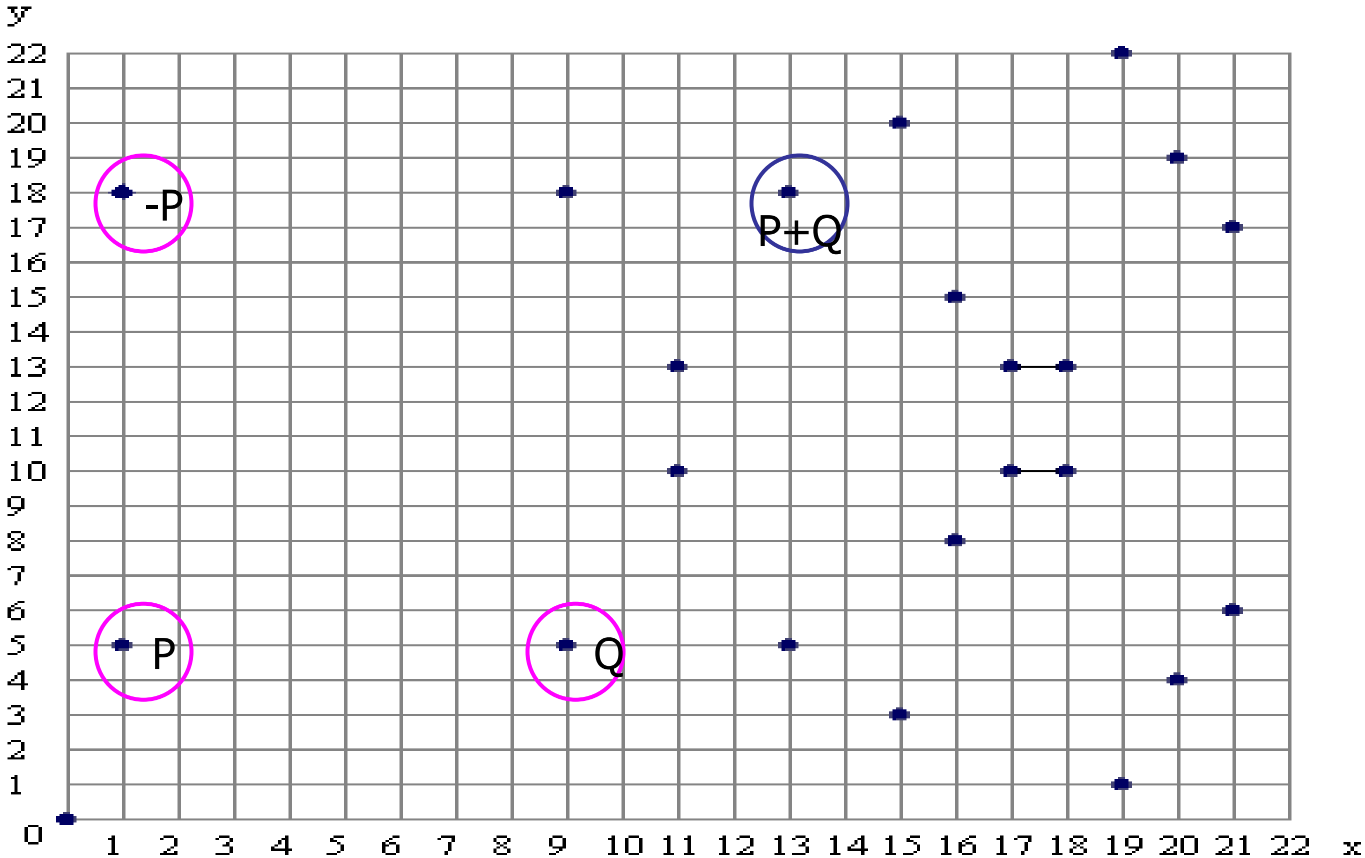
    compute $P_3(x_3,y_3) = P_1+P_2$



Example of EC over GF(p)

- **Example:** $p = 23, a = 1, b = 0$

$$E_{a,b}(Z_{23}) = \{(x, y) \in Z_{23}^{\;2} : y^2 = x^3 + x\} \cup \{O\}$$



Elliptic curve equation: $y^2 = x^3 + x$ over F_{23}

Example of EC over GF(p)

- Addition ($P_1 \neq P_2$)

Computational Cost
I + 3 M

$$\lambda = \frac{y_2 - y_1}{x_2 - x_1}$$
$$x_3 = \lambda^2 - x_1 - x_2$$
$$y_3 = (x_1 - x_3)\lambda - x_3 - y_1$$

- Doubling ($P_1 = P_2$)

Computational Cost
I + 4 M

$$\lambda = \frac{3x_1^{\ 2} + a}{2y_1}$$
$$x_3 = \lambda^2 - 2x_1$$
$$y_3 = (x_1 - x_3)\lambda - x_3 - y_1$$

1. Elliptic Curves

- **Over Fields of Characteristic 2**
  - Curve form

    E: $Y^2 + XY = X^3 + aX^2 + b$

    where a, b $\in F_q$, $b \neq 0$, $q = 2^n$
  - Group operation

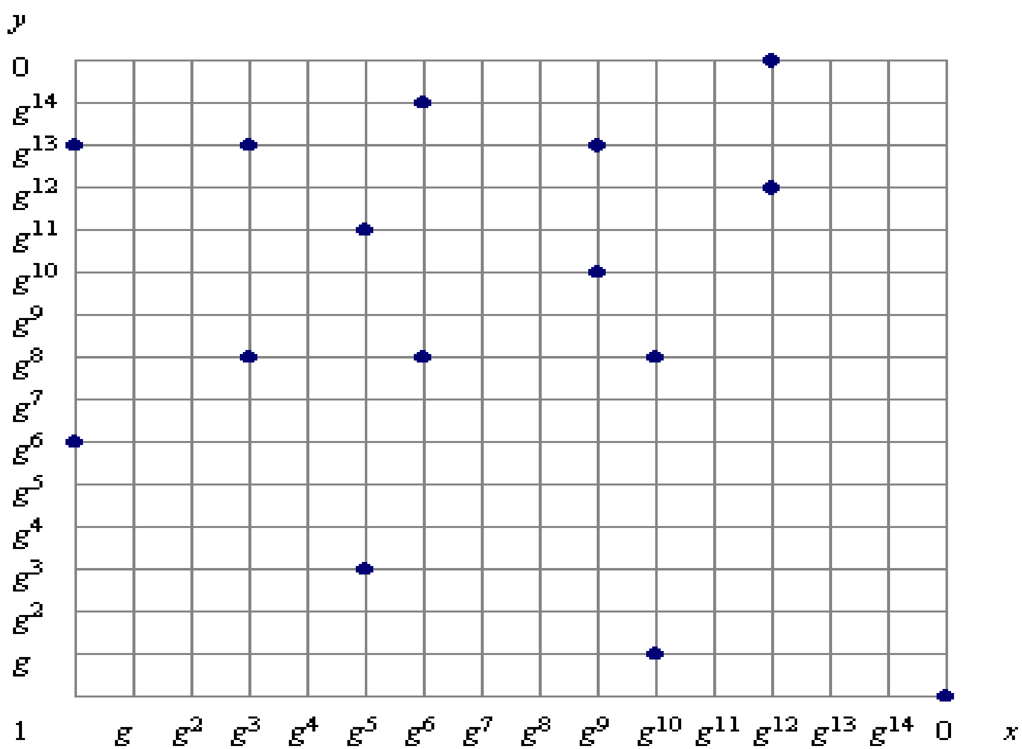
    given $P_1(x_1,y_1)$ and $P_2(x_2,y_2)$

    compute $P_3(x_3,y_3) = P_1+P_2$

Example of EC over GF(2^m)

$$GF(2^m) = Z_2[x]/\,p(x) \quad , p(x) = x^4 + x + 1$$

$$E:\ y^2 + xy = x^3 + g^4x^2 + 1$$



$y^2 + xy = x^3 + g^4x^2 + 1$ **over** F_{2^4}

$$g^4 = (0011)$$

$$1 = g^0 = (0001)$$

Example of EC over GF(2^m)

- Addition ($P_1 \neq P_2$)

Computational Cost
I + 2 M + S

$$\lambda = \frac{y_2 + y_1}{x_2 + x_1}$$

$$x_3 = \lambda^2 + \lambda + x_1 + x_2 + a$$

$$y_3 = (x_1 + x_3)\lambda + x_3 + y_1$$

- Doubling ($P_1 = P_2$)

Computational Cost
I + 2 M + S

$$\lambda = \frac{y_1}{x_1} + x_1$$

$$x_3 = \lambda^2 + \lambda + a$$

$$y_3 = (x_1 + x_3)\lambda + x_3 + y_1$$

2. Elliptic Curve DLP

- **Basic computation of ECC**
  - $Q = kP = \underbrace{P + P + \ldots + P}_{k\ times}$

    where P is a curve point, k is an integer

- **Strength of ECC**
  - Given curve, the point P, and kP

    It is hard to recover k

    - Elliptic Curve Discrete Logarithm Problem (ECDLP)

Elliptic Curve Security

Symmetric Key Size (bits)	RSA and Diffie-Hellman Key Size (bits)	Elliptic Curve Key Size (bits)	Years
80	1024	160	~2010
112	2048	224	~2030
128	3072	256	
192	7680	384	
256	15360	521	

NIST Recommended Key Sizes

Pairing-based Cryptography (PBC)

1. Pairings

- **Divisors**
  - Definition
  - Principal Divisors
- **Pairings**
  - Tate Pairings
  - Weil Pairings
  - More on Pairings

Definition of Divisors

- E/K , $P \in E(\overline{K})$, $[P]$: a formal symbol of P

  (1) Definition
  A divisor D on E is a finite linear combination of the formal symbols with integer coefficients:

$$D = \sum_j a_j[P_j], \quad a_j \in \mathbb{Z}$$

  (2) Definition
  $Div(E)$: group of divisors in (1)

  (3) Define degree and sum of a divisor

$$\deg\left(\sum_j a_j[P_j]\right) = \sum_j a_j \ \in \mathbb{Z}$$
$$sum\left(\sum_j a_j[P_j]\right) = \sum_j a_j P_j \ \in E(\overline{K})$$

Functions on E

- $E/K : y^2 = x^3 + Ax + B$

(1) Definition

A function on E is a rational function

$$f(x, y) \in \overline{K}(x, y)$$

that is defined for at least one point in $E(\overline{K})$. (e.g. rational function $1/(y^2 - x^3 - Ax - B)$ is not allowed.)

(2) Examples

$E : y^2 = x^3 - x$

$f(x, y) = x/y$ is defined at $(0, 0)$ on E !

$$\because \ \frac{x}{y} = \frac{y}{x^2 - 1} = 0 \quad \text{at } (0, 0)$$

(3) Definition

A function f has a zero at P if $f(P) = 0$

A function f has a pole at P if $f(P) = \infty$

Order of f at P

(1) Definition

For each P , $\exists$ a function u_P (a uniformizer at P) with $u_P(P) = 0$ and such that every function $f(x, y)$ can be written in

$$f = u_P^r g, \quad \text{with} \quad r \in \mathbb{Z} \text{ and } g(P) \neq 0, \infty$$

$$r \triangleq ord_P(f) \ : \ \text{order of } f \text{ at } P$$

(2) Example

$y^2 = x^3 - x, \quad u_{(0,0)}(x, y) = y$ a uniformizer at $(0, 0)$

$ord_{(0,0)}(x) = ?$

$$\because x = y^2 \frac{1}{x^2 - 1} \qquad \therefore ord_{(0,0)}(x) = 2$$

and $ord_{(0,0)}(x/y) = 1$

Principal Divisors (1/3)

(1) Definition

f is a function on E , $f \neq 0$

the divisor of f

$$div(f) \triangleq \sum_{P \in E(\overline{K})} ord_P(f)[P] \in Div(E)$$

(2) Proposition

$f \neq 0$ is a function on E . Then

1. f has only finitely many zeros and poles
2. $\deg(div(f)) = 0$
3. If f has no zero or pole (so $div(f) = 0$), then f is a constant.

(3) Definition

A divisor D is a principal divisor if it is the divisor of a function.

i.e. $D = div(f)$, for some f

Principal Divisors (2/3)

(4) Suppose P_1, P_2, P_3 are 3 points on E that lie on the line $ax + by + c = 0$

Then $f(x, y) = ax + by + c$ has zeros at P_1, P_2, P_3 . If $b \neq 0$ then f has a triple pole at ∞ .

Therefore

$$div(ax + by + c) = [P_1] + [P_2] + [P_3] - 3[\infty]$$

The line through $P_3 = (x_3, y_3)$ and $-P_3$ is $x - x_3 = 0$.

$$div(x - x_3) = [P_3] + [-P_3] - 2[\infty]$$

Principal Divisors (3/3)

(4) Therefore,

$$\begin{aligned} div\left(\frac{ax+by+c}{x-x_3}\right) &= div(ax+by+c) - div(x-x_3) \\ &= [P_1]+[P_2]-[-P_3]-[\infty] \end{aligned}$$

Since $P_1 + P_2 = -P_3$ on E . So

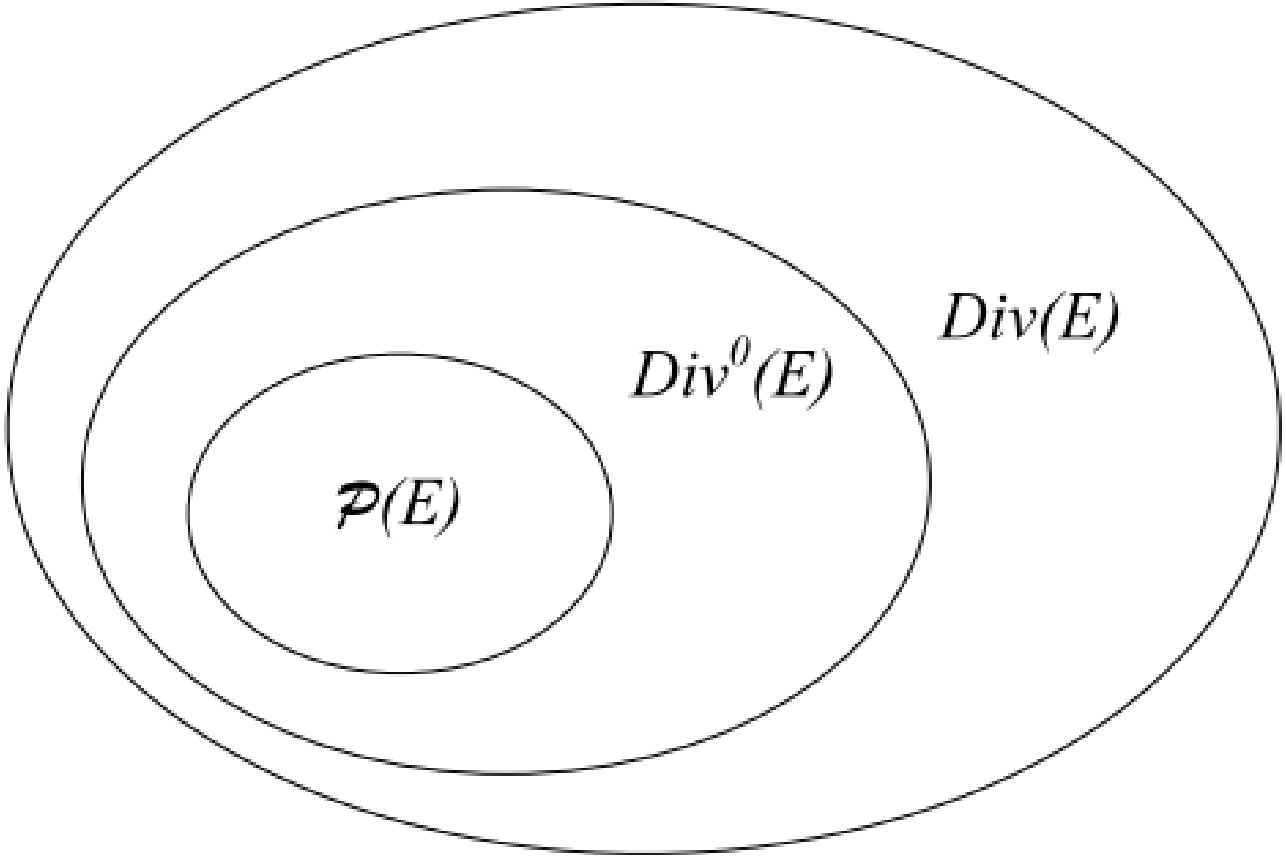
$$[P_1]+[P_2] = [P_1+P_2]+[\infty]+div\left(\frac{ax+by+c}{x-x_3}\right)$$

Theorem

D : *divisor on* E *with* $\deg(D)=0$

$\rightarrow\ \exists\ f$ *on* E *with* $div(f)=D$ *if and only if* $sum(D)=\infty$

Group Relation



$\mathcal{P}(E)$: principal divisors on E

Example (1/2)

- $E/F_{11} : y^2 = x^3 + 4x$,
  $D = [(0,\ 0)] + [(2,\ 4)] + [(4,\ 5)] + [(6,\ 3)] - 4[\infty]$
  $\because\ \deg(D) = 0, \quad sum(D) = \infty$
  By theorem, D is a divisor of a function.
  Let's find the function.

  (1) The line through $(0,\ 0)$, $(2,\ 4)$ is $y - 2x = 0$.
  It is tangent to E at $(2,\ 4)$, so $div(y - 2x) = [(0,\ 0)] + 2[(2,\ 4)] - 3[\infty]$

  (2) The vertical line through $(2,\ 4)$ is $x - 2 = 0$,
  $div(x - 2) = [(2,\ 4)] + [(2,\ -4)] - 2[\infty]$

$$\therefore\ D = [(2, -4)] + div\left(\frac{y - 2x}{x - 2}\right) + [(4,\ 5)] + [(6,\ 3)] - 3[\infty]$$

Example (2/2)

- (Continue):

  (3) Similarly,

$$[(4,\ 5)] + [(6,\ 3)] = [(2,\ 4)] + [\infty] + div\left(\frac{y+x+2}{x-2}\right)$$

$$\rightarrow\ D = [(2,\ -4)] + div\left(\frac{y-2x}{x-2}\right) + [(2,\ 4)] + div\left(\frac{y+x+2}{x-2}\right) - 2[\infty]$$

$$\rightarrow\ D = div(x-2) + div\left(\frac{y-2x}{x-2}\right) + div\left(\frac{y+x+2}{x-2}\right)$$

$$= div\left(\frac{(y-2x)(y+x+2)}{x-2}\right)$$

  (4) $(y-2x)(y+x+2) = y^2 - xy - 2x^2 + 2y - 4x$

$$= x^3 - xy - 2x^2 + 2y \quad \text{(Since } y^2 = x^3 + 4x \text{)}$$

$$= (x-2)(x^2 - y)$$

$$\therefore\ D = div(x^2 - y)$$

Pairings

- In the following slides, we use

$$[n]P \text{ for } nP$$

$$n(P) \text{ for } n[P]$$

$$(f) \text{ for } div(f)$$

Preliminaries (1/2)

- E is an elliptic curve defined over $\mathbb{F}_q$, whose characteristic is p.
- r is a large prime which divides $\#E(\mathbb{F}_q)$, where $\gcd(r, p) = 1$.
- $\mu_r = \{u \in \overline{\mathbb{F}}_q \mid u^r = 1\}$.
- The *embedding degree* k is the smallest positive integer such that $r \mid q^k - 1$.
- Then, $\mathbb{F}_{q^k} = \mathbb{F}_q(\mu_r)$.

Preliminaries (2/2)

- $(\mathbb{F}^*_{q^k})^r = \{u^r \mid u \in \mathbb{F}^*_{q^k}\}$.
  - $(\mathbb{F}^*_{q^k})^r$ is a subgroup of $\mathbb{F}^*_{q^k}$.
  - The group $\mathbb{F}^*_{q^k}/(\mathbb{F}^*_{q^k})^r$ is isomorphic to μ_r.
- $E(\mathbb{F}_{q^k})[r] = \{P \in E(\mathbb{F}_{q^k}) \mid [r]P = \infty\}$.
- $rE(\mathbb{F}_{q^k}) = \{[r]P \mid p \in E(\mathbb{F}_{q^k})\}$.
  - $rE(\mathbb{F}_{q^k})$ is a subgroup of $E(\mathbb{F}_{q^k})$.
  - $|E(\mathbb{F}_{q^k})[r]| = |E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})|$
  - In many cases of relevance for cryptography, one can represent $E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$ using the points of $E(\mathbb{F}_{q^k})[r]$.

Tate Pairing (1/2)

- Let f be a function and $D = \Sigma_P n_P(P)$ be a divisor, then

$$f(D) = \Pi_P f(P)^{n_P}.$$

- Let $P \in E(\mathbb{F}_{q^k})[r]$.
  - Since $[r]P = \infty$, there is a function f such that $(f) = r(P) - r(\infty)$.
- Let $Q \in E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$.
  - Construct a divisor $D = (Q+S) - (S)$ by choosing an arbitrary point $S \in E(\mathbb{F}_{q^k})$ such that the supports of (f) and D are disjoint.

Tate Pairing (2/2)

- The *Tate pairing*

$$\langle \cdot, \cdot \rangle_r : E(\mathbb{F}_{q^k})[r] \times E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k}) \rightarrow \mathbb{F}_{q^k}^* / (\mathbb{F}_{q^k}^*)^r$$

  is defined by

$$\langle P, Q \rangle_r = f(D).$$

- For practical purposes, the *reduced* Tate pairing unifies the result of the Tate pairing by

$$e(P, Q) = \langle P, Q \rangle_r^{(q^k-1)/r}$$

  which maps into the group $\mu_r \subset \mathbb{F}_{q^k}^*$.

- If $k > 1$ or $P \in rE(\mathbb{F}_q)$, then

$$e(P, P) = 1.$$

Properties of Tate Pairing

- **Bilinearity:** For all $P, P_1, P_2 \in E(\mathbb{F}_{q^k})[r]$ and $Q, Q_1, Q_2 \in E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$,

$$\langle P_1 + P_2, Q\rangle_r = \langle P_1, Q\rangle_r \langle P_2, Q\rangle_r$$

  and

$$\langle P, Q_1 + Q_2\rangle_r = \langle P, Q_1\rangle_r \langle P, Q_2\rangle_r.$$

- **Non-degeneracy:**
  - For all $P \in E(\mathbb{F}_{q^k})[r] \backslash \{\infty\}$, there is some $Q \in E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$ such that $\langle P, Q\rangle_r \neq 1$.
  - Similarly, for all $Q \in E(\mathbb{F}_{q^k})/rE(\mathbb{F}_{q^k})$ with $Q \notin rE(\mathbb{F}_{q^k})$, there is some $P \in E(\mathbb{F}_{q^k})[r]$ such that $\langle P, Q\rangle_r \neq 1$.

The Idea of Miller's Algorithm

- To compute the Tate pairing, we need to construct a function f such that $(f) = r(P) - r(\infty)$.
- Write f_i for a function such that

$$(f_i) = i(P) - ([i]P) - (i-1)(\infty).$$

  Note that $f_1 = 1$

- Let l be the straight line across $[i]P$ and $[j]P$, and v be the verticle line across $[i+j]P$, then

$$(l/v) = ([i]P) + ([j]P) - ([i+j]P) - (\infty).$$

- So,

$$f_{i+j} = f_i f_j \frac{l}{v}.$$

Weil Pairing

- E is an elliptic curve defined over $\mathbb{F}_q$, whose characteristic is p.
- r is a large prime which divides $\#E(\mathbb{F}_q)$, where $\gcd(r, p) = 1$.
- k' is the smallest positive integer such that $E[r] \subset E(\mathbb{F}_{q^{k'}})$.
- Let $P, Q \in E[r]$ and construct degree zero divisors $D = (P + S) - (S)$, $D' = (Q + T) - (T)$ such that the supports of D and D' are disjoint.
- Let $(f) = rD$, and $(g) = rD'$.
- The *Weil pairing* is a map

$$e_r : E[r] \times E[r] \to \mu_r \subseteq \mathbb{F}_{q^{k'}}$$

defined by

$$e_r(P, Q) = f(D')/g(D).$$

Properties of Weil Pairing

- **Bilinearity:** For all $P, P', Q, Q' \in E[r]$,

$$e_r(P + P', Q) = e_r(P, Q)e_r(P', Q)$$

  and

$$e_r(P, Q + Q') = e_r(P, Q)e_r(P, Q').$$

- **Non-degeneracy:** If $e_r(P, Q) = 1$ for all $Q \in E[r]$, then $P = \infty$.
- **Alternating:** $e_r(P, P) = 1$ and so $e_r(P, Q) = e_r(Q, P)^{-1}$.

Tate Pairing vs. Weil Pairing

- If $\mu_r \not\subseteq (\mathbb{F}^*_{q^{k'}})^r$, then

$$e_r(P, Q) = \frac{e(P, Q)}{e(Q, P)}.$$

- The Tate pairing requires working over $\mathbb{F}_{q^k}$ while the Weil pairing requires the potentially much larger field $\mathbb{F}_{q^{k'}}$.
  - If $r \nmid (q-1)$ and $\gcd(r, q) = 1$, then $k = k'$.

More on Pairings

- **Distortion Maps:**
  - Let $P \in E(\mathbb{F}_q)$ have prime order r, and suppose $k > 1$.
  - Suppose $E(\mathbb{F}_{q^k})$ has no points of order r^2.
  - Let ϕ be an endomorphism of E such that $\phi(P) \notin E(\mathbb{F}_q)$.

  1. $e(P, \phi(P)) \neq 1$.
  2. The endomorphism ϕ is called a distortion map.

- If an elliptic curve E has a distortion map, then E is supersingular.

- Use distortion maps, and restrict the pairing to a single cyclic subgroup.

  1. In this case, $Q = [m]P$.
  2. **Symmetry:**

$$e(Q, \phi(P)) = e([m]P, \phi(P)) = e(P, [m]\phi(P)) = e(P, \phi(Q)).$$

Distortion Maps

k	Elliptic curve data
2	$E:\ y^2 = x^3 + a$ over $\mathbb{F}_p$, where $p \equiv 2 \pmod 3$ $\#E(\mathbb{F}_p) = p + 1$ Distortion map $(x, y) \mapsto (\zeta_3 x, y)$, where $\zeta_3^3 = 1$.
2	$E:\ y^2 = x^3 + x$ over $\mathbb{F}_p$, where $p \equiv 3 \pmod 4$ $\#E(\mathbb{F}_p) = p + 1$ Distortion map $(x, y) \mapsto (-x, iy)$, where $i^2 = -1$.
3	$E:\ y^2 = x^3 + a$ over $\mathbb{F}_{p^2}$, where $p \equiv 5 \pmod 6$ and $a \in \mathbb{F}_{p^2}, a \notin \mathbb{F}_p$ is a square which is not a cube. $\#E(\mathbb{F}_{p^2}) = p^2 - p + 1$ Distortion map $(x, y) \mapsto \left(x^p/\left(\gamma a^{(p-2)/3}\right), y^p/a^{(p-1)/2}\right)$, where $\gamma \in \mathbb{F}_{p^6}$ satisfies $\gamma^3 = a$.
4	$E_i:\ y^2 + y = x^3 + x + a_i$ over $\mathbb{F}_2$, where $a_1 = 0$ and $a_2 = 1$ $\#E_i(\mathbb{F}_{2^l}) = 2^l \pm 2^{(l+1)/2} + 1$ (l odd) Distortion map $(x, y) \mapsto (u^2x + s^2, y + u^2sx + s)$, where $u \in \mathbb{F}_{2^2}$ and $s \in \mathbb{F}_{2^4}$ satisfy $u^2 + u + 1 = 0$ and $s^2 + (u + 1)s + 1 = 0$.
6	$E_i:\ y^2 = x^3 - x + a_i$ over $\mathbb{F}_3$, where $a_1 = 1$ and $a_2 = -1$ $\#E_i(\mathbb{F}_{3^l}) = 3^l \pm 3^{(l+1)/2} + 1$ (l odd) Distortion map $(x, y) \mapsto (\alpha - x, iy)$, where $i \in \mathbb{F}_{3^2}$ and $\alpha \in \mathbb{F}_{3^3}$ satisfy $i^2 = -1$ and $\alpha^3 - \alpha - a_i = 0$.

Modified Pairings

- $E(\mathbb{F}_q)$ is supersingular with $r \mid \#E(\mathbb{F}_q)$ for some prime r.
- ϕ is the distortion map of E.
- The embedding degree $k > 1$ and assume $E(\mathbb{F}_{q^k})$ has no points of order r^2.
- Put $G_1 = \langle P \rangle$, where $P \in E(\mathbb{F}_q) \backslash \{\infty\}$, and $G_3 = \mu_r$.
- **Modified Pairings:**
  1. $Q, R \in G_1$.
  2. The modified pairing
  $$\hat{e} : G_1 \times G_1 \to G_3$$
  is defined by
  $$\hat{e}(Q, R) = e(Q, \phi(R))$$
  3. It has *bilinearity*, *symmetry*, and *non-degeneracy*.

2. Cryptography from Pairings

- **Key Distribution Schemes**
  - Identity-based Non-interactive Key Distribution
  - Three-party Key Distribution
- **Signature Schemes**
  - Identity-based Signature
  - Short Signature

ID-based Non-interactive Key Distribution

- Sakai, Ohgishi, and Kasahara (SCIS 2000)
- **Setup & Extract:** The same as Identity-Based Encryption
  - The system parameters: $\langle G_1, G_3, \hat{e}, H, \rangle$.
  - User A: public key $Q_A = H(ID_A)$, and private key $S_A = [s]Q_A$.
  - User B: public key $Q_B = H(ID_B)$, and private key $S_B = [s]Q_B$.
- **Key Agreement:**
  - User A computes $\hat{e}(S_A, Q_B) = \hat{e}(Q_A, Q_B)^s$.
  - User B computes $\hat{e}(Q_A, S_B) = \hat{e}(Q_A, Q_B)^s$.

Three-party Key Distribution

- Joux (ANTS 2000)
- **Setup:**
  - The system parameters: $\langle G_1, G_3, \hat{e}, P \rangle$.
- **Key Agreement:**
  - User A select a number a, and broadcast $[a]P$.
  - User B select a number b, and broadcast $[b]P$.
  - User C select a number c, and broadcast $[c]P$.
  - User A computes $\hat{e}([b]P, [c]P)^a = \hat{e}(P, P)^{abc}$.
  - User B computes $\hat{e}([a]P, [c]P)^a = \hat{e}(P, P)^{abc}$.
  - User C computes $\hat{e}([a]P, [b]P)^a = \hat{e}(P, P)^{abc}$.

ID-based Signature

- Cha and Cheon (PKC 2003)
- **Setup & Extract:** The same as Identity-Based Encryption
  - The system parameters: $\langle G_1, G_3, \hat{e}, P, Q_0, H, H_2 \rangle$.

    where $H_2 : \{0,1\}^* \times G_1 \rightarrow \mathbb{Z}_r$.
- **Sign:** A want to sign the message M
  1. Choose a random $t \in \mathbb{Z}_r^*$.
  2. Compute $U = [t]Q_A$, $h = H_2(M, U)$, and $V = [t+h]S_A$.
  3. The signature $\sigma = \langle U, V \rangle$
- **Verify:** Someone verify the signed message $(M, \langle U, V \rangle)$.
  1. Compute $h = H_2(M, U)$, and $Q_A = H(ID_A)$.
  2. Check if $\hat{e}(Q_0, U + [h]Q_A) = \hat{e}(P, V)$.

Short Signature

- Boneh, Lynn, and Shacham (ASIACRYPT 2001)
- **Setup:**
  - The system parameters: $\langle G_1, G_3, \hat{e}, P \rangle$.
- **Extract:** The user A chooses his own private key.
  - Choose $x \in \mathbb{Z}_r^*$ as the private key, and compute the public key $Q_A = [x]P$
- **Sign:** A want to sign the message M
  1. The signature $\sigma = [x]H(M)$.
- **Verify:**
  1. Get the public key Q_A of A.
  2. Check if $\hat{e}(\sigma, P) = \hat{e}(H(M), Q_A)$.

Applications of PBC

1. ID-based Encryption

- **History**
- **Certificate-based Cryptography**
- **Identity-based Cryptography**

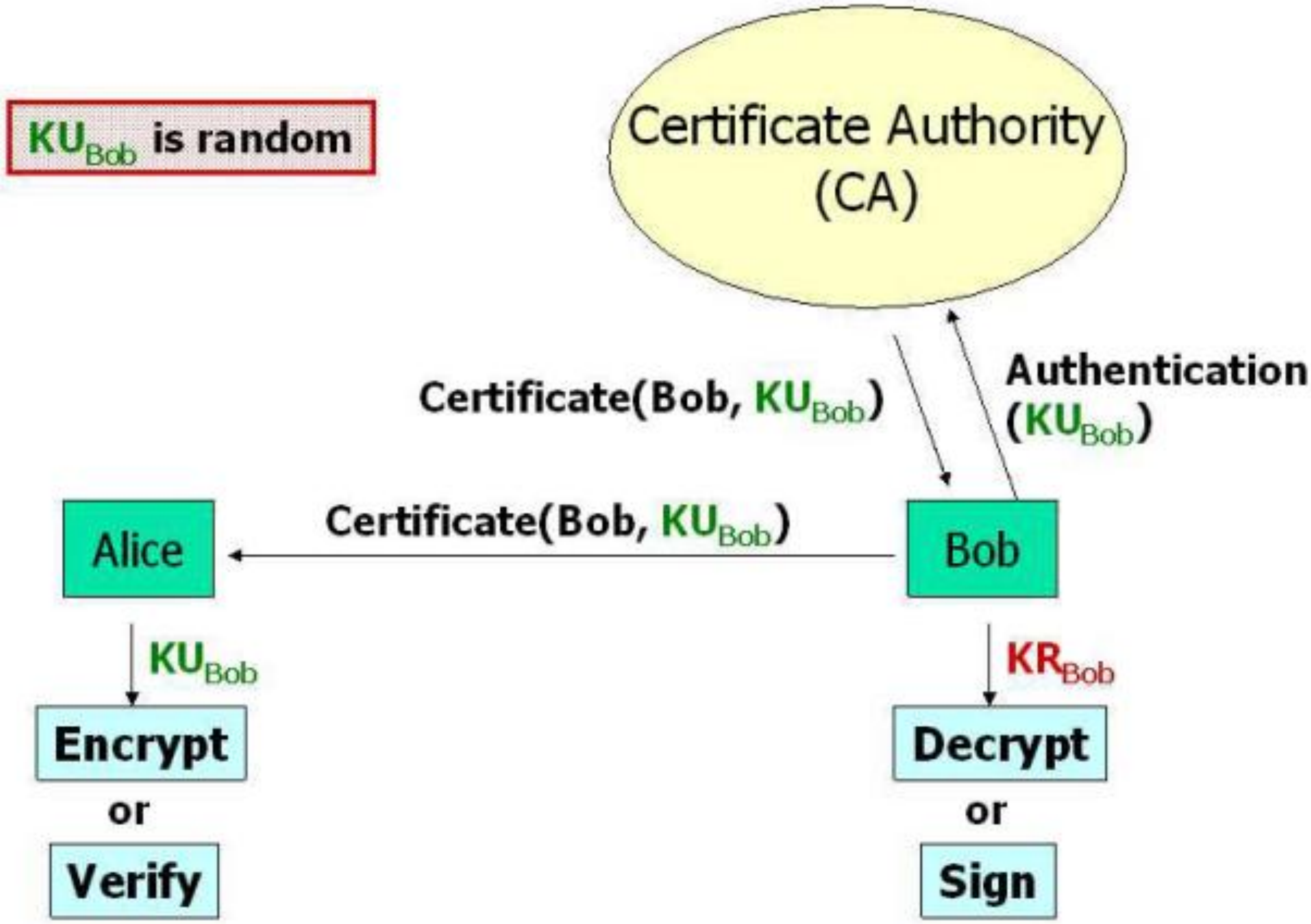
History

- **Shamir (CRYPTO 1984) raised the open problem.**

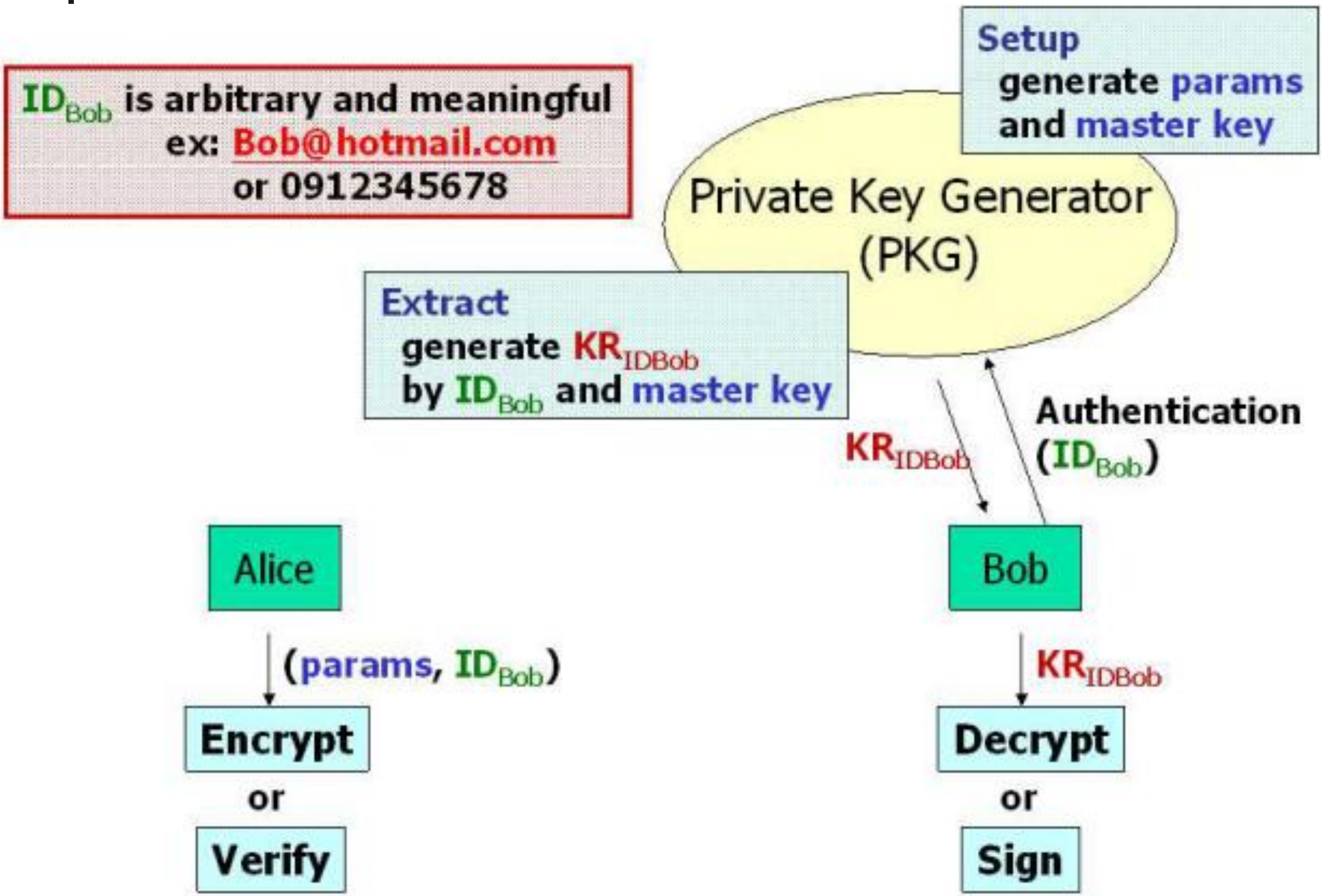
- **Two solutions:**
  - Pairing-based approach:
    Boneh and Franklin (CRYPTO 2001)

  - Based on the Quadratic Residuosity problem:
    Cocks (Crypto and Coding 2001)

Certificate-based Cryptography



Identity-based Cryptography



Protocol (1/2)

- **Setup:**
  - Common parameters: G_1, G_3, $\hat{e}$, and P.
  - PKG select a master key s, and keep in secret. The public parameter $Q_0 = P_{pub} = [s]P$.
  - Two hash functions:

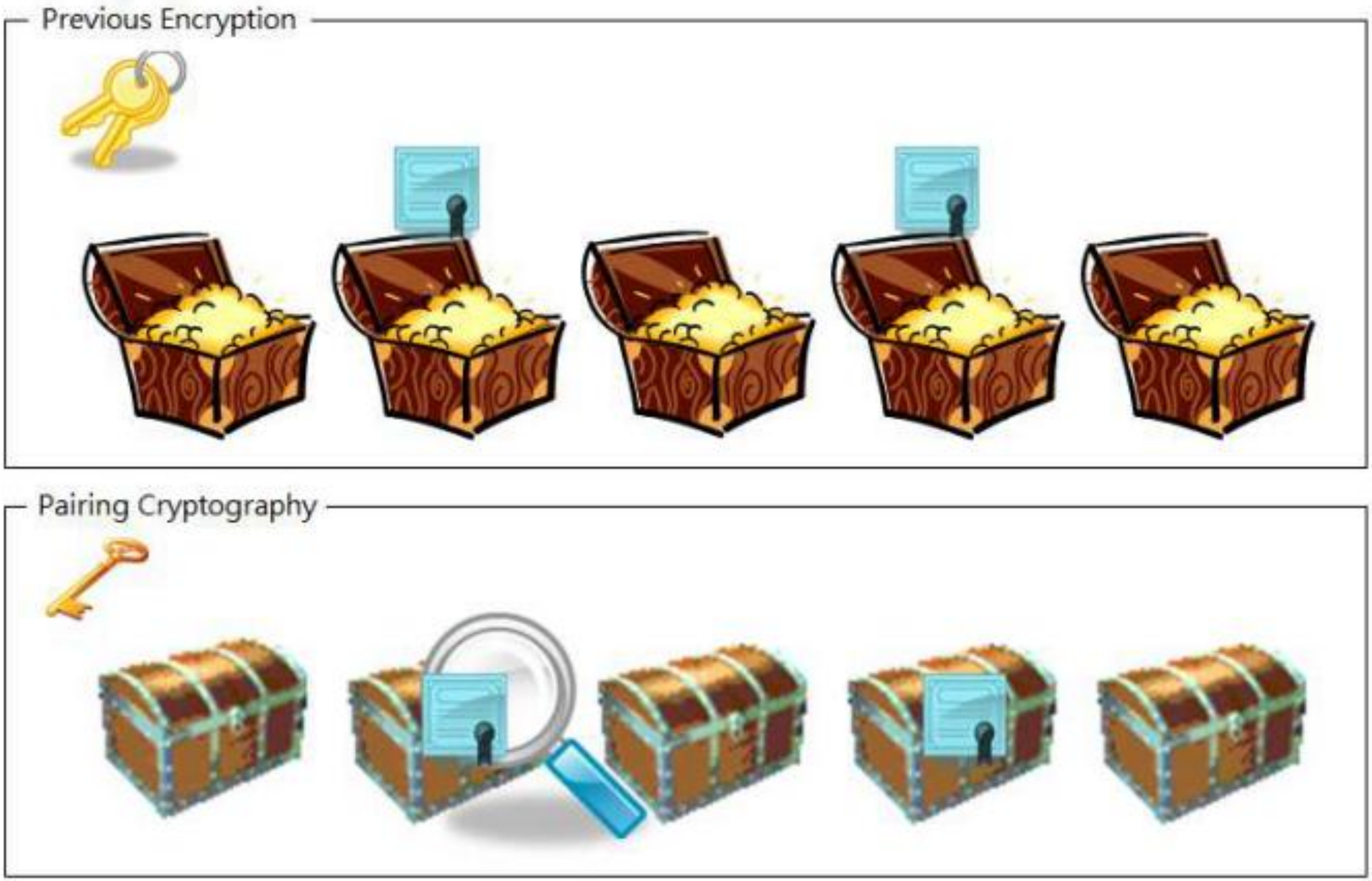
    $H : \{0,1\}^* \to G_1$ (Map to Point), and

    $H_1 : G_3 \to \{0,1\}^n$ for some chosen n.
  - The system parameters: $\langle G_1, G_3, \hat{e}, P, Q_0, n, H, H_1 \rangle$.
- **Extract:**
  - Given the ID of A $ID_A \in \{0,1\}^*$, the public key of A is $Q_A = H(ID_A)$.
  - The private key of A is $S_A = [s]Q_A$.

Protocol (2/2)

- **Encrypt:** Someone would like to encrypt message M for A.
  1. Get the public key of A by $Q_A = H(ID_A)$.
  2. Choose a random $t \in \mathbb{Z}_r^*$.
  3. The cipher $C = \langle tP, M \oplus H_1(\hat{e}(Q_A, Q_0)^t) \rangle$.
- **Decrypt:** A receives the encrypted message $C = \langle U, V \rangle$
  1. Check if $rU = \infty$.
  2. The message $M = V \oplus H_1(\hat{e}(S_A, U))$.

2. Searchable Encryption [BCOP 2003]



Goal

- Goal: searching on encrypted data.
- Example:
  - Bob sends email to Alice encrypted under Alice's public key.
  - Both contents and keywords are encrypted.
  - The email is stored on a mail server.
  - Alice want to specify a few keywords to read email.
  - The mail server should be able to search, but learn nothing else about the email.

BCOP Scheme

- Bob encrypts his email using a standard public key system.
- He appends to the resulting ciphertext a *Public-key Encryption with Keyowrd Search* (PEKS) of each keyword.
- To send a message M with keywords $W_1, \ldots, W_m$, Bob sends

$$E_{A_{pub}}(M) \quad \| \quad \text{PEKS}(A_{pub}, W_1) \quad \| \quad \ldots \quad \| \quad \text{PEKS}(A_{pub}, W_m)$$

- There is a certain trapdoor T_W for a specific keyword W.
- The mail server can test whether $W = W'$ by use of PEKS(A_{pub}, W') and T_W.
- If $W \neq W'$, the mail server learns nothing more about W'.

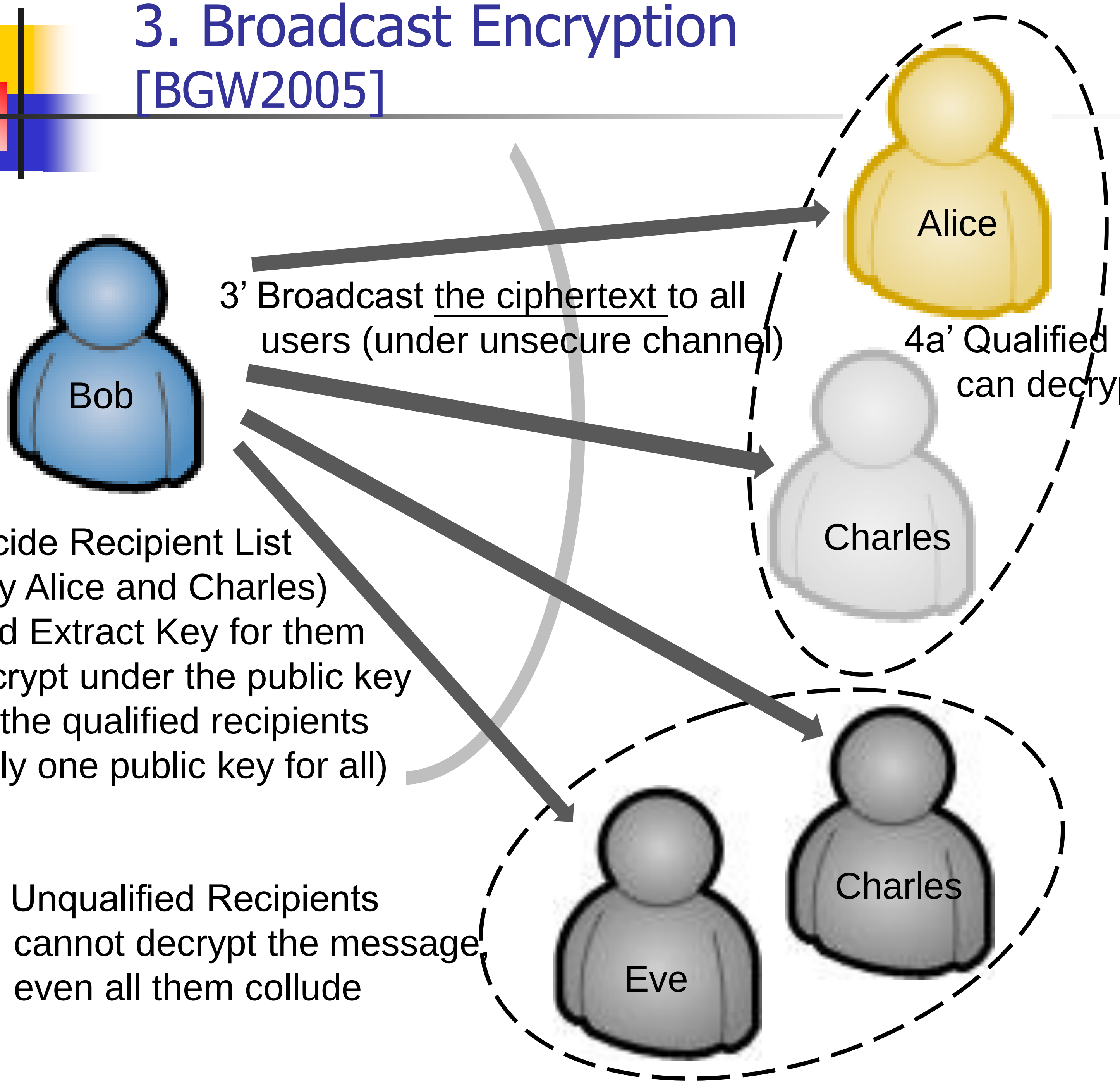
PEKS

- A public key encryption with keyword search scheme consists the following polynomial time randomized algorithms:
  - KeyGen(s): takes a security parameter, s, and generates a public/private key pair A_{pub}, A_{priv}.
  - PEKS(A_{pub}, W): for a public key A_{pub} and a work W, produces a searchable encryption of W.
  - Trapdoor(A_{priv}, W): given Alice's private key A_{priv} and a word W, produces a trapdoor T_W.
  - Test(A_{pub}, S, T_W): given Alice's public key A_{pub}, a searchable encryption S = PEKS(A_{pub}, W'), and a trapdoor T_W = Trapdoor(A_{priv}, W), outputs 'yes' if $W = W'$ and 'no' otherwise.

Construction of PEKS

- Using the Weil pairing $e : G_1 \times G_1 \to G_3$, where $|G_1| = |G_3| = p$.
- The hash functions: $H_1 : \{0,1\}^* \to G_1$ and $H_2 : G_3 \to \{0,1\}^{\log p}$.
- The PEKS works as follows:
  - KeyGen(s): The input security parameter determines the size, p, of the groups G_1 and G_3. The algorithm picks a random $\alpha \in \mathbb{Z}_p^*$ and a generator P of G_1. It outputs $A_{pub} = \{P, Q = [\alpha]P\}$ and $A_{priv} = \alpha$.
  - PEKS(A_{pub}, W): First compute $t = e(H_1(W), [r]Q) \in G_3$ for a random $r \in \mathbb{Z}_p^*$. Output PEKS(A_{pub}, W) $= \{[r]P, H_2(t)\}$.
  - Trapdoor(A_{priv}, W): output $T_W = [\alpha]H_1(W) \in G_1$.
  - Test(A_{pub}, S, T_W): let $S = \{A, B\}$. Test if $H_2(e(T_W, A)) = B$.
- $e(T_W, A) = e([\alpha]H_1(W), [r]P) = e(H_1(W), P)^{r\alpha} = e(H_1(W), [r]([\alpha]P)) = e(H_1(W), [r]Q)$

3. Broadcast Encryption
[BGW2005]



1’ Decide Recipient List
(say Alice and Charles)
and Extract Key for them

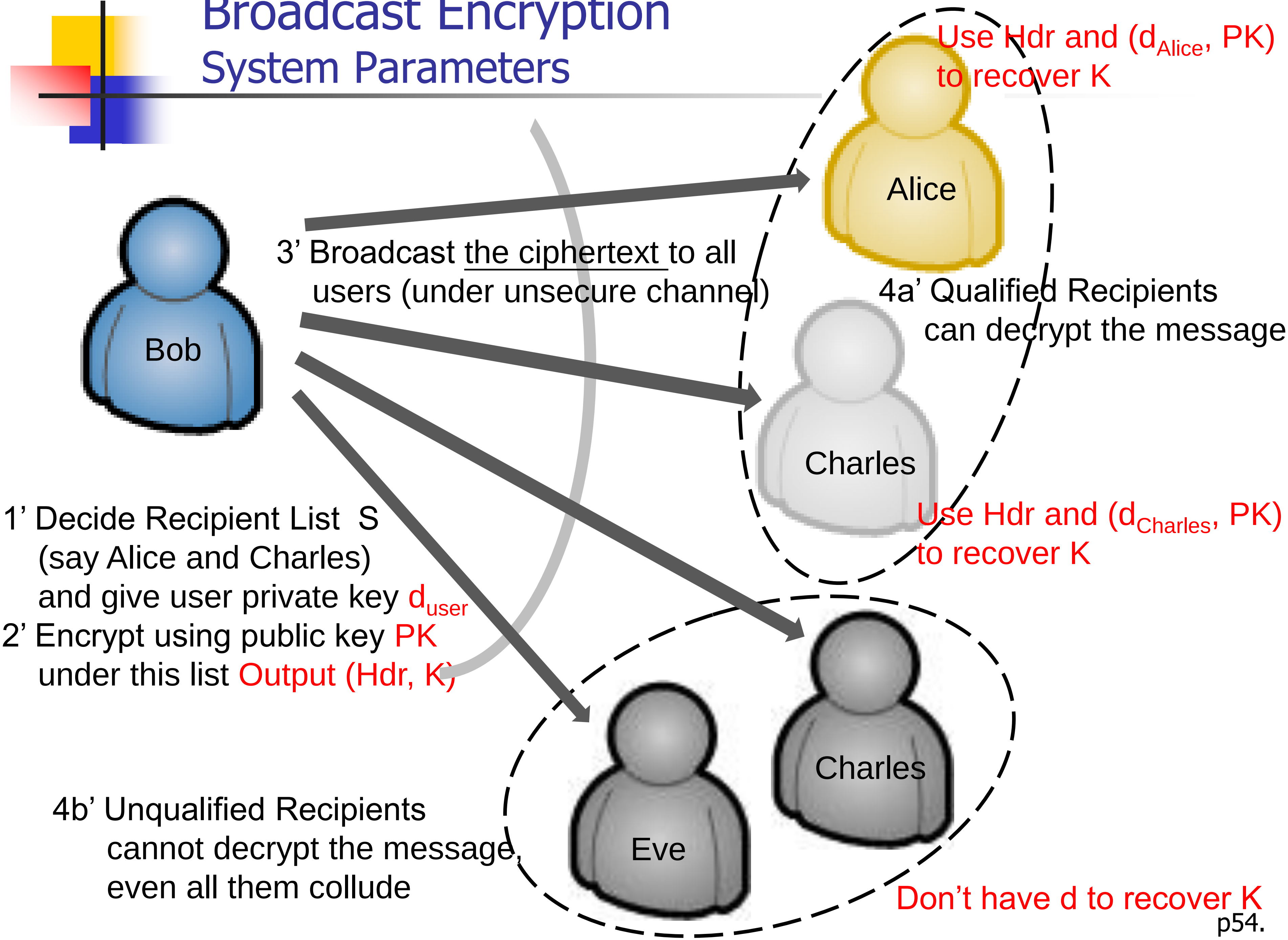
2’ Encrypt under the public key
for the qualified recipients
(only one public key for all)

3’ Broadcast the ciphertext to all
users (under unsecure channel)

4a’ Qualified Recipients
can decrypt the message

4b’ Unqualified Recipients
cannot decrypt the message,
even all them collude

Broadcast Encryption
System Parameters



BGW Scheme - Setup

- ***Setup(n)***
  - *in: # of intended users*
  - out: *n private keys (d_1, .. d_n), one public key PK*

Public Key: $$PK = (P, P_1, \ldots, P_n, P_{n+2}, \ldots, P_{2n}, v)$$

Private Key: $$d_i = \alpha^i v = \alpha^i \gamma P = \gamma P_i, i = i \ldots n$$

Where $P_i = \alpha^i P, v = \gamma P$

BGW Scheme - Encrypt

- ***Encrypt(S, PK)***
  - *in: $S \subseteq \{1, . . ., n\}$, public key PK*
  - *out: a pair (Hdr, K)*
    - *Hdr is called the header. (aka broadcast ciphertext)*
    - *$K \in \mathcal{K}$ is a message encryption key chosen from a finite key set $\mathcal{K}$.*

$$Hdr = (tP, t(v + \sum_{j \in S} P_{n+1-j}))$$

$$K = e(P_{n+1,} P)^t$$

BGW Scheme - Decrypt

- ***Decrypt(S, i, di, Hdr, PK)***
  - *If $i \in S$, then the algorithm outputs a message encryption key $K \in K$.*

$$Hdr = (tP, t(v + \sum_{j \in S} P_{n+1-j})) = (C_0, C_1)$$

$$K = \frac{e(P_i, C_1)}{e(d_i + \sum_{j \in S, j \neq i} P_{n+1-j+i}, C_0)}$$

Note: $d_i = \alpha^i v = \alpha^i \gamma P = \gamma P_i, i = i....n$

$P_i = \alpha^i P, v = \gamma P$

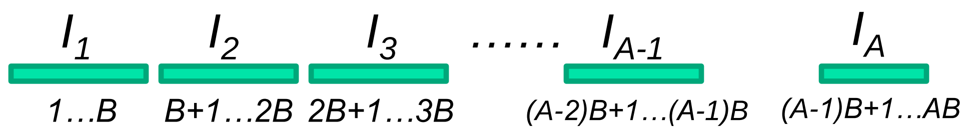
$$= e(P,P)^{t(\gamma\alpha^i + \sum_{j \in S} \alpha^{n+1-j+i}) - t(\gamma\alpha^i + \sum_{j \in S, j \neq i} \alpha^{n+1-j+i})} = e(P_{n+1,} P)^t$$

Session Key

If you don't have d_i, you cannot cross out this term to gain K

BGW Scheme – Setup (Generalized)

- ***IDEA: run A parallel instances of special case where each instance can broadcast to at most B<n users***
- ***Setup$_B$(n):*** $n = AB, A = \left\lceil \frac{n}{B} \right\rceil$



  - *in: # of intended users*
  - out: *n private keys (d_1, .. d_n), one public key PK*

Public Key: $PK = (P, P_1, \ldots, P_B, P_{B+2}, \ldots, P_{2B}, v_1, \ldots, v_A)$

Private Key: $d_i = \alpha^b v_a = \alpha^b \gamma_a P = \gamma_a P_b, i = i \ldots n$

Where $P_i = \alpha^i P, v_a = \gamma_a P$

Write i as $i = (a-1)B + b$

i.e. $a = \left\lceil \frac{i}{B} \right\rceil, b = i \bmod B$