Bilinear accumulators

Verifiable Computation (VC)

server

client

function f

result = f(data) +

digest δ

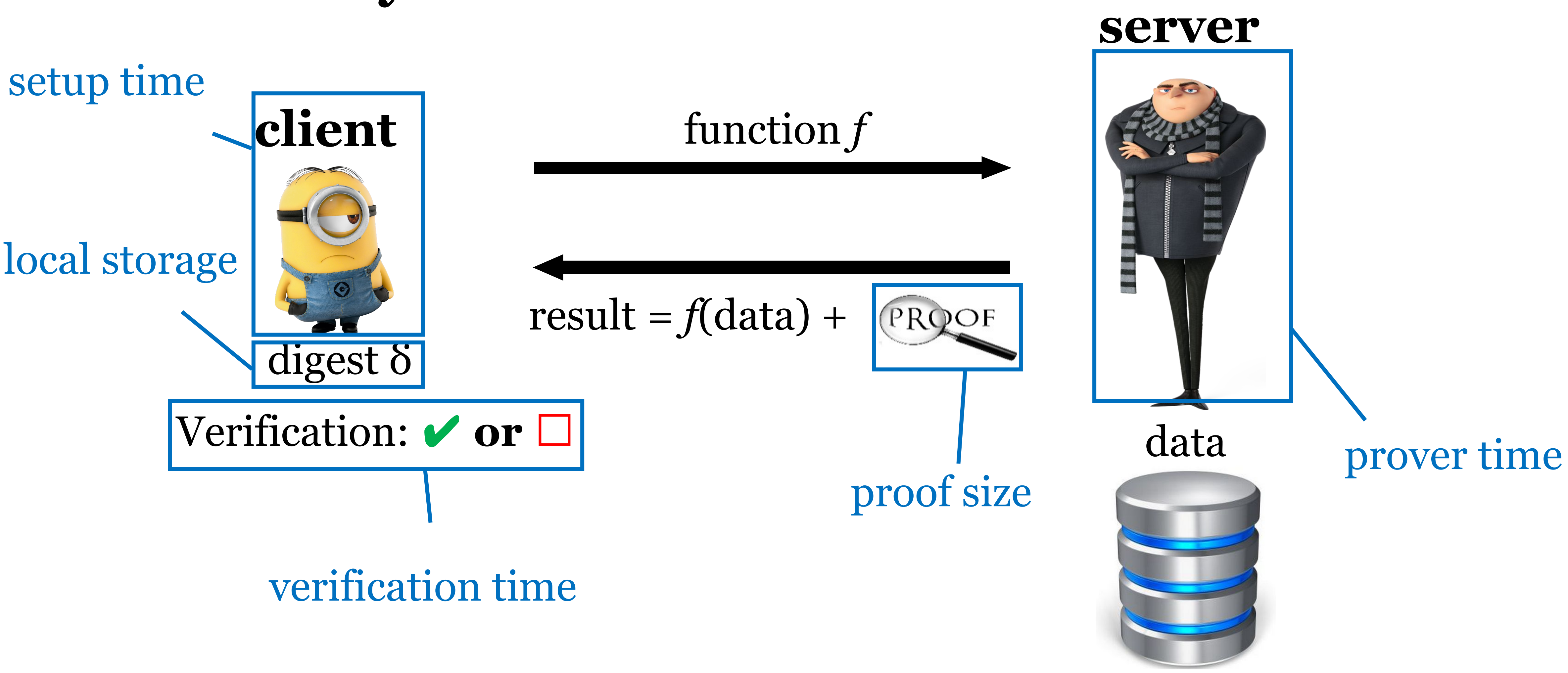
Verification: ✔ **or** ☐

data

Correctness/completeness: Pr[result = f(data) and proof is honest and verification is ✓] = 1

Soundness/security: Pr [result ≠ f (data) and verification is ✓] ≤ $\frac{1}{2^{100}}$

Efficiency measures



Group and field

Group: under 1 operation •

1. Closure: For all a, b in G, the result of the operation, a • b, is also in G
2. Associativity: For all a, b and c in G, (a • b) • c = a • (b • c)
3. Identity element: There exists an element e in G such that, for every element a in G, the equation e • a = a • e = a holds. Such an element is unique
4. Inverse element: For each a in G, there exists an element b in G, commonly denoted a−1 (or −a, if the operation is denoted "+"), such that a • b = b • a = e, where e is the identity element

Field: under 2 operations + and ×

1. F is an abelian group under + (abelian or commutative: a • b = b • a)
2. F – {0} (the set F without the additive identity 0) is an abelian group under ×.

Generator

An element that generates all elements in the group by repeating the operation on itself (Cyclic group*)

$Z_7^* = \{1,2,3,4,5,6\}$

$2^0 = 1;\ 2^1 = 2;\ 2^2 = 4;\ 2^3 = 1$

$3^0=1;\ 3^1 = 3;\ 3^2 = 2;\ 3^3 = 6;\ 3^4 =4;\ 3^5 = 5;\ 3^6 = 1$

Discrete-log

Z_p^* has an alternative representation as the powers of g:
$\{g, g^2, g^3, \dots, g^{p-1}\}$

Discrete-log: given $a \in Z_p^*$, find k s.t. $g^k = a$

Diffie-Hellman

x

Alice

g^{xy}

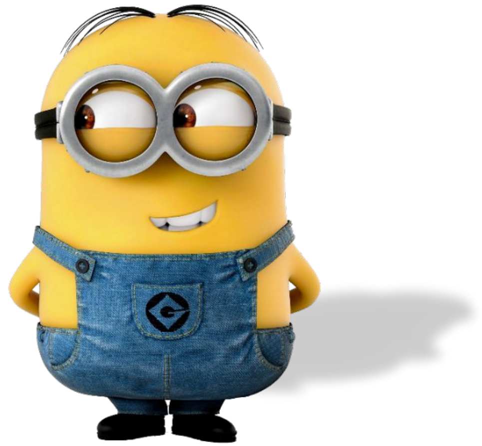
Z_p^* and generator g

$$\xrightarrow{g^x}$$

$$\xleftarrow{g^y}$$

y

Bob



g^{xy}

Diffie-Hellman assumption: give Z_p^*, g, g^x, g^y, cannot compute g^{xy}

RSA

Publick key: N, e

Enc(m, pk): $c = m^e \ mod \ N$

Dec (c, sk): $m = c^d \ mod \ N$

Alice

RSA assumption: given N,e, $c = m^e$, cannot find m

1. Pick random large primes p,q, publish N = pq
2. Compute $\phi(N) = (p-1)(q-1)$
3. Pick random $e \in Z^*_{\phi(N)}$, publish e
4. Compute d as the inverse of e in $Z^*_{\phi(N)}$

$$e * d = 1 \ mod \ \phi(N)$$

d is the private key

RSA accumulators

- Public: N, generator g
- Private: p, q

- Elements must be primes
- Accumulate set $\{x_1, x_2, \dots x_n\}$: digest $= \ g^{x_1 \cdot x_2 \cdot \dots \cdot x_n}\ mod\ N$
- Membership proof for x_i: $\pi_i = g^{x_1 \cdot \dots \cdot x_{i-1} \cdot x_{i+1} \cdot \ \dots \cdot x_n}$
- Verification: digest $= \ \pi^{x_i}$

Bilinear accumulator, prime field

- Idea 1: set $\{a_1, a_2, \dots a_n\}$, digest = $g^{a_1 \cdot a_2 \cdot \dots \cdot a_n} \, mod \, p$?

Bilinear accumulator, prime field

Characteristic polynomial of set $A = \{a_1, a_2, \ldots a_n\}$:

$$(a_1 + x)(a_2 + x)(a_3 + x) \ldots (a_n + x) = \Pi_{a \in A}(a + x)$$

Idea: replace variable x with a secret value s

$$\text{digest} = g^{\Pi_{a \in A}(a+s)}$$

Public key: $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \ldots, g^{s^q}$

q-strong Diffie-Hellman assumption: given $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \ldots, g^{s^q}$, cannot compute c, h s.t. $h = g^{\frac{1}{s+c}}$

Membership proof

- Setup: digest $= g^{\Pi_{a \in A}(a+s)}$ O(n) given s, O(n log n) given pk
- Membership proof for a_i: $\pi_i = g^{\Pi_{a \in A \setminus \{a_i\}}(a+s)}$
- Verification: $\pi_i^{a_i+s} =$ digest

- Security on q-strong Diffie-Hellman assumption

given $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$, cannot compute c, h s.t. $h = g^{\frac{1}{s+c}}$

Assumptions

- Discrete-log
- Diffie-Hellman: given Z_p^*, g, g^x, g^y, cannot compute g^{xy}

- exponent q-strong Diffie-Hellman: $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$, cannot compute $h = g^{s^{q+1}}$
- q-weak Diffie-Hellman: $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$, cannot compute $h = g^{\frac{1}{s}}$
- q-strong Diffie-Hellman: $p, g, g^s, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$, cannot compute c, h s.t. $h = g^{\frac{1}{s+c}}$

Public verifiability

s

Alice

$p, g, g^{s}, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$

Bob

Is element a_i in the set?

result + 

$$\text{digest} = g^{\prod_{a \in A}(a+s)}$$

Verification: $\pi_i^{a_i+s} = \text{digest}$

Solution: bilinear group

- $(p, G, g, \ G_T, e\ )$:

  - G and G_T are both multiplicative cyclic group of order p, g is the generator of G. G:base group, G_T target group

  - Pairing: $e\left(P^a, Q^b\right) = e(P, Q)^{ab} : G \times G \rightarrow G_T$ I.e., $e\left(g^a, g^b\right) = e(g, g)^{ab}$

Bilinear map

s

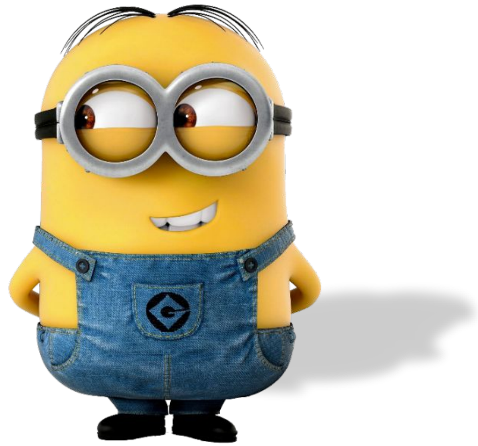
Alice

$(p, g, G, e, G_T),$
$g^s, g^{s^2}, g^{s^3}, g^{s^4}, \dots, g^{s^q}$

Bob

Is element a_i in the set?

result + 



$$\text{digest} = g^{\prod_{a \in A}(a+s)}$$

$$\pi_i = g^{\prod_{a \in A \setminus \{a_i\}}(a+s)}$$

Verification: $e(\text{digest}, g) = e(\pi_i, g^s \cdot g^{a_i})$

Bilinear map

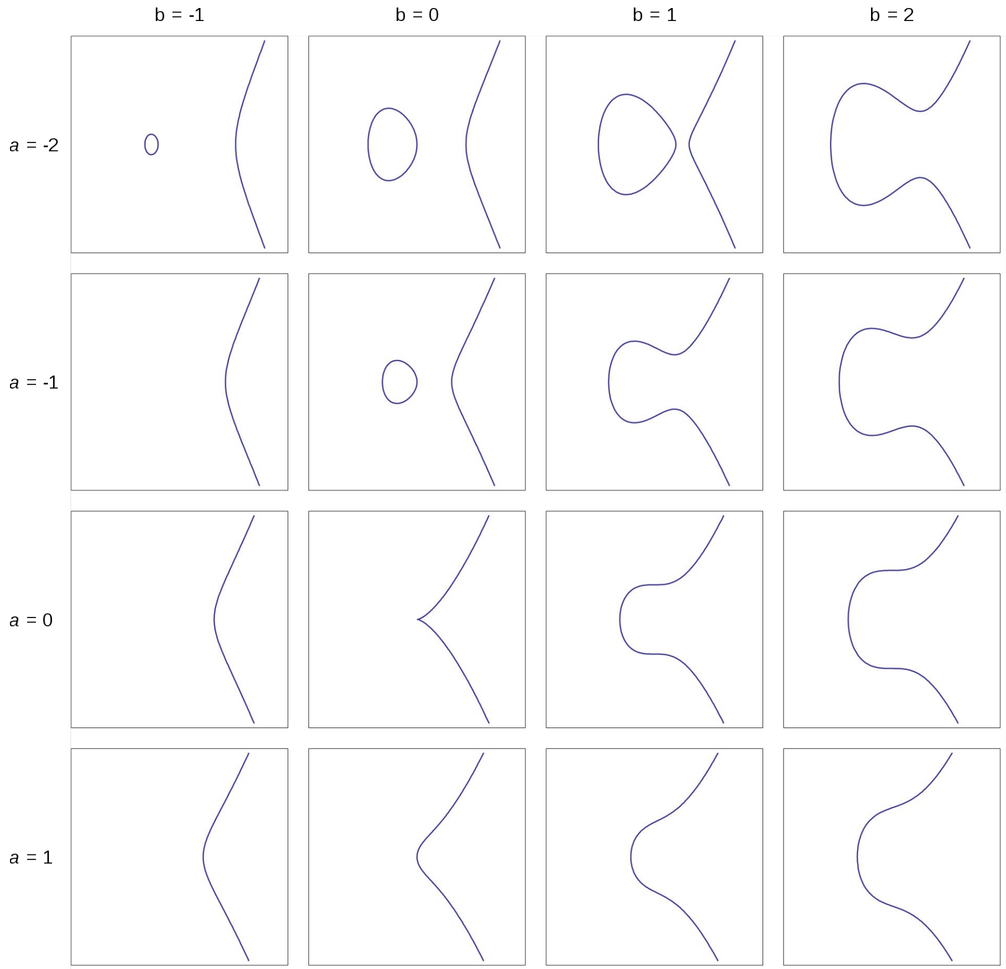
- Security relies on q-strong **bilinear** Diffie-Hellman assumption:

  given $(p, \mathbb{G}, g, e, G_T), g^{s}, g^{s^2}, g^{s^3}, g^{s^4}, \ldots, g^{s^q}$, cannot compute $c, h \text{ s.t. } h = e(g,g)^{\frac{1}{s+c}}$

- Applies to subset $A \subset B$

Elliptic curve

- $$y^2 = x^3 + ax + b$$



Non-membership

- Extended Euclidean algorithm for polynomials
  - If $p(x)$ and $q(x)$ has no common root
  - Find $p(x)a(x) + q(x)b(x) = 1$ in O(n $\log^2$ n loglog n)

- If $p(x) = a_i + x$, $q(x) = \Pi_{a \in A}(a + x)$, find $a(x), b(x)$, set $\pi = g^{a(s)}, g^{b(s)}$

- Verification: $e(g^{a_i+s}, \pi_1)e(\text{digest}, \pi_2) = e(g, g)$

Intersection is empty

- Extended Euclidean algorithm for polynomials
  - If $p(x)$ and $q(x)$ has no common root
  - Find $p(x)a(x) + q(x)b(x) = 1$ in $O(n \log^2 n \log\log n)$

- If $p(x) = \Pi_{a \in A}(a + x)$ $q(x) = \Pi_{a \in B}(a + x)$, find $a(x), b(x)$, set $\pi = g^{a(s)}, g^{b(s)}$

- Verification: $e(\mathrm{digest}_{\mathrm{A}}, \pi_1)e(\mathrm{digest}_{\mathrm{B}}, \pi_2) = e(g, g)$

RSA Non-membership proofs

- x is not in the set, then x and $u = x_1 \cdot \ldots \cdot x_n$ are co-prime
- Extended Euclidean algorithm: find $ax + bu = 1$
- Proof $a, d = g^b$
- Verification: $\delta^a d^x = g$

- Cannot be generalized to two sets!!

Set intersection

- $I = A \cap B \iff 1.\, I \subset A, I \subset B$ and $2.\, (A - I) \cap (B - I) = \emptyset$

- Proof:
  1. $\pi_A = g^{\Pi_{a \in A-I}(a+s)}, \pi_B = g^{\Pi_{a \in B-I}(a+s)}$
  2. $p(x)a(x) + q(x)b(x) = 1, p(x) = \Pi_{a \in A-I}(a + x)\;\; q(x) = \Pi_{a \in B-I}(a + x),$
     $\pi_1, \pi_2 = g^{a(s)}, g^{b(s)}$

- Verification:
  1. $e(\mathrm{digest}_\mathrm{A}, g) = e\big(\, g^{\Pi_{a \in I}(a+s)}, \pi_A\big), e(\mathrm{digest}_\mathrm{B}, g) = e\big(\, g^{\Pi_{a \in I}(a+s)}, \pi_B\big)$
  2. $e(\pi_A, \pi_1)e(\pi_B, \pi_2) = e(g, g)$

Complexity

- Local storage, size of accumulator: $O(1)$
- Setup: $O(n \log n)$
- Prover time: $O(1)$ with $O(n)$ storage, or $O(n \log n)$
- Proof size: $O(1)$
- Verification time: $O(1)$

- Update: add $O(1)$, delete $O(1)$ with secret key; $O(n)$ without secret key for both
- Update proof: add $O(1)$, delete $O(1)$ both with new digest

- Set operations: prover $(n \log^2 n \log\log n)$, proof size $O(1)$, verification time $O(1)$