Signature

message m (document to be signed)

Alice (signer) — **Bob** (verifier)

private key a
public key A

Signing

create signature S (as a function of message m, private key a)

$\xrightarrow{S}$

verifying

as a function of S, m, A

returns: "valid" or "invalid"

Properties: 1) Verifying should return "valid" on a well-formed signature

2) It should be computationally infeasible to forge a signature on a given document for a given public key, i.e. create S so that S, m, A verifies as "valid".

Gives: authenticity: only Alice could have signed
integrity: document can't be changed after signing
non-repudiation: signer can't claim they didn't sign.

RSA Signature

Idea: pretend document is a "ciphertext" & "decrypt" it

Alice

private key: d (dec. exp.)

public key: $n = pq$ (modulus)

e (encr. exp.)

message m

Signing: $S = m^d \pmod n$

S

Bob

S

Verifying:

$V \equiv S^e \pmod n$

If $V \equiv m$ then return valid

O/w invalid.

Forging? Find S, given m, n, e Exactly RSA problem

"plaintext" "ciphertext"

Existential forgery Start w/ S, Compute $m = S^e \pmod n$

Then $S, m, (e, n)$ will return valid

but m is gobbledeegook.

Hash function (in signatures, hash the message to make it shorter for signing.)

A cryptographic hash f^n:

"message" → "digest" or "hash"

$h: \{0,1\}^* \longrightarrow \{0,1\}^k$

$\{0,1\}^*$: binary strings of arbitrary length

$\{0,1\}^k$: strings of length k

Warning!

∞-to-1 function (not injection)

w/ properties:

1) efficient to compute
2) preimage resistance: Given digest d, it is computationally infeasible to find m w/ $h(m) = d$.
3) collision-resistance: Computationally infeasible to find $m_1 \neq m_2$ s.t. $h(m_1) = h(m_2)$.
4) avalanche effect: small changes in input should in very different hash.

Ex / Non-ex.

① checksum: given message m, $h(m) = \Sigma$ bits of m (mod 2)

eg. $m = 1101$, $h(m) = 1$

Terrible.

In real world.

MD5 (90's) – seconds to find a collision

SHA-1 (95 ish) – broken 2017

RIPEMD – Whirlpool

SHA-2 (NIST, 2001)

SHA-3 (NIST, 2015)

} length extension attacks

Give an example system of equations (give a, b)

$$\begin{cases} x \equiv a \pmod{15} \\ x \equiv b \pmod{6} \end{cases}$$

variable x

which has no solutions.

$a = 2 \quad b = 1$

$$x \equiv 2 \pmod{15} \implies x \equiv 2 \pmod{3}$$
$$x \equiv 1 \pmod{6} \implies x \equiv 1 \pmod{3}$$

↺ ✗

How many solutions can $x^2 \equiv a \pmod{6}$ have?

$a = 5, 2$ no sol^{n}s

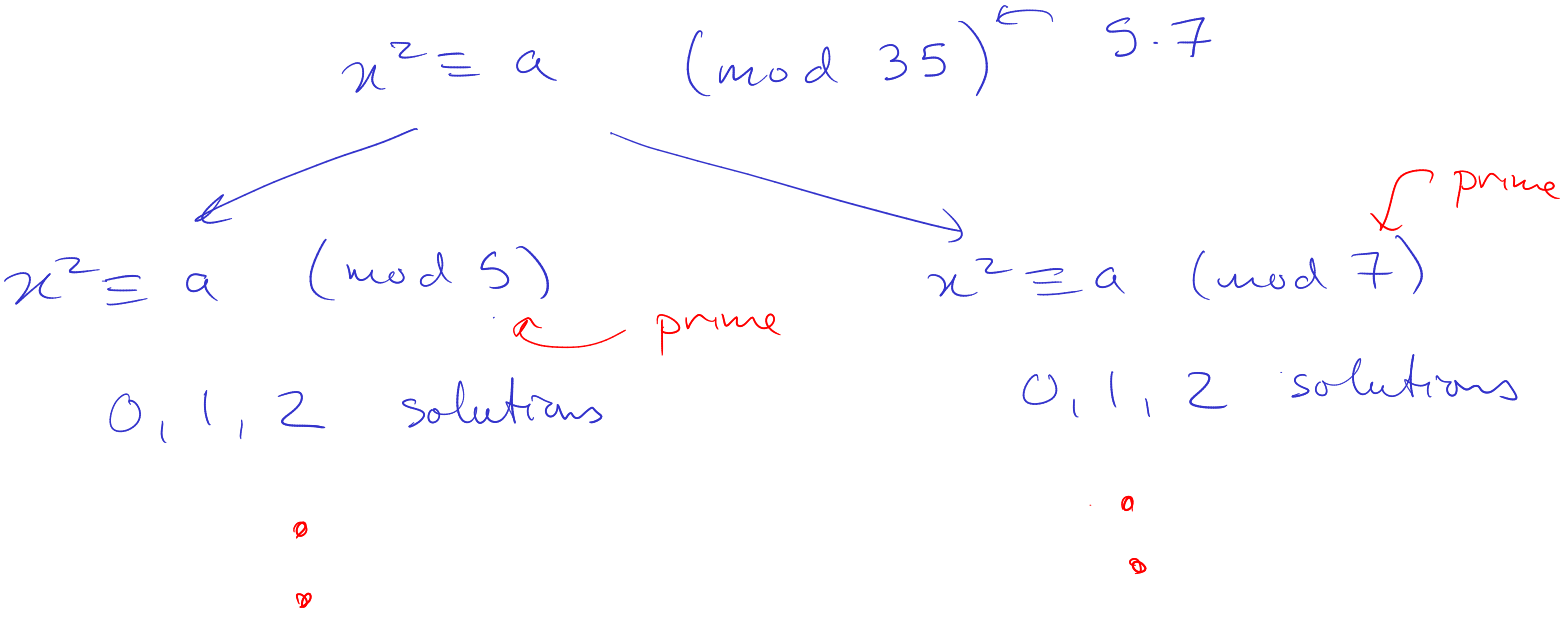
$a = 0, 3$ 1 sol^{n}s

$a = 1, 4$ 2 sol^{n}s

$1^2 \equiv 1 \quad 4^2 \equiv 4$

$2^2 \equiv 4 \quad 5^2 \equiv 1$

$3^2 \equiv 3 \quad 0^2 \equiv 0$


$x^2 \equiv a \pmod{35}$
5 · 7
$x^2 \equiv a \pmod{5}$
prime
0, 1, 2 solutions
$x^2 \equiv a \pmod{7}$
prime
0, 1, 2 solutions