Public-Key Cryptography

Aggelos Kiayias

A Basic Problem in Cryptography

“Alice”

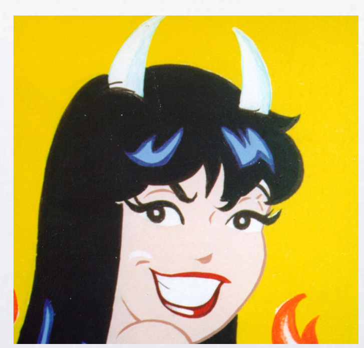
“Bob”

secret message from Alice →

← *secret message from Bob*

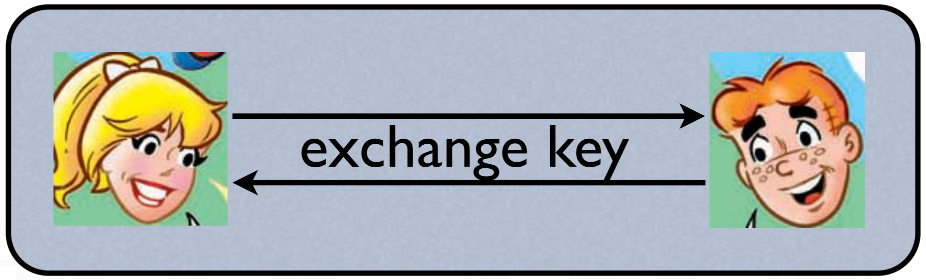
?

“Eve”



Secret-Key Cryptography

Secret-Key Cryptography



"Alice"

secret messages

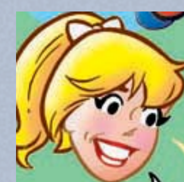
"Bob"

"Eve"

what is a "key"?

0101101001010111101
0101010101110111101
0001000100001011011
0010101101000100001
0111101011010110111

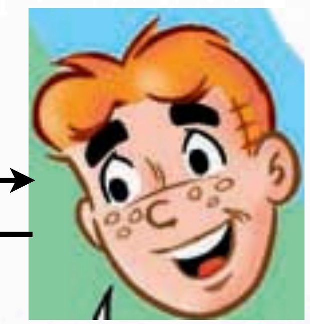
Secret-Key Cryptography


exchange key


“Alice”

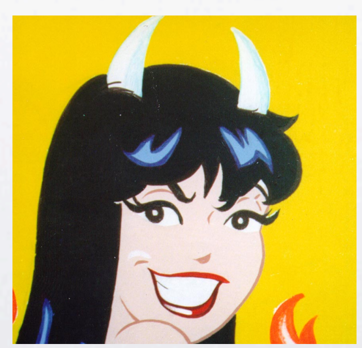
secret messages

“Bob”



C. Shannon :
perfect security
can only be achieved
when |key|=|msg|

“Eve”



what is a “key”?
0101101001010111101
0101010101110111101
0001000100001011011
0010101101000100001
0111101011010110111

Public Key-Exchange?

A paradox

- From a certain point of view it is impossible:
  - Alice and Bob should come up with some sort of secret information just by looking at data placed in the insecure channel.
  - Any information they can extract, the adversary can extract as well.
    - ***key question*** *in how much time?*

Using Time Complexity

"Alice"

read/send messages in time T

secret messages till time T'

"Bob"

read/send messages in time T

"Eve"

read messages in time T' > T

Merkle's Puzzles

Ralph Merkle, Secure Communications Over Insecure Channels, 1978

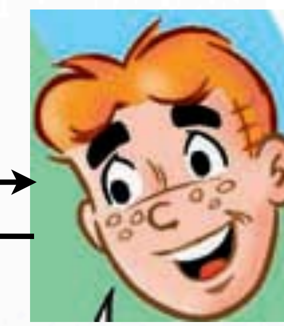
Alice generates puzzles

N puzzles

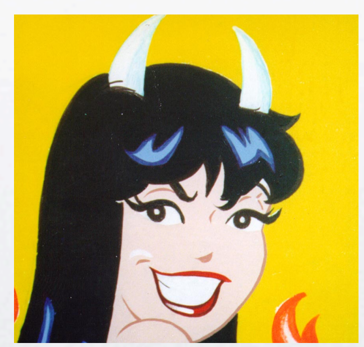
$P_1, \ldots, P_N$

Use X_i as the key

Bob chooses **one** puzzle to solve



Each puzzle requires N steps to solve



Choose i randomly

Find X_i, the solution :

$X_i : P_i(X_i) = \text{solved}$

Time
N+dec

Merkle's Puzzles

Ralph Merkle, Secure Communications Over Insecure Channels, 1978

Alice generates puzzles

N puzzles

$P_1, \ldots, P_N$

Use X_i as the key

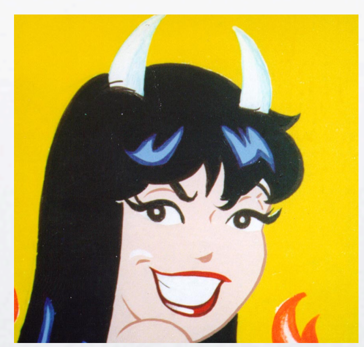
Bob chooses **one** puzzle to solve

Choose i randomly

Find X_i , the solution :

$X_i : P_i(X_i) = \text{solved}$

Each puzzle requires N steps to solve



Time N+dec

Merkle's Puzzles

Time N+enc

Ralph Merkle, Secure Communications Over Insecure Channels, 1978

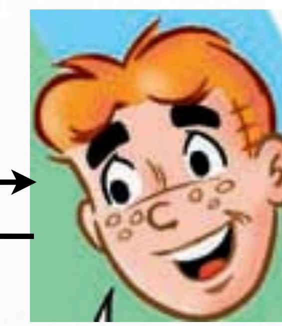
Alice generates puzzles

N puzzles

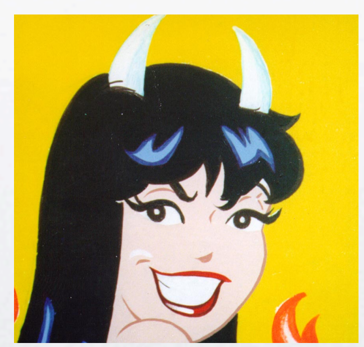
$P_1, \ldots, P_N$

Use X_i as the key

Bob chooses **one** puzzle to solve



Each puzzle requires N steps to solve



Choose i randomly

Find X_i, the solution :

$X_i : P_i(X_i) = \text{solved}$

Merkle's Puzzles

Ralph Merkle, Secure Communications Over Insecure Channels, 1978

Time
N+dec

Time
N+enc

Alice generates puzzles

Bob chooses **one** puzzle to solve

N puzzles

$P_1, \ldots, P_N$

Use X_i as the key

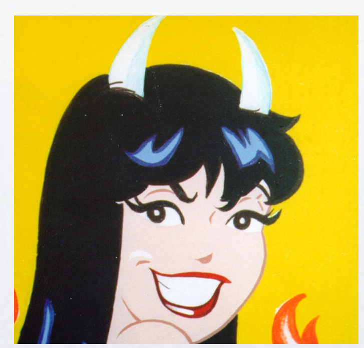
Choose i randomly

Find X_i , the solution :

$X_i : P_i(X_i) = \text{solved}$

Each puzzle requires N steps to solve

Time
~N²



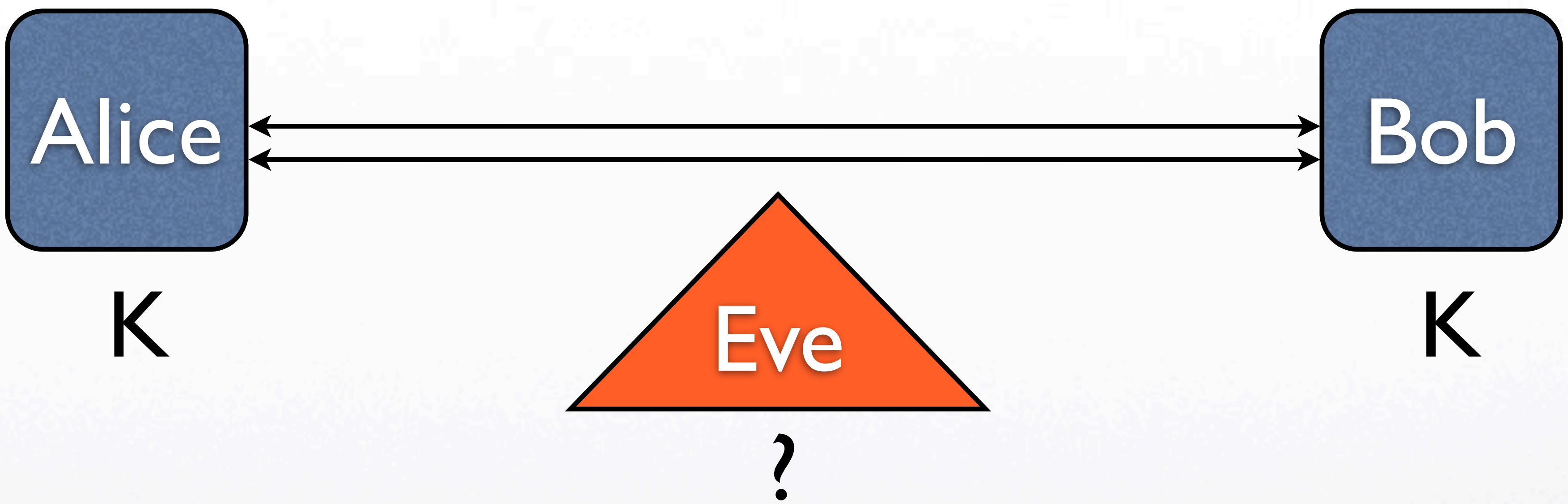
Can we do better?

polynomial-time

- Make computation for Alice and Bob feasible.
- Make computation for Eve infeasible.

exponential-time

Key Exchange



Modular Arithmetic

$$\mathbb{Z} = \{0, \pm 1, \pm 2, \ldots\} \qquad \mathbb{Z}_n = \{a \bmod n \mid a \in \mathbb{Z}\}$$

$$a \bmod n = \min\Big(\{a - v \cdot n \mid v \in \mathbb{Z}\} \cap \{0, 1, 2, \ldots\}\Big)$$

if $\overline{a}, \overline{b} \in \mathbb{Z}_n$ then $\begin{array}{l} \overline{a} + \overline{b} = (a + b) \bmod n \\ \overline{a} \cdot \overline{b} = (a \cdot b) \bmod n \end{array}$ definition

if $g \in \mathbb{Z}_n$ define $g^x = \underbrace{g \cdot g \cdot \ldots \cdot g}_{x}$ definition

A group theoretic problem

- *Given* finite multiplicative group.
- Consider cyclic subgroup generated by g

$$g, g^2, g^3, g^4, \dots$$

A problem that will concern us:

Given $h \in \langle g \rangle$ find min $x \in \mathbb{Z}$ such that $g^x = h$

Easy?

Hard?

Order of group

$$\exists k > 0 : g^k = 1$$

$$\{1, g, g^2, \ldots, g^{|G|}\} \longrightarrow g^k = g^l$$

$$k > l \to g^{k-l} = 1$$

Lagrange : $\text{order}(g) \mid |G|$

Exponentiation

$$
\begin{aligned}
g^x &= g^{x_0 + 2x_1 + \ldots + 2^{\nu-1} x_{\nu-1}} \\
&= g^{x_0} \cdot (g^2)^{x_1} \cdot \ldots (g^{2^{\nu-1}})^{x_{\nu-1}} \\
&= \prod_{\ell : x_\ell = 1} g^{2^\ell}
\end{aligned}
$$

Easy to compute.

Required: ν group squarings

$\#\{\ell : x_\ell = 1\}$ group operations

$$\nu = \lceil \log_2(\text{order}(g)) \rceil$$

Discrete-Logarithm Computation

Given $h \in \langle g \rangle$ find min $x \in \mathbb{Z}$ such that $g^x = h$

The obvious algorithm tries all possible $x \in \mathbb{Z}$

Is this efficient?

Order of g is known.
Order of g is unknown: (but we know it must divide order of the group)

2^ν group operations

Number Theory Problems

definition

$$\text{Prime} = \{p \mid p \in \mathbb{Z}^+ \ \forall a, b \in \mathbb{Z} : p = a \cdot b \Rightarrow \{a, b\} = \{1, p\}\}$$

The Discrete-Logarithm Problem (over a prime finite field):

Given $p \in \text{Prime}, g \leftarrow_R \mathbb{Z}_p, y = g^x \bmod p$ Find: x

The Factoring Problem

Given $n = pq$

Find: p, q

The $e-\mathrm{th}$ Root Problem

Given $z \in \mathbb{Z}_n, e \in \text{Prime}\ n = pq$

Find: $z^{1/e} \bmod n$

Diffie Hellman KE

public

Alice $\longleftarrow$ g, p $\longrightarrow$ Bob

$$y_A \leftarrow g^{x_A} \bmod p \qquad\qquad y_B \leftarrow g^{x_B} \bmod p$$

$$\xrightarrow{\quad y_A \quad}$$

$$\xleftarrow{\quad y_B \quad}$$

$$K = g^{x_B \cdot x_A} = y_B^{x_A} \qquad\qquad K = g^{x_A \cdot x_B} = y_A^{x_B}$$

Public-Key Cryptography

- Alice wants to send a message to Bob.
- Bob publishes his public-key.
- Alice reads Bob's public-key.
- Encrypts the message with the public-key.
- Transmits the ciphertext.
- Bob decrypts the ciphertext with the secret-key

PK Encryption

$$(Gen, Enc, Dec)$$

$$(pk, sk) \leftarrow Gen(1^k) \qquad k: \text{ key size in bits.}$$

$$Enc(pk, M) = \text{ a distribution of ciphertexts encoding } M$$

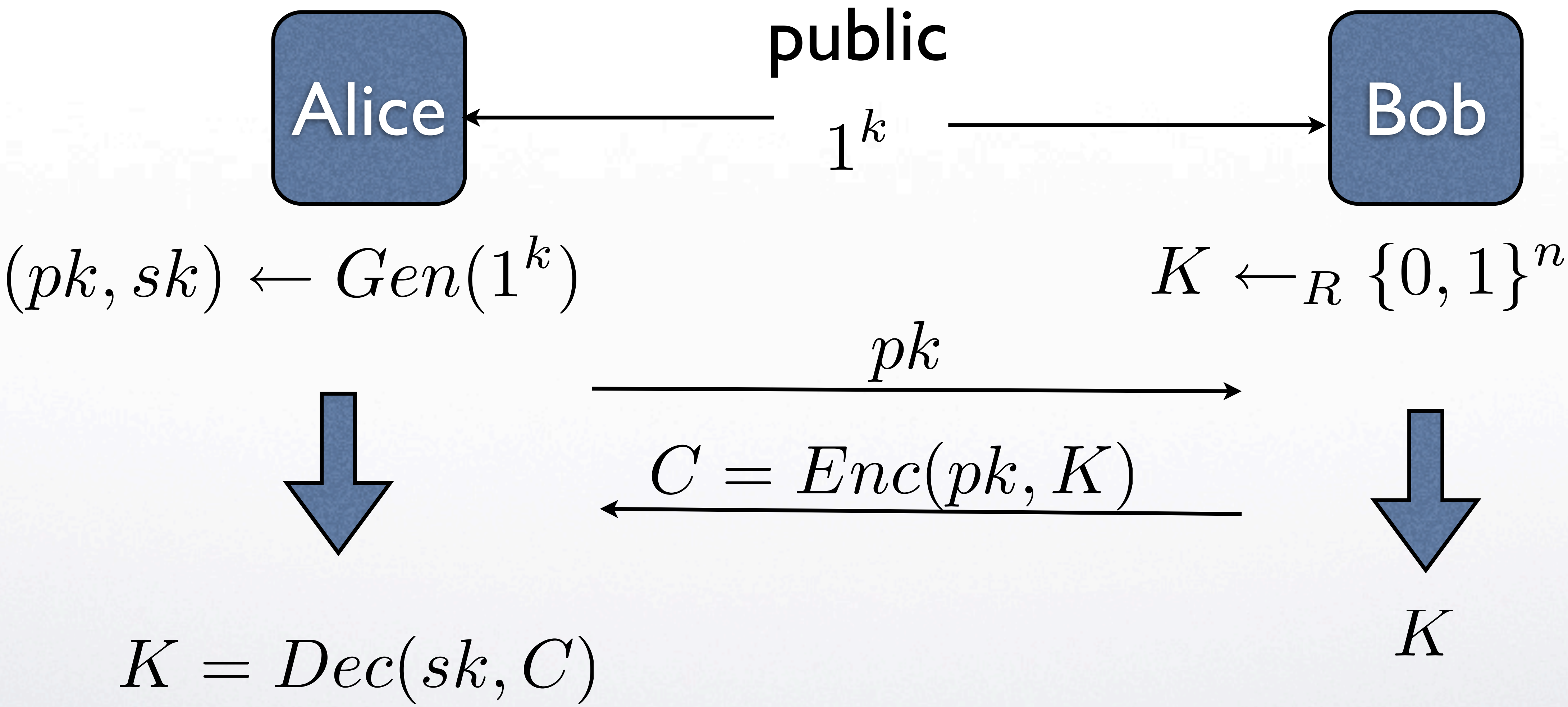
$$Dec(sk, C) = \text{ decryption of } C \text{ under } sk$$

$$\forall M : Dec(sk, Enc(pk, M)) = M$$

PK Enc => Implies KE

- Using a public-key encryption we can achieve key-exchange.

PK Enc Based KE



RSA PK Encryption

public-key = $n = pq, e \in \text{Prime}$

secret-key = $d, \text{s.t.} \forall x : x^{e \cdot d} = x \bmod n$

encryption $c = m^e \bmod n$

decryption $m = c^d \bmod n$

plaintext is an $e-\text{th}$ root of ciphertext

Historical

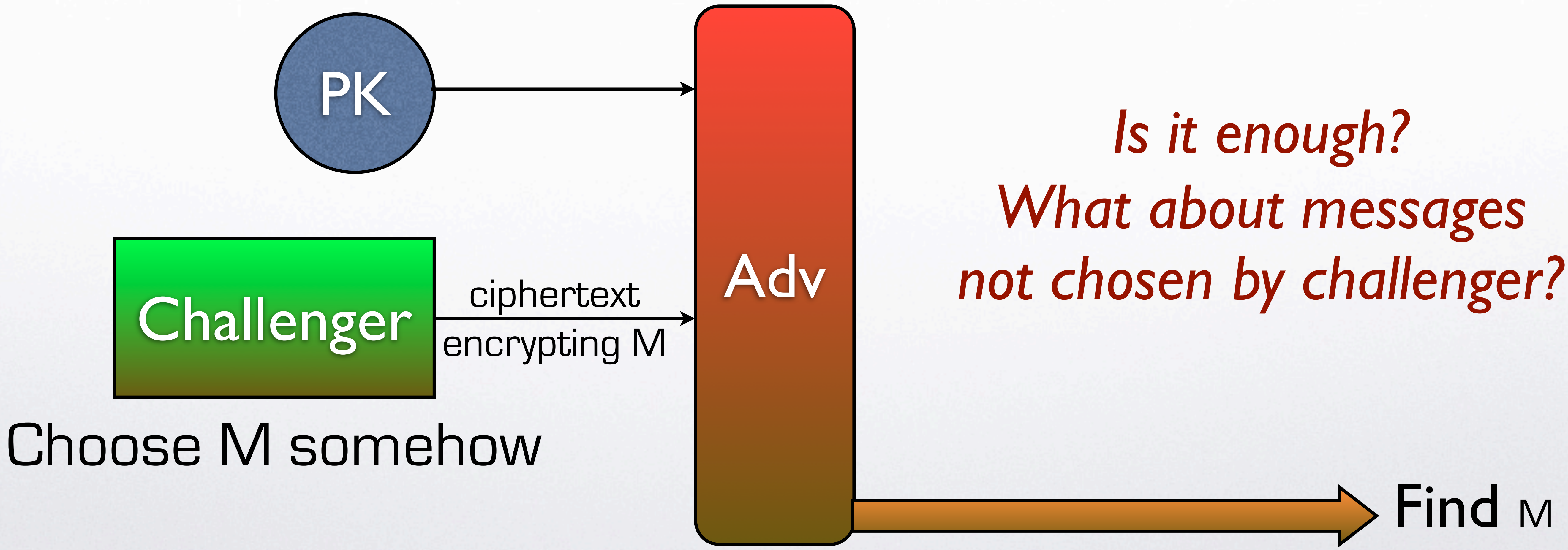
- In 1976, W. Diffie and M. Hellman publish:
  New Directions in Cryptography
  [modern cryptography is born]

- In 1978, R. Rivest, A. Shamir, L. Adleman publish: *A Method for Obtaining Digital Signatures and Public-key Cryptosystems*. The first public-key encryption.

- Interesting: Both techniques were discovered earlier by the CESG (Communications Electronics Security Group, UK) in '74 and '73 respectively.

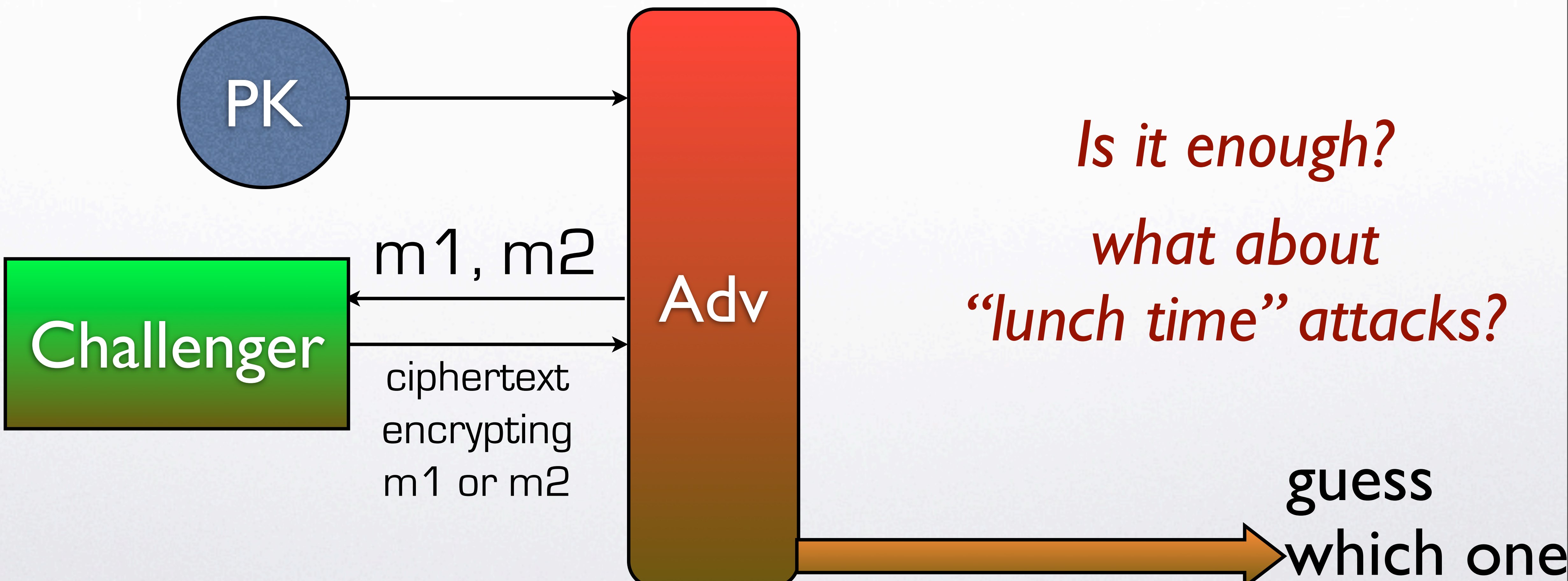
Modeling Security, I

- Ciphertext-only attack for PK encryption



Modeling Security, II

- IND-CPA attack for PK encryption

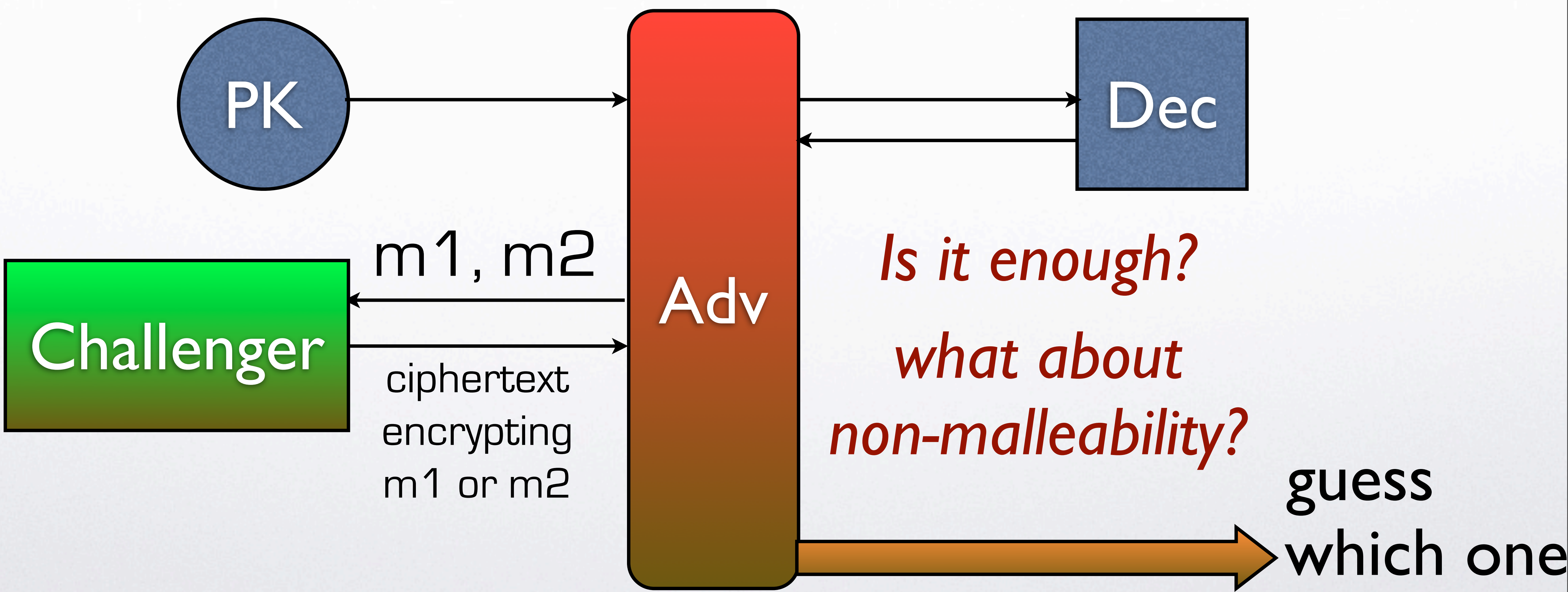


Is it enough?

what about “lunch time” attacks?

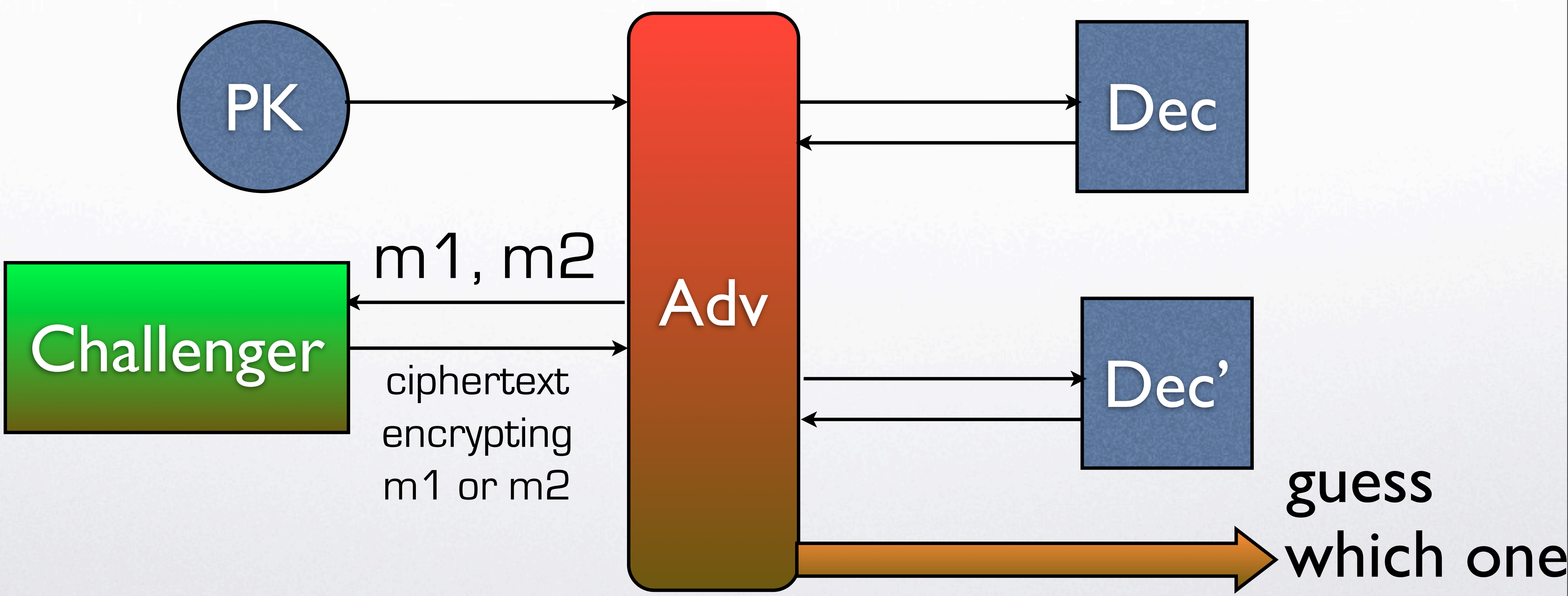
Modeling Security, III

- IND-CCA1 attack for PK encryption



Modeling Security, IV

- IND-CCA2 attack for PK encryption



Check:

- Is the RSA cryptosystem (in the way just presented) secure in IND-CCA2 sense?
    - think: encryption is "deterministic"
    - think: ciphertexts are "malleable"

Encryption vs. Authentication

- What about authenticating the origin of the message?
- Public-key equivalent of a MAC ?

Digital Signatures

- In some sense the reverse of public-key encryption:
  - Given message apply to it secret-key to obtain *digital signature*. Release message and signature.
  - Using the public-key third parties *verify* the message-signature pair as coming from the owner of the public-key pair.

RSA Signature

public-key = $n = pq, e \in \text{Prime}$

secret-key = $d, \text{s.t.} \forall x : x^{e \cdot d} = x \bmod n$

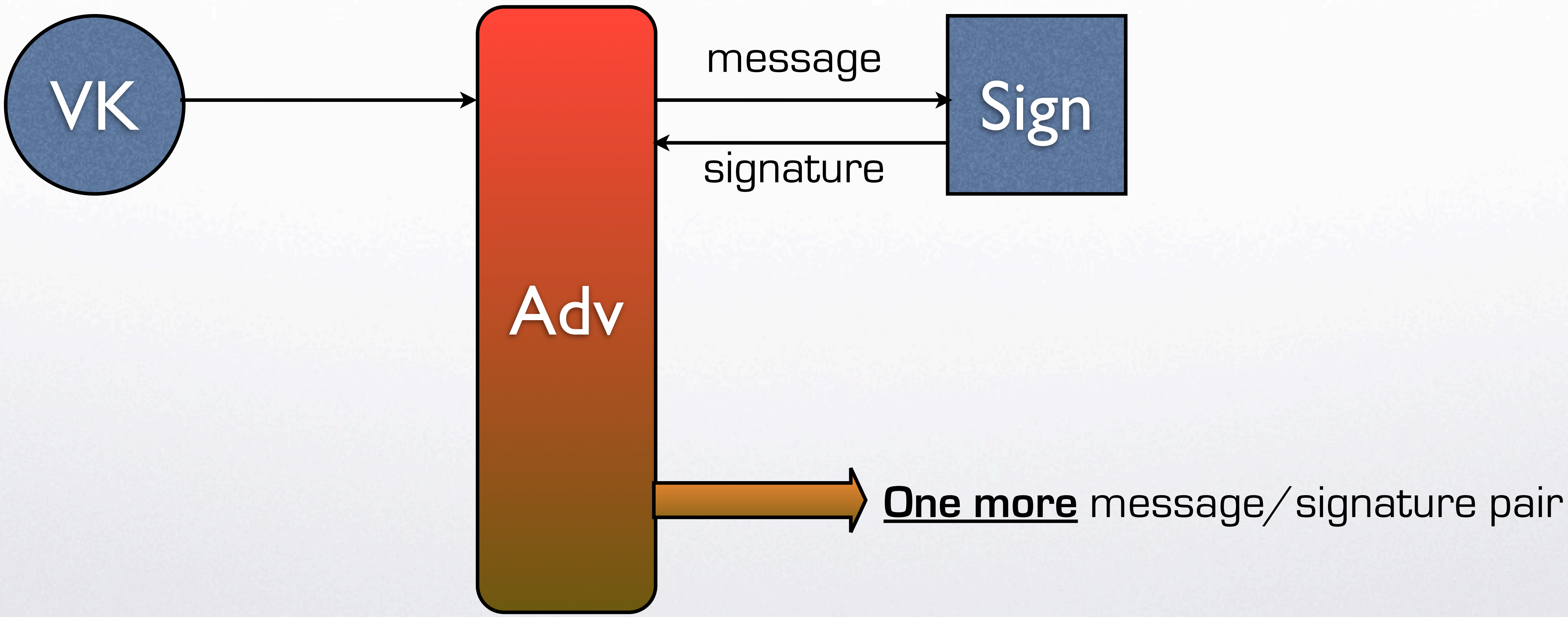
signing $\sigma = m^d (\bmod n)$

verification $\sigma^e \stackrel{?}{=} m (\bmod n)$

signature is an $e - \text{th}$ root of message

Modeling Security

- Adaptive One-more Forgery Attack for Digital signatures



Check:

- Is the RSA signature (in the way just presented) secure against an Adaptive One-more Forgery Attack?
  - think: how can a proof of security be structured?
  - The reductionist approach: does an attacker imply an e-th root solver?

Provable Security, I

- RSA encryption and signature as described in previous slides are not secure according to the given security definitions.
    - some more work is required to obtain the suitable reductions.

Provable Security, II

- A cryptographic scheme has provable security, if:
  - the existence of an adversary implies the solvability of a well-known hard problem.
    - Note: no unconditional proofs of security are known: cf. the **P vs. NP problem**.

Modern Schemes

- Diffie Hellman Key Exchange. DLog based
- RSA encryption & signatures *with random padding schemes.* Root finding based + Random Oracle
- ElGamal public-key encryption and signatures. The digital signature algorithm. DLog based + Random Oracle
- Paillier public-key encryption. Root Finding based
- Strong-RSA based digital signatures. Root Finding based

Challenges of PK Crypto

- 99% of schemes based on related assumptions. Diversify?
- Basic PK cryptographic operation: modular exponentiation. Expensive!

Comparison on a 64-bit Athlon

AES 128 bits approx. 80 MB/s

RSA-sig 1024 bits **sign**=88 Kb/s **verify**=1.56 Mb/s

note the asymmetry

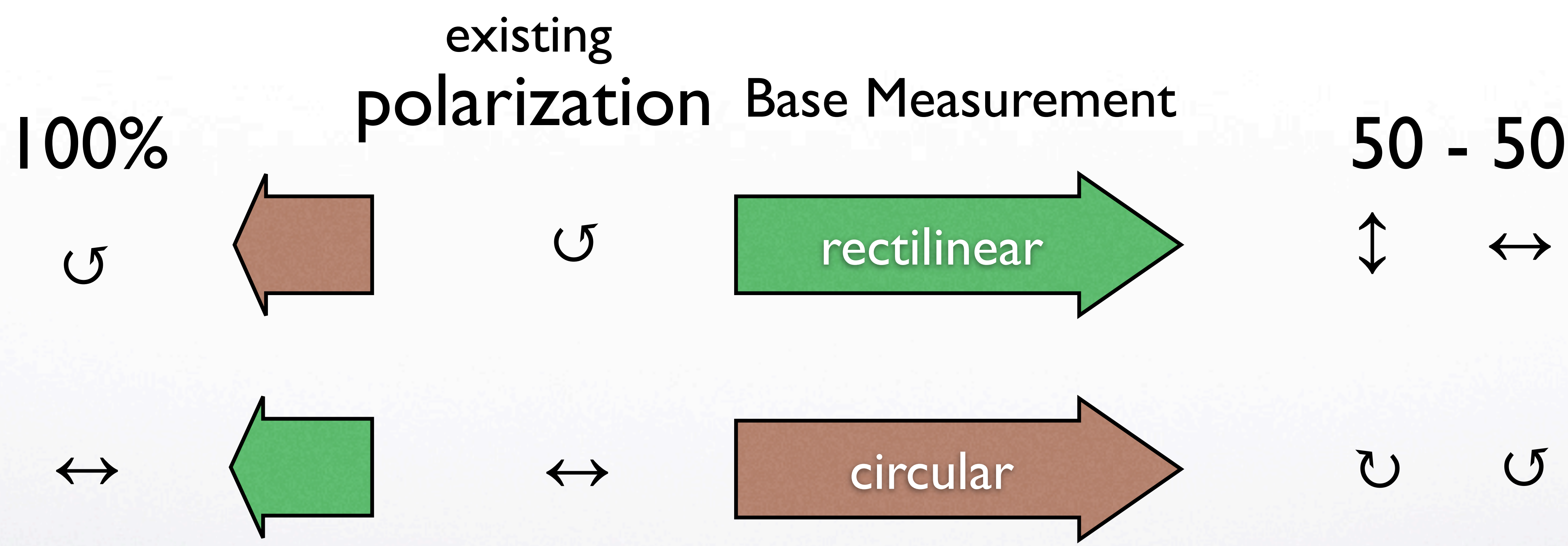
Exotic Cryptography

- Based on other computational problems:
  - Knapsack Problem-based cryptography.
  - Lattices-based cryptography.
  - Braid Groups-based cryptography.
  - Random Linear Codes-based cryptography.
  - Polynomial Reconstruction-based cryptography.

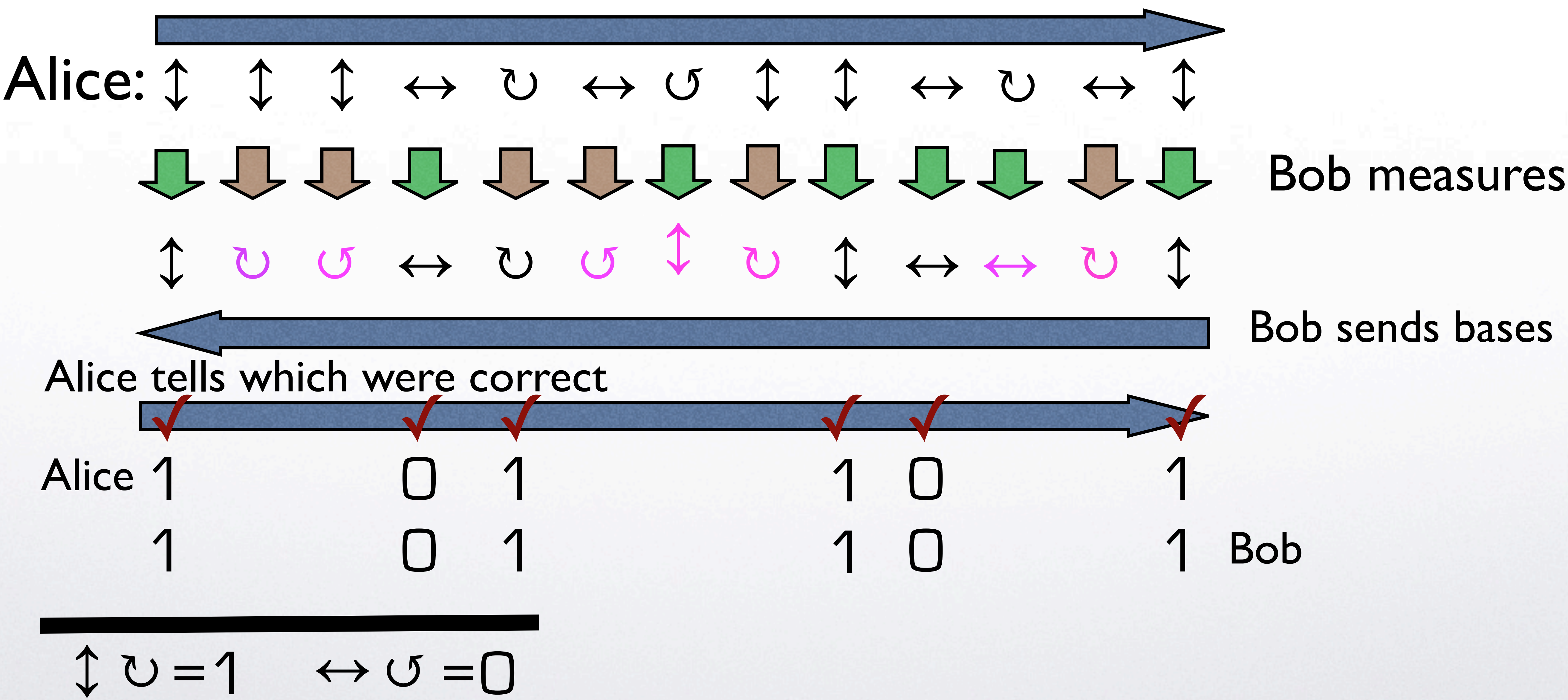
Quantum Cryptography

- Heisenberg uncertainty principle:
  - There are complementary variable measurements that the observation of one affects the other.
    - The polarization of a photon has such measurements: horizontal/vertical vs circular polarization.
    - A measurement will stabilize the variable.

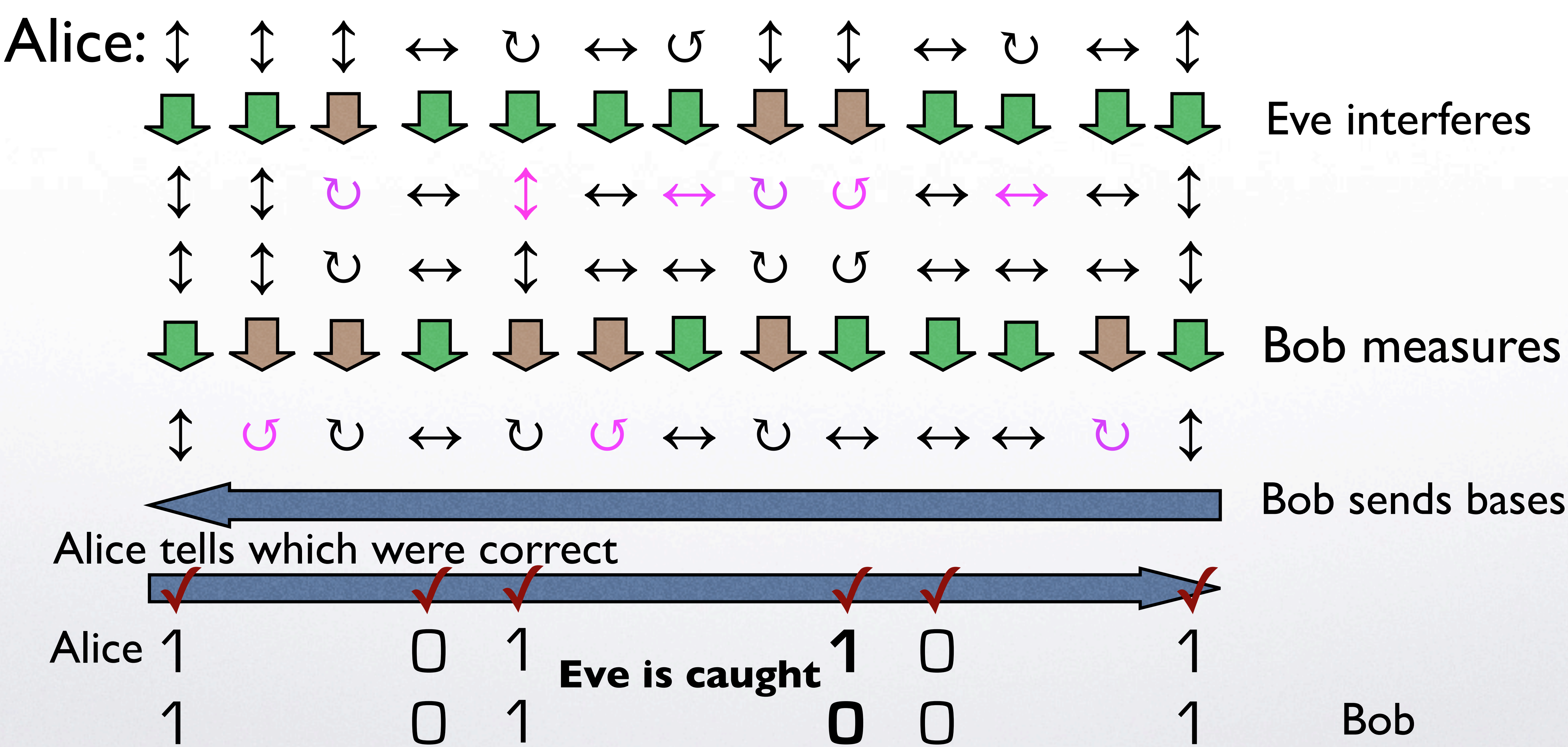
Photon Polarization & Measurement



Quantum Key Exchange



Quantum Adversary Detection



Quantum Key Exchange, II

- Different (randomly selected) portions of photon stream can be used for key extraction and for estimating of eavesdropper interference.
- QKE *is a reality*. But is it marketable?
  - current implementations <100km.
  - wireless is possible!
  - check: www.magiqtech.com (and *buy* for < $50K)

Quantum Computing

- *Not a reality at present.* Should not to be confused with Quantum Cryptography. It is much harder.
  - In a quantum computer "Qubits" should be processed in a coordinated fashion whereas quantum crypto is a "single qubit operation".
- If realized it can bring the death of many classical crypto algorithms (but most likely not all - "exotic" crypto remains secure).
- Realization problems: *Decoherence* requires quantum error correction which in turn increases the required number of qubits by several orders of magnitude.