Discrete Square Roots in $\mathbb{Z}_p$

Q: Given y, find x: $x^2 = y$

- 0
- $\frac{p-1}{2}$ elements with 2 square roots — Quadratic residues
- $\frac{p-1}{2}$ elems with no square root — Quadratic non-residues

- quadratic poly has at most 2 roots
- if $x^2 = y$, then $(-x)^2 = y$
  so x is sqrt $\Leftrightarrow$ $-x$ is sqrt
  $\Rightarrow$ non-zero y has even # of sqrts

} 0 or 2 sqrts

$\Rightarrow$ set of QRs is a subgroup of $\mathbb{Z}_p^*$

in $\mathbb{Z}_5$: $1^2 = 4^2 = 1$, $2^2 = 3^2 = 4$, $0^2 = 0$

$\exists g$ generator

$\mathbb{Z}_p^* = \{ g^0, g^1, g^2, \dots, g^{p-2} \}$

$g^{2k} \to g^k$ is a sqrt

↑ $\frac{p-1}{2}$ elts of this kind

all sqrts used $\Rightarrow$ no other QRs

So: parity of $\text{dlog}(x)$ tells if x is a QR.

$x \cdot y = (a \cdot b)^2$, $x = a^2$, $y = b^2$

Thm: x is a QR $\Leftrightarrow$ $x^{\frac{p-1}{2}} \equiv 1 \pmod p$

Pf: $(g^{2k})^{\frac{p-1}{2}} \equiv g^{2k \cdot \frac{p-1}{2}} \equiv (g^{p-1})^k \equiv 1^k \equiv 1$

$(g^{2k+1})^{\frac{p-1}{2}} = \underbrace{(g^{2k})^{\frac{p-1}{2}}}_{1} \cdot \underbrace{g^{\frac{p-1}{2}}}_{\text{sqrt}(1)} \equiv -1$

Legendre symbol

$\left(\frac{x}{p}\right) := x^{\frac{p-1}{2}} \bmod p$

p prime

$$\left(\frac{x}{p}\right) = \begin{cases} 0 & p \mid x \\ +1 & \text{if } x \text{ is QR mod } p \\ -1 & \text{if non-QR mod } p \end{cases}$$

$\left(\frac{xy}{p}\right) = \left(\frac{x}{p}\right) \cdot \left(\frac{y}{p}\right)$

homomorphism $\mathbb{Z}_p^* \to (\{-1, +1\}, \cdot)$

Computing the sqrt:

- mod p: randomized alg. [Tonelli & Shanks] 1891 1973
- mod composite n: — if we can factorize n: use CRT
  — otherwise hard
  ↳ Rabin cryptosystem

RSA [Rivest, Shamir, Adleman 1978 / GCHQ 1973 public 1997]

Messages: $\mathbb{Z}_n$

Setup: p, q big random primes ($p \neq q$) secret!

$n := p \cdot q$ modulus

$\varphi(n) = (p-1)(q-1)$

$e \perp \varphi(n)$ encryption exponent

d: $e \cdot d \equiv 1 \pmod{\varphi(n)}$ decryption exponent

(e, n) encr. key

(d, n) decr. key

Encryption: $E(x) := x^e \bmod n$

Decryption: $D(y) = y^d \bmod n$

Prove: $(x^e)^d \equiv x^{ed} \equiv x^{1+k\cdot\varphi(n)} \equiv x \cdot \underbrace{x^{k\cdot\varphi(n)}}_{(x^{\varphi(n)})^k \equiv 1^k \equiv 1} \equiv x \pmod n$

by Euler's thm

Works only for $x \perp n$ ($p, q \nmid x$)

for $x = p \cdot x'$ ... prove mod p, mod q, combine using CRT

Efficiency: slow, but polynomial → use hybrid ciphers ... (see Lecture #11)

↳ Improvements: ① choose small public exponent ($3, 17, 65537 = 2^{16}+1$)

② use CRT for private calculations (compute mod p, mod q)

Commutative with the same modulus

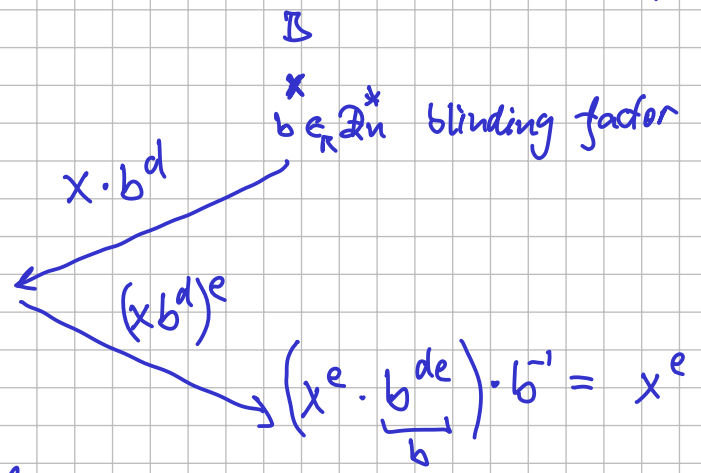
$$D_2(D_1(E_2(E_1(x)))) = x$$

$(x_1 \cdot x_2)^e \equiv x_1^e \cdot x_2^e$

Homomorphic: $E(x_1 \cdot x_2) = E(x_1) \cdot E(x_2)$

App: Blind signatures Alice signs anything

Bob wants message x signed, but keep it secret from Alice



Attacks on RSA

$p \cdot q = n$

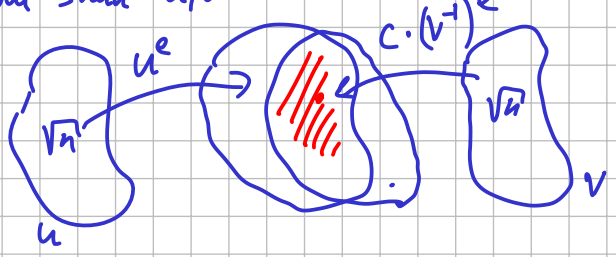
$(p-1)(q-1) = \varphi(n)$

$pq - p - q - 1$

$p+q$ known

- if $x < n^{1/e}$ ⇒ decryption is e-th root in $\mathbb{Z}$
- if anybody knows $\varphi(n)$ → factorize n
- if e, d are known → factorize n
- [Wiener 1990] if $d < n^{1/4}$, $e \to d$ probabilistically
- Meet-in-the middle: we know $c \equiv m^e$

try random "small" u, v



Likely we find:

$u^e \equiv \underbrace{c}_{m^e} \cdot v^{-e}$

$u^e \cdot v^e \equiv m^e$

$(uv)^e \equiv m^e$

$uv \equiv m$

- **Similar messages**: m $\quad m+\delta$
  $c \equiv m^e \quad c' \equiv (m+\delta)^e$
  m is root of $p(x) = x^e - c$
  $p'(x) = (x+\delta)^e - c'$
  } if e is small, p, p' have low degree
  x is root $\gcd(p, p')$ with high pr. this is degree 1
- p, q are close to each other ... factorize (ex.)
- Single message encrypted by same e, but different moduli
  Example: $e=3$ 3 diff. moduli n_1, n_2, n_3
  $(3, n_1)$ $(3, n_2)$ $(3, n_3)$ public keys

  we know: $x^3 \equiv y_1 \pmod{n_1}$ $\quad N := n_1 \cdot n_2 \cdot n_3$
  $x^3 \equiv y_2 \pmod{n_2}$ $\quad x < \min(n_1, n_2, n_3)$
  $x^3 \equiv y_3 \pmod{n_3}$ $\quad x^3 < N$

  ↳ by CRT we calculate x^3
  & compute cube root in $\mathbb{Z}$.

  Solutions: high e / randomize messages

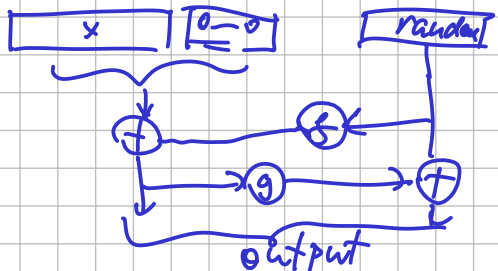
- **Semantic Security of RSA**
  ↑ any property of plaintext is not (efficiently) computable from the ciphertext
  ↑ efficiently testable
  - RSA leaks the $\left(\frac{x}{p}\right) \cdot \left(\frac{x}{q}\right)$ (Jacobi symbol)
    ↳ ~1 bit of information
  - computing parity(x) from $E(x)$ is equivalent to full decryption

Use Padding!!

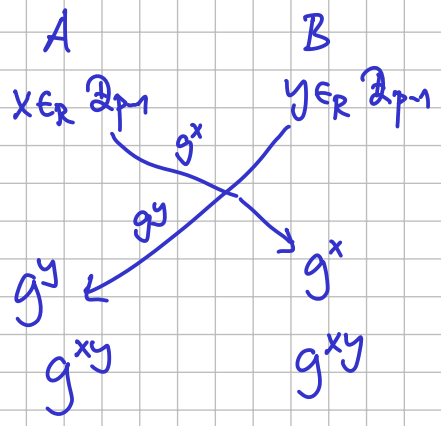
PKCS v1.5

! attack using padding oracle !
[Bleichenbacher 1998]

PKCS v2.0

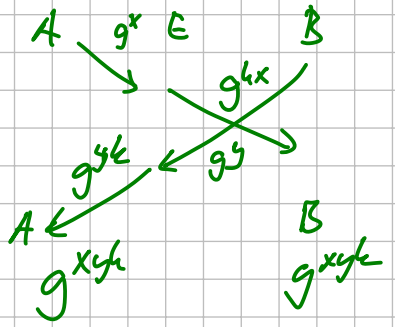


Diffie-Hellman Key Exchange Protocol



Parameters: public
- p ... a prime
- g ... generator of $\mathbb{Z}_p^*$

Problems:
- Man-in-the-middle
  $A \longleftrightarrow E \longleftrightarrow B$
  Solution: sign the result
- attack on parameters: replace g by g^k
  ↳ check the parameters! ↳ generates $H \subset \mathbb{Z}_p^*$
- powering attack



... A/B tricked by E into a subgroup

→ sign the whole communication

- D-H leaks whether g^{xy} is a QR → ≤ 1 bit leak
- Common trick: $p = 2q - 1$ } safe primes
  (q also prime)

$|\mathbb{Z}_p^*| = 2q$ only subgroups of $\mathbb{Z}_p^*$: $\{\pm 1\}$, quadratic residues

use g^2 instead of g

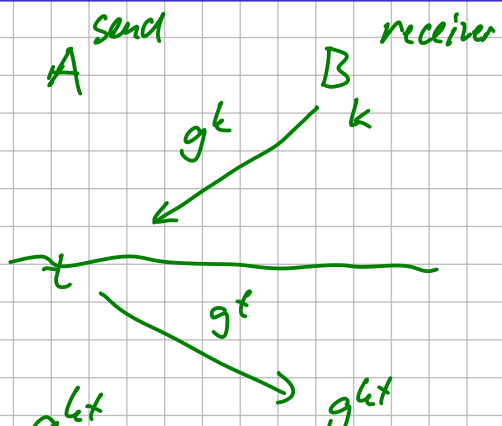
ElGamal cipher based on DH

Params: prime p, generator of g

Keys: $k \in_R \{0 \dots p-2\}$ secret
$h = g^k \bmod p$ public

Encryption: $t \in_R \{0 \dots p-2\}$ g^{kt}
$s = h^t$
$y = x \cdot s$
} send (g^t, y)

Decryption: $s = (g^t)^k$ g^{kt}
$x = y \cdot s^{-1} \pmod p$



More:
- different algebraic structures in which dlog is hard
  ↳ elliptic curves

Analogous protocols for asym. signature:
ElGamal signature, DSA